AI Magazine Volume 26 Number 4 (2005) (© AAAI)

25th Anniversary Issue

If Not Turing's Test, Then What?

Paul R. Cohen

■ If it is true that good problems produce good science, then it will be worthwhile to identify good problems, and even more worthwhile to discover the attributes that make them good problems. This discovery process is necessarily empirical, so we examine several challenge problems, beginning with Turing's famous test, and more than a dozen attributes that challenge problems might have. We are led to a contrast between research strategies—the successful "divide and conquer" strategy and the promising but largely untested "developmental" strategy—and we conclude that good challenge problems encourage the latter strategy.

Turing's Test: The First Challenge

More than fifty years ago, Alan Turing proposed a clever test of the proposition that machines can think (Turing 1950). He wanted the proposition to be an empirical, one and he particularly wanted to avoid haggling over what it means for anything to think.

> We now ask the question, 'What will happen when a machine takes the part of [the man] in this game?' Will the interrogator decide wrongly as often when the game is played like this as he does when the game is played between a man and a woman? These questions replace our original, "Can machines think?"

More recently, the test has taken slightly different forms. Most contemporary versions ask simply whether the interrogator can be fooled into identifying the machine as human, not necessarily a man or a woman.

There are many published arguments about Turing's paper, and I want to look at three *kinds* of argument. One kind says Turing's test is irrelevant; another concerns the philosophy of machines that think; the third is methodological.

Ignore It, and Maybe It Will Go Away…

Blay Whitby (1996) offers this humorous history of the Turing test:

> 1950–1966: A source of inspiration to all concerned with AI.
>
> 1966–1973: A distraction from some more promising avenues of AI research.
>
> 1973–1990: By now a source of distraction mainly to philosophers, rather than AI workers.
>
> 1990: Consigned to history.

Perhaps Whitby is right, and Turing's test should be forgotten as quickly as possible and should not be taught in schools. Plenty of people have tried to get rid of it. They argue that the test is methodologically flawed and is based in bad philosophy, that it exposes cultural biases and naïveté about what Turing calls the "programming" required to pass the test. Yet the test still stands as a grand challenge for artificial intelligence, it is part of how we define ourselves as a field, it won't go away, and, if it did, what would take its place?

Turing's test is not irrelevant, though its role has changed over the years. Robert French's (2000) history of the test treats it as an indicator of attitudes toward AI. French notes that among AI researchers, the question is no longer, "What should we do to pass the test?" but, "Why can't we pass it?" This shift in attitudes—from hubris to a gnawing worry that AI is on the wrong track—is accompanied by another, which, paradoxically, requires even more encompassing and challenging tests. The test is too behavioral—the critics say—too oriented to language, too symbolic, not grounded in the physical world, and so on. We needn't go into the details of these arguments to see that Turing's test continues to influence the debate on what AI can or should do.

There is only one sense in which Turing's test is irrelevant: almost nobody thinks we should devote any effort in the foreseeable future to trying to pass it. In every other sense, as a historical challenge, a long-term goal for AI, a philosophical problem, a methodological case study, and an indicator of attitudes in AI, the Turing test remains relevant.

Turing the Philosopher

Would Turing mind very much that his test no longer has the role he intended? If we take Turing at his word, then it is not clear that he ever intended his test to be attempted:

> There are already a number of digital computers in working order, and it may be asked, 'Why not try the experiment straight away?...' The short answer is that we are not asking whether all digital computers would do well in the game nor whether the computers at present available would do well, but whether there are imaginable computers which would do well.

Daniel Dennett thinks Turing intended the test as "a conversational show-stopper," yet the philosophical debate over Turing's test is ironically complicated. As Dennett says, "Alas, philosophers—amateur and professional—have instead taken Turing's proposal as a pretext for just the sort of definitional haggling and interminable arguing about imaginary counterexamples he was hoping to squelch" (Dennett 1998).

I am confident that if we pose the right sorts of challenges, then we will make good progress in AI.

Philosophers wouldn't be interested if Turing hadn't been talking about *intentional* attributes of machines—beliefs, goals, states of knowledge, and so on—and because we in AI are about building machines with intentional attributes, philosophers will always have something to say about what we do. However, even if the preponderance of philosophical opinion was that machines can't think, it probably wouldn't affect the work we do. Who among us would stop doing AI if someone proved that machines can't think? I would like to know whether there is life elsewhere in the universe; I think the question is important, but it doesn't affect my work, and neither does the question of whether machines can think. Consequently, at least in this article, I am unconcerned with philosophical arguments about whether machines can think.

Turing's Test as Methodology

Instead I will focus on a different, entirely methodological question: *Which attributes of tests for the intentional capabilities of machines lead to more capable machines?* I am confident that if we pose the right sorts of challenges, then we will make good progress in AI. This article is really about what makes challenges good, in the sense of helping AI researchers make progress. Turing's test has some of these good attributes, as well as some really bad ones.

The one thing everyone likes about the Turing test is its *proxy function*, the idea that the test is a proxy for a great many, wide-ranging intellectual capabilities. Dennett puts it this way:

> "Nothing could possibly pass the Turing test by winning the imitation game without being able to perform indefinitely many other intelligent actions. ... [Turing's] test was so severe, he thought, that nothing that could pass it fair and square would disappoint us in other quarters." (Dennett 1998)

No one in AI claims to be able to cover such a wide range of human intellectual capabilities. We don't say, for instance, "Nothing could possibly perform well on the UCI machine learning test problems without being able to perform indefinitely many other intelligent actions." Nor do we think word sense disambiguation, obstacle avoidance, image segmentation, expert systems, or beating the world chess champion are proxies for indefinitely many other intelligent actions, as Turing's test is. It is valuable to be reminded of the breadth of human intellect, especially as our field fractures into subdisciplines, and I suppose one methodological contribution of Turing's test is to remind us to aim for broad, not narrow competence. However, many find it easier and more productive to specialize, and, even though we all know about Turing's test and many of us consider it a worthy goal, it isn't enough to encourage us to develop broad, general AI systems.

So in a way, the Turing test is impotent: It has not convinced AI researchers to try to pass it. Paradoxically, although the proxy function is the test's most attractive feature, it puts the cookie jar on a shelf so high that nobody reaches for it. Indeed, as Pat Hayes and Ken Ford point out, "The Turing Test is now taken to be simply a rather fancy way of stating that the goal of AI is to make an artificial human being" (Hayes and Ford 1995).

A second notable methodological failing of Turing's test is that it pushes many aspects of intelligence into one test that has a yes or no answer. This isn't necessary. We could follow the lead of the multiple intelligences movement in cognitive psychology and devise tests of different sorts of intelligence. In fact, Turing's test is not even very complete, when viewed in terms of, say, Howard Gardner's catalog of intelligences (Gardner 1983). It focused mostly on logical, linguistic, and interpersonal intelligence, not on intrapersonal, bodily-kinesthetic, naturalist, musical, and visual-spa-

tial intelligence (rounding out the eight in Gardner's catalog).

Robert French goes further and criticizes the test for its focus on culturally oriented human intelligence: "The Test provides a guarantee not of intelligence but of culturally-oriented human intelligence" (French 2000). The test also says nothing about neonatal or infant intelligence—which I think are worth understanding and emulating. In fact, to the extent that Turing provided an operational definition of intelligence with his test, it was a kind of middle-class English intellectual, midcentury dinner party kind of intelligence and represents neither the pinnacle nor the broad plain of human intellectual capability.

A third failing of the test is that it cannot be passed today. Tests should be challenging, but tests that *cannot* be passed provide no information. Periodically the Loebner Prize organization conducts a sort of Turing test. Consider the following brief snippet of a dialogue I had with one of the entrants:

PC: Where is your nose?

Entrant: Oh, my nose? Why are you so interested?

PC: Is it near your eyes?

Entrant: Such is Human Perversity.

PC: Is your nose between your mouth and your eyes?

Entrant: I can't tell you that.

PC: How many hands do I have?

Entrant: I don't understand why you have, and furthermore, I don't really care.

PC: My daughter's name is Allegra. What is my daughter's name?

Entrant: Is she a big part of your problem?

It takes only a moment to see I am not conversing with a human being. My point isn't to make fun of the people who try to win the Loebner Prize, nor do I think this snippet is the best that we can do with today's technology. My point is that even the very best technology in AI today would not bring us anywhere close to passing the Turing test, and this has a very bad consequence: Few AI researchers try to pass the test.

Said more positively, a good test is only slightly out of reach, and the path to success is at least partly clear.

Not only is Turing's goal remote, but attempts to pass his test are not diagnostic: They don't tell us what to do to pass the test next time. Blay Whitby puts it this way: "If the Turing test is read as something like an operational definition of intelligence, then two very important defects of such a test must be considered. First, it is all or nothing: it gives no indication as to what a partial success might look like. Second, it gives no direct indications as to how success might be achieved" (Whitby 1996). And Dennett notes the asymmetry of the test: "Failure on the Turing test does not predict failure on ... others, but success would surely predict success" (Dennett 1998). Attempting the test is a bit like failing a job interview: Were my qualifications suspect? Was it something I said? Was my shirt too garish? All I have is a rejection letter—the same content-free letter that all but one other candidate got—and I have no idea how to improve my chances next time.

So let's recognize the Turing test for what it is: A goal, not a test. Tests are diagnostic, and specific, and predictive, and Turing's test is neither of the first two and arguably isn't predictive, either. Turing's test is not a challenge like going to the moon, because one can see how to get to the moon and one can test progress at every step along the way. The main functions of Turing's test are these: To substitute tests of *behavior* for squabbles about definitions of intelligence, and to remind us of the enormous breadth of human intellect. The first point is accepted by pretty much everyone in the AI community, the second seems not to withstand the social and academic pressure to specialize.

So now we must move on to other tests, which, I hope, have fewer methodological flaws; tests that work for us.

So let's recognize the Turing test for what it is: A goal, not a test.

New Challenges

Two disclaimers: First, artificial intelligence and computer science do not lack challenge problems, nor do we lack the imagination to provide new ones. This section is primarily about *attributes* of challenge problems, not about the problems, themselves. Second, assertions about the utility or goodness of particular attributes are merely conjectures and are subject to empirical review. Now I will describe four problems that illustrate conjectured good attributes of challenge problems.

Challenge 1: Robot Soccer

Invented by Alan Mackworth in the early 1990s to challenge the simplifying assumptions of good old-fashioned AI (Mackworth 1993), robot soccer is now a worldwide movement. No other AI activity has involved so many people at universities, corporations, primary and secondary schools, and members of the public.

What makes robot soccer a good challenge problem? Clearly the problem itself is exciting, the competitions are wild, and students stay up

late working on their hardware and software. Much of the success of the robot soccer movement is due to wise early decisions and continuing good management. The community has a clear and easily stated fifty-year goal: to beat the human world champion soccer team. Each year, the community elects a steering committee to moderate debate on how to modify the rules and tasks and league structure for the coming year's competition. It is the responsibility of this committee to steer the community toward its ultimate goal in manageable steps. The bar is raised each year, but never too high; for instance, this year there will be no special lighting over the soccer pitches.

From the first, competitions were open to all, and the first challenges could be accomplished. The cost of entry was relatively low: those who had robots used them, those who didn't played in the simulation league. The first tabletop games were played on a misshapen pitch—a common ping-pong table—so participants would not have to build special tables. Although robotic soccer seems to offer an endless series of research challenges, its evaluation criterion is familiar to any child: win the game! The competitions are enormously motivating and bring in thousands of spectators (for example, 150,000 at the 2004 Japan Open). Two hundred Junior League teams participated in the Lisbon competition, helping to ensure robotic soccer's future.

It isn't all fun and games: RoboCup teams are encouraged to submit technical papers to a symposium. The best paper receives the RoboCup Scientific Challenge Award.

A defining feature of the Handy Andy challenge, one it shares with Turing's test, is its universal scope.

Challenge 2: Handy Andy

As ABC News recently reported, people find ingenious ways to support themselves in college: "For the defenders of academic integrity, their nemesis comes in the form of a bright college student at an Eastern university with a 3.78 GPA. Andy—not his real name—writes term papers for his fellow students, at rates of up to $25 a page."

Here, then, is the Handy Andy challenge: *Produce a five-page report on any subject.* One can administer this test in vivo, for instance, as a service on the World Wide Web; or in a competition. One can imagine a contest in which artificial agents go against invited humans—students and professionals—in a variety of leagues or tracks. Some leagues would be appropriate for children. All the contestants would be required to produce three essays in the course of, say, three hours, and all would have access to the web. The essay subjects would be designed with help from education professionals, who also would be responsible for scoring the essays.

As a challenge problem, Handy Andy has several good attributes, some of which it shares with robot soccer. Turing's test requires simultaneous achievement of many cognitive functions and doesn't offer partial credit to subsets of these functions. In contrast, robot soccer presents a *graduated series of challenges:* it gets harder each year but is never out of reach. The same is true of the Handy Andy challenge. In the first year, one might expect weak comprehension of the query, minimal understanding of web pages, and reports merely cobbled together from online sources. Later, one expects better comprehension of queries and web pages, perhaps a clarification dialog with the user, and some organization of the report. Looking further, one envisions strong comprehension and not merely assembly of reports but some original writing. The first level is within striking distance of current information retrieval and text summarization methods. Unlike the Turing test—an all-or-nothing challenge of heroic proportions—we begin with technology that is available today and proceed step-by-step toward the ultimate challenge.

Because a graduated series of challenges begins with today's technology, we do not require a preparatory period to build prerequisites, such as sufficient commonsense knowledge bases or unrestricted natural language understanding. This is a strong methodological point because those who wait for prerequisites usually cannot predict when they will materialize, and in AI things usually take longer than expected. The approach in Handy Andy and robot soccer is to *come as you are* and develop new technology over the years in response to increasingly stringent challenges.

The five-page requirement of the Handy Andy challenge is arbitrary—it could be three pages or ten—but the required length should be sufficient for the system to make telling mistakes. A test that satisfies the *ample rope requirement* provides systems enough rope to hang themselves. The Turing test has this attribute and so does robot soccer.

A defining feature of the Handy Andy challenge, one it shares with Turing's test, is its *universal scope.* You can ask about the poetry of Jane Austen, how to buy penny stocks, why the druids wore woad, or ideas for keeping kids busy on long car trips. Whatever you ask, you get five pages back.

The universality criterion entails something about evaluation: we would rather have a system produce crummy reports on any subject than excellent reports on a carefully selected,

narrow range of subjects. Said differently, the challenge is first and foremost to handle any subject and only secondarily to produce excellent reports. If we can handle any subject, then we can imagine how a system might improve the quality of its reports. On the other hand, half a century of AI engineering leaves me skeptical that we will achieve the universality criterion if we start by trying to produce excellent reports about a tiny selection of subjects. It's time to grasp the nettle and go for all subjects, even if we do it poorly.

The web already exists, already has near universal coverage, so we can achieve the universality criterion by making good use of the knowledge the web contains. Our challenge is not to build a universal knowledge base but to make better use of the one that already exists.

Challenge 3: Never-Ending Language Learning

Proposed by Murray Burke in 2002, this challenge takes up a theme of Lenat and Feigenbaum's (1987) paper "On the Thresholds of Knowledge." That paper suggested knowledge-based systems would eventually know enough to read online sources and, at that point, would "go critical" and quickly master the world's knowledge. There are no good estimates of when this might happen. Burke's proposal was to focus on the bootstrapping relationship between learning to read and reading to learn.

We always must worry that challenge problems reward clever engineering more than scientific research. Robot soccer has been criticized on these grounds. Among its many positive attributes, never-ending language learning presents us with some fascinating scientific hypotheses. One states that we have done enough work on the semantics of a core of English to bootstrap the acquisition of the whole language. Another hypothesis is that learning by reading provides sufficient information to extend an ontology of concepts and so drive the bootstrapping. Both hypotheses could be wrong; for example, some people think that the meanings of concepts must be grounded in interaction with the physical world and that no amount of reading can make up for a lack of grounding. In any case, it is worth knowing whether one can learn what one needs to understand text from text itself.

Challenge 4: The Virtual Third Grader

One answer to the question, "if not the Turing test, then what?" was suggested by David Gunning in 2004: If we cannot pass the Turing test today, then perhaps we should set up a "cognitive decathlon" or "qualifying trials" of capabilities that, collectively, are required for Turing's test. Howard Gardner's inventory of multiple intelligences is one place to look for these capabilities. However, it isn't clear how to test whether machines have them. Another place to look is elementary school. Every third-grader is expected to master the skills in table 1. All of them can be tested, although some tests will involve subjective judgments. Here is what my daughter wrote for her "convincing letter" assignment:

> Dear Disney,
>
> It disturbs me greatly that in every movie you make with a dragon, the dragon gets killed by a knight. Please, if you could change that, it would be a great happiness to me. The Dragon is my school mascot. The dragon isn't really bad, he/she is just made bad by the villan [sic]. The dragon is not the one who should be killed. For example, Sleeping Beauty, the dragon is under the villaness's [sic] power, so it is not neccisariliy [sic] bad or evil. Please change that.
>
> Your sad and disturbed writer,
>
> *Allegra.*

Although grading these things is subjective, there are many diagnostic criteria for good letters: The author must assert a position (stop killing the dragons) and reasons for it (the dragon is my school mascot, and dragons aren't intrinsically bad). Extra points might be given for tact, for suggesting that the recipient of the letter isn't malicious, just confused (the dragon isn't the one who should be killed, you got it wrong, Disney!)

Criteria for Good Challenges

You, the reader, probably have several ideas for challenge problems. Here are some practical suggestions for refining these ideas and making them work on a large scale. The success of robot soccer suggests starting with easily understood long-term goals (such as beating the human world soccer team) and an organization whose job is to steer research and development in the direction of these goals. The challenge should be administered frequently, every few weeks or months, and the rules should be changed at roughly the same frequency to drive progress toward the long-term goals.

The challenge itself should test important cognitive functions. It should emphasize comprehension, semantics, and knowledge. It should require problem solving. It should not "drop the user at approximately the right location in information space and leave him to fend for himself," as Edward Feigenbaum once put it.

A good challenge has simple success criteria.

The challenge itself should test important cognitive functions. It should emphasize comprehension, semantics, and knowledge. It should require problem solving.

Understand and follow instructions

Communicate in natural language
(for example, dialog)

Learn and exercise procedures
(for example, long division, outlining a report)

Read for content
(for example, show that one gets the main points of a story)

Learn by being told
(for example, life was hard for the pioneers)

Common sense inference
(for example, few people wanted to be pioneers)
and learning from commonsense inference

Understand math story problems and solve them correctly

Master a lot of facts (math facts, history facts, and so on).
Mastery means using the facts to answer questions
and solve problems.

Prioritize
(for example, choose one book over another,
decide which problems to do on a test)

Explain something
(for example, why plants need light)

Make a convincing argument
(for example, why recess should be longer)

Make up and write a story about an assigned subject
(for example, Thanksgiving)

Table 1. Third-Grade Skills (thanks to Carole Beal).

However an attempt is scored, one should get specific, diagnostic feedback to help one understand exactly what worked and what didn't. Scoring should be transparent so one can see exactly why the attempt got the score it did. If possible, scoring should be objective, automatic, and easily repeated. For instance, the machine translation community experienced a jump in productivity once translations could be scored automatically, sometimes daily, instead of subjectively, slowly, and by hand.

The challenge should have a kind of monotonicity to it, allowing one to build on previous work in one's own laboratory and in others'. This "no throwaways" principle goes hand-in-hand with the idea of a graduated series of challenges, each slightly out of reach, each providing ample rope for systems to hang themselves, yet leading to the challenge's long-term goals. It follows from these principles that the challenge itself should be easily modified, by changing rules, initial conditions, requirements for success, and so on.

A successful challenge captures the hearts and minds of the research community. Popular games and competitions are good choices, provided that they require new science. The cost of entry should be low; students should be able to scrape together sufficient resources to participate, and the organizations that manage challenges should make grants of money and equipment as appropriate. All participants should share their technologies so that new participants can start with "last year's model" and have a chance of doing well.

In addition to these pragmatic and, I expect, uncontroversial suggestions, I would like to suggest three others which are not so obviously right.

First, Turing proposed his test to answer the question "Can machines think?" but this does not mean a challenge for AI must provide evidence for or against the proposition that computers have intentional states and behaviors. I do not think we have any chance of testing this proposition. There are no objective characterizations of human intentional states, and the states of machines can be described in many ways, from the states of registers up to what Newell called the knowledge level. It is at least technically challenging and perhaps impossible to establish correspondences between ill-specified human intentional states and machine states, so the proposition that machines "have" intentional states probably cannot be tested. Perhaps the most we can require of challenge problems is that they include tasks that humans describe in intentional terms.

Second, in any given challenge, we should accept poor performance but insist on universal coverage. I admit that it is hard to define universal coverage, but examples are easily found or imagined: Reading and comprehending any book suitable for five year olds; producing an expository essay on any subject; going up the high street to several stores for the week's shopping; playing Trivial Pursuit; creating a reading list for any undergraduate essay subject; learning classifiers for a thousand data

sets without manually retuning the learner's parameters for each; playing any two-person strategy game well with minimal training; beating the world champion soccer team. Each of these problems requires a wide range of capabilities, or has a great many nonredundant instances, or both. One could not claim success by solving only a part of one of these problems or only a handful of possible problem instances. What should we call a program that plays chess brilliantly? History! What should we call a program that plays any two-person strategy game, albeit poorly? A good start! A program that analyzes the plot of Romeo and Juliet? History! A program that summarizes the plot of any children's book, albeit poorly? A good start! Poor performance and universal scope are preferred to good performance and narrow scope.

My third and final point is related to the last one. Challenge problems should foster what I'll call a *developmental* research strategy instead of the more traditional and generally successful *divide and conquer* strategy. The word *developmental* reminds us that children do many things poorly, yet they are complete, competent agents who learn from each other, and adults, and books, and television, and playing, and physical maturation, and other ways, besides. In children we see gradually increasing competence across many domains. In AI we usually see deep competence in narrow domains, but there are exceptions: robotic soccer teams have played soccer every year since the competitions began. If the organizers had followed the traditional divide and conquer strategy, then the first few annual competitions would have tested bits and pieces—vision, navigation, communication, and control—and we probably would still be waiting to see a complete robotic team play an entire game. Despite the success of divide-and-conquer in many sciences, I don't think it is a good strategy for AI. Robotic soccer followed the other, developmental strategy, and required complete, integrated systems to solve the whole problem. Competent these systems were not, but competence came with time, as it does to children.

Conclusion

In answer to the question, "if not the Turing Test, then what," AI researchers haven't been sitting around waiting for something better; they have been very inventive. There are challenge problems in planning, e-commerce, knowledge discovery from databases, robotics, game playing, and numerous competitions in aspects of natural language. Some are more successful or engaging than others, and I have discussed some attributes of problems that might explain these differences. My goal has been to identify attributes of good challenge problems so that we can have more. Many of these efforts are not supported directly by government, they are the efforts of individuals and volunteers. Perhaps you can see an opportunity to organize something similar in your area of AI.

Acknowledgments

This article is based on an invited talk at the 2004 National Conference on Artificial Intelligence. While preparing the talk I asked for opinions and suggestions from many people who wrote sometimes lengthy and thoughtful assessments and suggestions. In particular, Manuela Veloso, Yolanda Gil, and Carole Beal helped me very much. One fortunate result of the talk was that Edward A. Feigenbaum disagreed with some of what I said and told me so. Our discussions helped me better understand some issues and refine what I wanted to say. The Defense Advanced Research Projects Agency (DARPA), especially Ron Brachman, David Gunning, Murray Burke, and Barbara Yoon, have supported this work (cooperative agreement number F30602-01-1-0583) as they have supported many other activities to review where AI is heading and to help steer it in appropriate directions. I would like to thank David Aha, Michael Berthold, Jim Blythe, Tom Dietterich, Mike Genesereth, Jim Hendler, Lynette Hirschmann, Jerry Hobbs, Ed Hovy, Adele Howe, Kevin Knight, Alan Mackworth Daniel Marcu, Pat Langley, Tom Mitchell, Natasha Noy, Tim Oates, Beatrice Oshika, Steve Smith, Milind Tambe, David Waltz, and Mohammed Zaki for their discussions and help.

References

Dennett, D. 1998. Can Machines Think? In *Brainchildren, Essays on Designing Minds*. Cambridge, MA: MIT Press.

French, R. 2000. The Turing Test: The First Fifty Years. *Trends in Cognitive Sciences* 4(3): 115–121.

Gardner, H. 1983. *Frames of Mind: The Theory of Multiple Intelligences*. New York: Basic Books.

Hayes, P., and Ford, K. 1995. Turing Test Considered Harmful. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, p. 972. San Francisco: Morgan Kaufmann Publishers.

Lenat, D. B., and Feigenbaum, E. A. 1987. On the Thresholds of Knowledge. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, 1173–1182. San Francisco: Morgan Kaufmann Publishers.

Mackworth, A. 1993. On Seeing Robots. In *Computer Vision: Systems, Theory and Applications*, ed. A. Basu and X. Li, 1–13. Singapore: World Scientific Press, 1993. Reprinted in *Mind Readings*, ed. P. Thagard. Cambridge, MA: MIT Press, 1998.

Turing, A. 1950. Computing Machinery and Intelligence. *Mind* 59(236):433–460.

Whitby, B. R. 1996. The Turing Test: AI's Biggest Blind Alley? In *Machines and Thought: The Legacy of Alan Turing, Vol. 1.*, ed. P. Millican and A. Clark. Oxford: Oxford University Press. (www.informatics.sussex.ac.uk/users/blayw/tt.html.)

Paul Cohen is with the Intelligent Systems Division at USC's Information Sciences Institute and he is a Research Professor of Computer Science at USC. He received his Ph.D. from Stanford University in computer science and psychology in 1983 and his M.S. and B.A. in psychology from the University of California at Los Angeles and the University of California at San Diego, respectively. He served as a councilor of the American Association for Artificial Intelligence (AAAI) from 1991 to 1994 and was elected in 1993 as a fellow of AAAI.

AI Magazine Volume 27 Number 2 (2006) (© AAAI)

Comparative Analysis of Frameworks for Knowledge-Intensive Intelligent Agents

Randolph M. Jones and Robert E. Wray

■ A recurring requirement for human-level artificial intelligence is the incorporation of vast amounts of knowledge into a software agent that can use the knowledge in an efficient and organized fashion. This article discusses representations and processes for agents and behavior models that integrate large, diverse knowledge stores, are long-lived, and exhibit high degrees of competence and flexibility while interacting with complex environments. There are many different approaches to building such agents, and understanding the important commonalities and differences between approaches is often difficult. We introduce a new approach to comparing frameworks based on the notions of commitment, reconsideration, and a categorization of representations and processes. We review four agent frameworks, concentrating on the major representations and processes each directly supports. By organizing the approaches according to a common nomenclature, the analysis highlights points of similarity and difference and suggests directions for integrating and unifying disparate approaches and for incorporating research results from one framework into alternatives.

Overview

One frequently taken approach toward achieving human-level intelligent systems is to create foundational software systems that tightly integrate some number of representations and processes deemed sufficient for generating automated intelligent behavior. The design of these foundational software systems, which include both cognitive and agent architectures, have generally been based on some small set of theoretical principles. The agent architecture is an attempt to foster the development of uniform approaches for building intelligent systems. However, large-scale integrated software systems that attempt to approach human levels of intelligence through agent architectures exhibit some core commonalities across different architectures. For example, no matter the chosen architecture, there is a necessity for such systems to encode vast amounts of knowledge in efficient, organized, and maintainable ways. Additionally, these knowledge requirements have had relatively uniform effects on the evolution of these architectures, such that we observe a convergence of essential representations and processes across agent architectures.

A variety of frameworks currently exist for designing human-level intelligent agents and behavior models. Although they have different emphases, each of these frameworks provides coherent, high-level views of intelligent agency. However, more pragmatically, much of the complexity of building intelligent agents occurs in the low-level details, especially when building agents that exhibit high degrees of

competence while interacting in complex environments. To highlight the emphasis of our observations about the knowledge necessary for human-level artificial intelligence, we call such agents *knowledge-intensive agents*. This term is also meant to distinguish such agents from smaller-scale, single-task agents (for example, service brokers) that are often fielded in multiagent systems. Examples of fielded knowledge-intensive agents include a real-time fault diagnosis system on the Space Shuttle (Georgeff and Ingrand 1990) and a real-time model of combat pilots (Jones, Laird, and Nielsen 1999). Knowledge-intensive agents are also often used in "long-life" situations, where a particular agent needs to behave appropriately and maintain awareness of its environment for a long period of time (hours to days) while performing many different activities during the span of its existence. Additionally, knowledge-intensive agents must be engineered such that their knowledge can be easily modified (possibly by both extrinsic and intrinsic processes) as environment and task requirements change during deployment.

Transfer and generalization of results from one framework to others is usually slow and limited. The reasons for such limited transfer include differences in nomenclature and methodology that make it more difficult to understand and apply results, and the necessity of specifying low-level details that are not prescribed by the frameworks but that become important in actual implementation. In addition, high-level agent frameworks do not usually guide the agent developer in many finer-grained implementation issues, meaning that the frameworks underspecify necessary principles to build and field working intelligent agents. Our goal is to develop techniques that will minimize framework-specific descriptions and that bridge the gap between a framework's theory and the details of its implementation, especially clarifying which details are intrinsic to particular approaches and which are not. In the long run, this effort should foster reuse of architectural components and idioms across architectures as well as across individual agent models that use a single architecture.

This article reviews four existing agent frameworks in order to explore what they specify (and do not) about an agent's design and construction. The chosen frameworks have proven successful for building knowledge-intensive agents of various levels of complexity, or specifically address constraints on agents with high levels of competence (such as human behavior models). We identify the representations and agent processes that the frameworks dictate for agent design. This comparative analysis, to our knowledge, is novel and provides insights into the trade-offs inherent in these systems for building intelligent agents. The goal is truly comparative. Each system we review arguably has a unique application niche, and we are not seeking to suggest one framework is better than another. Rather, in comparing them, especially in noting convergences and divergences in knowledge-intensive agent applications, we seek to develop a uniform methodology for comparing frameworks and, ultimately, to speed the development and evolution of architectures by making research results more communicable and transparent to researchers not working within the specific subfield of AI or cognitive science in which new architecture developments are made.

One important result of this analysis is the observation that no single framework we review here directly supports all of the representations that have been usefully employed in various knowledge-intensive agent systems. The result is that an agent designer who adopts any of these particular frameworks often must also develop application-specific solutions for the representations and processes not directly supported by the chosen framework. This situation is undesirable because it leads to ad hoc solutions for different agent applications created within the same framework. Ad hoc solutions in turn increase development costs by hampering reuse. While the current analysis does not provide a complete set of necessary representations and processes for knowledge-intensive agents, it does serve as a starting point for future architectural research: creating and deploying robust, lower-cost, long-lived agent applications makes it essential to have direct architectural support of all the basic representations and processes.

Review of Agent Frameworks

We introduce four mature frameworks for intelligent agents that represent quite different theoretical traditions (philosophical and logical, functional, psychological, and formal computational). We have intentionally selected exemplar frameworks that are somewhat different in character in order to provide a broad first cut at an encompassing review. Our intent is to consider the primary representational constructs and processes directly supported by each. We focus on these aspects of agent frameworks because an agent is essentially the sum of a system's knowledge (represented with particular constructs) and the processes that operate on those constructs (Russell and Norvig 1994).

We focus on frameworks that have been used to build large-scale, highly capable agent systems because different programming paradigms are likely appropriate for systems with leaner knowledge. An example motivating factor for this analysis is the recognition that *implemented* BDI and Soar systems, while originating from different theoretical starting points, have converged on similar solutions for large-scale systems. However, this analysis can be extended to other frameworks as well, with still other representations and processes (for example, 4D/RCS [Albus 2001], ACT-R [Anderson and Lebiere 1998], Icarus [Langley, Choi, and Shapiro 2004], and RETSINA [Payne, Singh, and Sycara 2002]). In the long term, we will extend our analysis to these other frameworks as well.

BDI

The BDI (beliefs, desires, intentions) framework grew out of Bratman's (1987) theory of human practical reasoning. BDI is now a popular logic-based methodology for building competent agents (Georgeff and Lansky 1987; Rao and Georgeff 1995; Wooldridge 2000). A basic assumption in BDI is that intelligent agents ought to be rational in a formal sense, meaning rationality (as well as other properties) can be logically proven. Actions arise from internal constructs called intentions. An intelligent agent cannot make decisions about intentions until it has at least some representation of its beliefs about its situation. That is, the agent must maintain a set of beliefs about what is true in the world. Given a particular set of beliefs, there may be many different situations that the agent might consider desirable. Given limited resources, however, the agent can often only act on some subset of these desires, so the agent selects a subset, its intentions, to pursue. Using BDI terminology, the entire set of relevant activities represents the agent's *desires*, and the set of currently selected actions that address some subset of those desires are the *intentions*.

BDI was also designed with specific high-level constraints on intelligent behavior in mind. First, as mentioned, the framework insists on rational agents, in the sense that a BDI agent's actions must always be logically consistent with its combination of beliefs and goals. This property is not true of some of the other frameworks we analyze, particularly those with a heavy emphasis on psychology (where intelligent behavior that is not strictly rational is observed with some frequency). Second, the BDI framework also emphasizes supporting groups of agents that interact with each other. BDI is a high-level framework that has a number of distinct implementations, among them IRMA (Bratman, Israel, and Pollack 1988), PRS (Georgeff and Lansky 1987), dMARS (d'Inverno et al. 1997), JACK (Howden et al. 2001) and JAM (Huber 1999). Our discussion includes some small examples of differences in implemented architectures where those architectures have made specific commitments beyond the general BDI framework. However, in general our consideration of BDI is meant to be consistent with the common framework as presented by Wooldridge (2000).

GOMS

GOMS (goals, operators, methods, and selections) is a methodology based in psychology and human-computer interaction (Card, Moran, and Newell 1983). GOMS is not strictly an agent framework, but it formalizes many details of high-level human reasoning and interaction. However, GOMS is particularly interesting because knowledge-intensive agents are often used to simulate human behavior. Although GOMS has not been used to develop large-scale systems, it has been used to represent the human knowledge necessary for performing many tasks, including complex human activity. We include GOMS because the representation and process regularities it has identified are critical for knowledge-intensive agents that will encode this type of knowledge. In addition, improvements in efficiency increasingly allow executable cognitive models to compete with AI architectures in application areas (for example, John, Vera, and Newell [1994]).

GOMS systems explicitly encode hierarchical task decompositions, starting with a top-level task goal plus a number of methods, or plans, for achieving various types of goals and subgoals. Each goal's plan specifies a series of actions (called operators by the GOMS community) invoking subgoals or primitive actions to complete the goal. Selection rules provide conditional logic for choosing between plans based on the agent's current set of beliefs.

One key feature of GOMS is its support for hierarchical task decomposition. Although a hierarchical model is not a strict requirement, among the frameworks examined here GOMS most strongly encourages and supports hierarchical solutions. Like BDI, GOMS is a high-level framework, realized in a number of individual implementations, such as GLEAN (Kieras et al. 1995), APEX (Freed and Remington 2000), CPM-GOMS (Gray, John, and Atwood 1993), and NGOMSGL (Kieras 1997).

Soar

Soar has roots in cognitive psychology and computer science, but it is primarily a functional approach to encoding intelligent behavior (Laird, Newell, and Rosenbloom 1987). The continuing thread in Soar research has been to find a minimal but sufficient set of mechanisms for producing intelligent behavior. These goals have resulted in uniform representations of beliefs and knowledge, fixed mechanisms for learning and intention selection, and methods for integrating and interleaving all reasoning.

Like BDI, Soar's principles are based in part on assumed high-level constraints on intelligent behavior. Foremost among these are the problem space hypothesis (Newell 1982) and the physical symbol systems hypothesis (Newell 1980). Problem spaces modularize long-term knowledge so that it can be brought to bear in a goal-directed series of discrete steps (on the surface, this modularization is somewhat similar to the encapsulation of actions provided by FSMs, described later). The problem space hypothesis assumes rationality, similar to BDI. The physical symbol systems hypothesis argues that any entity that exhibits intelligence can be viewed as the physical realization of a formal symbol-processing system. The physical symbol systems hypothesis led to Soar's commitment to uniform representations of knowledge and beliefs.

There is no explicit assumption of hierarchical task representations in Soar (as there is in GOMS), but in practice the use of problem spaces often leads to the development of hierarchically organized behavior models, in which each portion of the hierarchy may represent a different problem space. However, the general notion of problem spaces also supports other types of goal arrangements and context switching.

While Soar shares with BDI the notion of agent rationality (agents appropriately select actions in pursuit of goals) and Soar uses logic-based knowledge representation, Soar does not share BDI's commitment to logical reasoning to produce rational behavior. Thus, Soar imposes strong constraints on fundamental aspects of intelligence, but it does not impose functionally inspired high-level constraints (in the spirit of BDI's use of logic, or GOMS's use of hierarchical goal decomposition). Soar is a lower-level framework for reasoning than BDI and GOMS. Either BDI principles (Wray and Jones 2005) or GOMS principles (Peck and John 1992) can be followed when using Soar as the implementation architecture.

FSMs

FSM (finite state machine) approaches to intelligent agents come from theoretical computer science (Carmel and Markovitch 1996; Hopcroft and Ullman 1979). Their primary appeal is the simplicity of their representational elements, which can usually be easily understood and encoded, automatically learned for some tasks, and implemented very efficiently. However, it is worth noting that some of these advantages diminish for models of large size or high complexity. Because states and transitions are relatively simple and low level, FSMs do not present the same types of high-level architectural constraints as the other frameworks we review here. Rather they provide theoretically sound elements from which more complex systems can be built. FSMs achieve complexity in behavior by the complex design of states and transitions. Because of the relatively simple level of support for knowledge representation idioms, some would argue that FSMs do not represent a knowledge-intensive agent framework at all. However, FSMs are used for a variety of agent applications, especially in computer games and human behavior representation (Ceranowicz, Nielsen, and Koss, 2000), so it is worth considering this approach in our analysis.

In the purest form of FSM, the only representational commitment is the state itself, which uniquely represents some point in the space of all possible combinations of beliefs and goals. While this commitment may seem minimal in comparison to the other frameworks, in practice FSMs provide additional ways to implement many of the constructs shared by other agent frameworks. For example, FSMs are functionally equivalent to a set of propositional stimulus-response rules, in which the state uniquely determines an agent's action, given its knowledge base. In practice, however, it is just as difficult to build a knowledge-intensive system using pure FSMs as it would be to use a purely propositional set of rules. Thus, practical FSM systems extend the approach by, for example, supporting variables within and across states, allowing conditional execution, and in some cases providing a global memory store.

Like BDI and GOMS, FSMs provide a general framework that has been implemented in a wide variety of systems. Because FSMs can be easily implemented within standard procedural programming languages, they are often equipped with additional features that violate the strict FSM paradigm. For example, FSMs can be hierarchically combined to allow multiple goals and task decompositions. In such an implementation, entering a state in one ma-

	Representation	Commitment	Reconsideration
Inputs			
BDI	Input language		
GOMS	Input language		
Soar	Working memory		
FSM	State transitions		
Justified Beliefs			
BDI	Beliefs	Logical inference	Belief revision
GOMS	Working memory		
Soar	Working memory	Match/assert	Reason maintenance
FSM	State variables		
Assumptions			
BDI	Beliefs	Plan language	Plan language
GOMS	Working memory	Operators	Operators
Soar	Working memory	Deliberation/Ops	Operators
FSM	State variables	Assignment	Assignment
Desires			
BDI	Desires	Logic	Logic
GOMS			
Soar	Proposed ops.	Preferences	Preferences
FSM			
Active Goals			
BDI	Intentions	Deliberation	Decision theory
GOMS	Goals	Operators	
Soar	Beliefs/Impasses	Deliberation	Reason maintenance
FSM	State machine	Context switching	Context switching
Plans			
BDI	Plans	Plan selection	Soundness
GOMS	Methods	Selection	
Soar			Interleaving
FSM	Transition networks	Context switching	
Actions			
BDI	Plan language	Atomic actions	
GOMS	Operators	Operators	
Soar	Primitive Ops	Deliberation	Reason maintenance
FSM	State transitions	Serial control flow	
Outputs			
BDI	Plan language	Plan language	
GOMS	Primitive ops.	Conditional ops.	
Soar	Working memory	Conditional ops.	
FSM	Output transitions	Serial control flow	

Table 1. Agent Framework Comparisons.

Black items are specific solutions provided by the framework. Gray items are general support provided by the framework. No entry means the framework does not explicitly address the element.

chine causes a jump into the initial state of another machine (with a subsequent jump back when the second machine completes execution).

Analysis of Agent Frameworks

Each of these frameworks provides a coherent view of agency and gives explicit attention to specific representations and processes for intelligent agents. They also reflect different points of emphasis, arising in part from the theoretical traditions that produced them. However, because none of the frameworks cover all the points of emphasis, agent designers have to make many more decisions about agent construction than provided by each framework's core principles. Each architectural implementation requires nonprimitive representations (and the processes to manipulate these representations). While it is likely that an agent will have compositional representations (for example, a representation of a map composed of an indexed set of beliefs), general representational constructs that span most applications should be directly supported within the framework.

Direct support simplifies the development process because the agent designer can concentrate exclusively on the domain knowledge. The practical point of an agent framework is to provide a set of reusable elements in order to reduce the costs of building new agents. One could therefore argue that, to maximize reuse, any representational element that is general across domains and useful in the majority of agent applications should be required to be addressed in each framework. However, this desire for reusability must be taken in context with the additional functional and theoretical constraints associated with each framework. For our analysis, we will comprehensively list each representational element supported by any of the frameworks, and note where individual frameworks provide support (or not) for those elements.

Table 1 lists the union of the base-level representations from BDI, GOMS, Soar, and FSMs. The representations are ordered to suggest the basic information flow from an external world into agent reasoning and then back out. The "Representation" column specifies each framework's substrate for the base-level representational element. Each representation also requires a decision point in the reasoning cycle, where an agent must choose from a set of alternatives. We generalize Wooldridge's (2000) concept of intention commitment to specify the process an agent uses to assert some instance of the base-level representation. In Table 1, the "Commitment" column identifies the general process used to select among alternatives. Most commitments also require maintenance; the "Reconsideration" column shows the determination of whether a commitment remains valid (a generalization of the notion of intention reconsideration [Schutt and Wooldridge 2001]).

Perceptions

Any interactive agent must have a perceptual or input system that provides a primitive representation of the agent's environment or situation. Neither BDI nor GOMS specifies any particular constraints on input. FSMs generally specify input conditions for initial states, as well as for state transitions. These conditions usually correspond to percepts in the environment, but the FSM approach makes no commitment to the specific representation of these conditions or to their grain size. Soar represents primitive perceptual elements in the same attribute-value representation as beliefs, although it does not dictate the structure of the perceptual systems that create these elements. However, the constraint that perceptual input must be represented in the same language as beliefs has important implications. Aside from their location in memory, primitive perceptual representations are indistinguishable from beliefs, which is consistent with Soar's principle of uniform knowledge representation. This makes it a relatively simple matter to allow agents to deliberate over potential input situations (or reflect on past or possible future input experiences) and transfer that knowledge directly to actual inputs.

Beliefs

From primitive perceptual elements, an agent creates a further elaborated set of beliefs, or interpretations of the environment. The set of beliefs is sometimes referred to as the *current state* of the agent, a concept that is made explicit in FSMs. Using the terminology of reason maintenance systems (Forbus and deKleer 1993), beliefs can be classified as either justified beliefs or assumptions. Justified beliefs remain in memory only as long as they are logically entailed by perceptual representations and assumptions. Assumptions, by definition, remain in memory independently of their continuing relevance to—and logical consistency with—the external environment. Assumptions remain asserted until the agent explicitly removes them, with the result that assumptions receive a high degree of commitment from the agent. Assumptions are necessary because not all beliefs can be grounded in current perception. For exam-

ple, if an agent needs to remember an object no longer in the field of view, then it must commit to maintaining a memory of that object. As long as the object remains in the field of view, the agent's perception of the object can be considered a *justified belief* (sometimes called an *entailment*). As soon as the perceptual grounding disappears, the agent must commit to the belief as an *assumption* if it is going to maintain the belief for some time.

Neither GOMS nor BDI makes an explicit distinction between justified beliefs and assumptions. Each provides general mechanisms for maintaining justified beliefs, but not specific solutions. Belief revision (Gardenfors 1988) is the mechanism of justified belief reconsideration in the BDI framework, although details of the process are only defined in various specific implementations. Soar uses a reason maintenance system to assert and retract justified beliefs automatically. Reason maintenance ensures that justified beliefs are logically consistent (Wray and Laird 2003). All four frameworks support the representation of assumptions. Soar requires that assumptions be created as the result of deliberate commitments (operator effects).

Pure FSMs would only be allowed to represent combinations of beliefs with individual states, because they are prohibited from maintaining internal state information. However, this would lead to an enormous and unmanageably complex set of states. Probably for this reason, we are not aware of any practical agents that are implemented using pure FSMs. Rather, the machines are generally augmented with variables that can hold various types of "nonstate" information. Variable values represent assumptions, because no primitive process maintains the continuing validity of a value with respect to the external situation.

Importantly, other frameworks use still other techniques for managing the commitment and reconsideration of beliefs. For example, 4D/RCS (Albus 2001) uses a limited capacity buffer, allowing only a fixed number of assumptions to be asserted at any one time. ACT-R (Anderson and Lebiere 1998) employs subsymbolic activation and decay mechanisms to manage assertions. By making such design decisions explicit in this analysis, we hope to facilitate a discussion of the trade-offs in these decisions among different approaches, and to make it more clear how to incorporate mechanisms from one architecture to another. For example, the activation and decay mechanisms of ACT-R have recently been incorporated into a hybrid architecture integrating Elements of ACT-R, Soar, and EPIC (EASE) (Chong and Wray 2005). EASE uses Soar's reason maintenance system to manage the assertion and retraction of justified beliefs, but uses ACTR's activation and decay mechanisms to manage assumptions. These alternative belief representations do not follow strict logical entailment, but also do not require deliberate agent reconsideration, so it is likely that we should include other types of beliefs as our analysis progresses. One of the contributions of this work is to provide a formal theoretical framework in which such variations in belief commitment and reconsideration can be labeled and characterized.

Desires

BDI is the only framework that clearly separates desires from "normal" active goals (below). Desires allow an agent to monitor goals that it has chosen not to pursue explicitly. An additional advantage is that an agent can communicate its desires to another agent that may be able to achieve them (Wooldridge 2000). Even in single-agent applications, however, there may be situations where an agent would need to reason about a desire, even if it does not have the resources to pursue that desire. Such situations may provide the possibility of opportunistically achieving desires in the context of other active goals.

Unlike BDI, agents built within many other frameworks do not bother to represent goals that they do not intend to pursue. Soar, GOMS, and FSMs do not specify that desires should exist, how they should be represented, or how they should influence reasoning. In these agents, expressing a desire would consist of a deliberate act in the service of a communication goal. BDI manages commitment to desires through logical inference.

Active Goals

A hallmark of intelligent behavior is the ability to commit to a particular set of concerns and then pursue them (Bratman 1987; Newell 1990). Most agent frameworks support explicit representation of the goals an agent has committed to pursue. However, the agent literature is somewhat inconsistent in its use of descriptive terms relevant to goals, which is a continuing source of confusion and miscommunication in the research community. Wooldridge (2000) calls *active goals* "intentions." In contrast, some implementations of BDI do not represent active goals distinctly from the selected plans that would achieve these goals. In such systems, *selected plans* are "intentions," but there is no explicit representation of an active goal apart from the plan. In Soar, an "intention" is the *next action selected* from a current

plan (which may itself directly activate a goal). GOMS does not use the term "intention," but requires the explicit representation of goals. In an attempt to avoid confusion, we call these commitments "active goals" to distinguish them from plans and ("inactive") desires. We also avoid altogether the ambiguous and overloaded term "intention."

Agents require a process for selecting the current active goal (or set of goals). BDI and Soar include explicit processes for deliberate goal commitment, although goals can be implemented in a variety of ways in Soar. In particular, goals created directly by the Soar architecture are limited to impasse goals; that is, goals to solve a particular problem in execution. While some approaches to Soar map task goals (for example, "intercept the aircraft") to impasse goals, this approach is only one of a number of "idioms" that are used to represent goals within Soar models (Lallement and John 1998). In GOMS, goal commitment occurs by invoking the plan associated with the goal. Although this is a deliberative process, it is not divided into separate steps as in the other frameworks. FSMs do not have an explicit notion of active goals. Implicitly, each FSM represents a plan that is associated with a particular goal (or set of goals).

Researchers have also explored the question of when an agent should reconsider an active goal (for example, Veloso, Pollack, and Cox [1998]; Schutt and Wooldridge [2001]; Wray and Laird [2003]). The BDI framework uses evaluations of soundness to determine when an active goal should be reconsidered; that is, given the agent's beliefs, the plan provably achieves the active goal. More recently, BDI researchers have also explored decision-theoretic processes for intention reconsideration (Schutt and Wooldridge 2001). Soar utilizes reason maintenance, which is essentially an implementation of the soundness criterion. GOMS uses selection rules to commit to a goal, but does not explicitly address later reconsidering a goal. An FSM would normally mark one or more of its states as states that achieve some (implicit) goal, perhaps terminating an individual state machine when a goal is achieved (although this approach would be different for *maintenance* goals).

Plans

Once there is an active goal to pursue, the agent must commit to a plan of action. BDI and GOMS assume there is a plan library or some other method for generating plans outside the basic agent framework (GOMS includes the notion of methods but does not prescribe how methods are implemented). Generally, plan execution is implicit in FSMs, so there is no explicit representation of finding a plan to achieve the goal. Each individual state machine *is* a plan, using states and transitions to capture the execution of the plan in the service of some (usually implicit) goal.

Soar does not require that an agent have an explicit representation of a plan. More commonly, Soar agents associate individual actions directly with goals (plan knowledge is implicit in the execution knowledge), or interleave planning and execution as individual cognitive tasks. Either way, Soar assumes that planning is a deliberative task requiring the same machinery as any other agent activity and involving the same concerns of commitment and resource usage. However, as with any other unsupported base-level representation, Soar forces the agent developer to implement the planning algorithm and the representation of any plans. Alternatively, external planning tools can be used to generate plans that must then be converted into Soar's belief representation language or production rules.

GOMS and BDI do not specify plan languages, although their implementations do. Soar has nothing like the relatively rich GOMS and BDI plan languages, instead using its operators to implement simple types of commitment. The trade-off is that complex plans are easier for a developer to program in GOMS and BDI, but potentially easier to learn by a Soar agent (because of the simpler, uniform target language). Developers of GOMS and BDI implementations must make decisions about plan languages, leading to nonuniform solutions from one implementation to another. The "plan language" for an FSM is generally just the same computer language used to implement the FSM.

Plan commitment in BDI can be quite simple: plans can be selected through a lookup table indexed by goal (Wooldridge 2000) or implied completely by goal selection, as in JAM. In sharp contrast, GOMS treats the choice of which method to choose to pursue a goal as a major element of the framework. Because Soar does not have an architectural notion of a plan, there is no plan-specific commitment mechanism. Soar also does not make an explicit distinction between plan generation/selection and plan execution. Creating (or finding) a plan involves a series of context-sensitive decisions and operations, just as executing a plan does.

An agent can consider abandoning its current plan, even when it has chosen to remain committed to its current goal (Wooldridge

2000). The frameworks here do not provide strong advice on when such a commitment should be given up; BDI and Soar at least dictate that any plan should be executed one action at a time, allowing reconsideration of the plan after each step (although the two disagree on how complex a single action can be). Frameworks that use explicit plans may provide support for abandoning a plan reactively (BDI) or ignore this problem completely (GOMS). Soar, because it does not require explicit plans, implicitly supports plan reconsideration, because there is no separate commitment to a plan in the first place. Thus, in Soar, an agent commits to one action at a time rather than committing to a whole plan. This embodies a strong *least-commitment* approach to plan selection and execution in general. The trade-off is that Soar agents must include extra knowledge to remain committed to a particular course of action, and the implementation of this knowledge is up to individual agent developers.

Other approaches to plan maintenance include using *completable plans* (Gervasio and DeJong 1994) and allowing agents to switch back and forth between two or more plans in support of multiple, orthogonal goals. Completable plans are plans that specify behavior to some abstract (but relatively low) level, and then allow the abstractions to instantiate conditionally and reactively to changing environments during execution. Plan switching is clearly a requirement for knowledge-intensive agents in many complex domains, but none of the frameworks specify how switching must occur. For example, it is not clear that any current implementations of BDI or GOMS support resumption of a partially executed plan. Many Soar systems implement task switching, but they rely on extra knowledge coded by the agent developer. Such *reentrant* execution of plans appears to be an essential element of opportunistic intelligent behavior.

Actions

Regardless of whether a plan has been explicitly represented, an agent must eventually commit to some type of action relevant to its active goals. In frameworks with explicit plans, like BDI and GOMS, this involves following and executing the steps in the plan. Explicit actions in FSMs simply involve moving from one state to another (or in augmented FSMs, possibly executing some code that changes the belief set or issues output commands). At their core, all four frameworks support three general types of actions: execute an output command, update the belief set, or commit to a new goal (or desire). GOMS and Soar define operators as the atomic level of action, allowing commitment and reconsideration for each plan action. As an alternative, BDI and FSM systems generally provide a plan language that is a complete programming language. Such languages provide powerful and flexible means of plan implementation, but may leave them outside the commitment regime of the framework. BDI dictates that reconsideration ought to occur after each plan step, but does not tightly constrain how much processing may occur in a single step. This imposes a trade-off between ease of programming (BDI and FSMs) and taking advantage of the uniformity of the framework's built-in processes (GOMS and Soar).

Soar uses actions to create assumptions in the belief set (thus, assumptions can only be the result of deliberate decision making). Tying assumptions to actions is an important issue. Automated, logical reason maintenance is attractive, but, pragmatically, there are limited resources for updating an agent's beliefs. Ideally, a rational agent would compute all relevant entailments from any input. But in complex environments, this is not computationally feasible (for example, Hill [1999]).

Regardless of the particular approach to plan representation or action languages, all the agent frameworks represent an action as a discrete step in a current plan's pursuit of a goal. If it happens to be a discrete step in an abstract plan, then it may get further decomposed (completable planning). In addition, each framework generally initiates a discrete action every "tick of the clock." This is how agents make progress towards their goals, and it allows a commitment scheme where reconsideration (of plans, goals, or beliefs, depending on the agent) can occur after each discrete action.

Outputs

The ultimate level of commitment is to initiate activity in the environment. To accomplish this, an agent invokes an output system. All four frameworks assume that output has to happen somehow, but do not impose strong constraints on the representation of output. BDI leaves output decisions up to the designer of the plan language. GOMS requires that primitive operators produce all output signals. As with perception, Soar requires that a motor command be represented in Soar's belief language, which allows the agent to reason about and execute output commands using the same agent knowledge.

Systems that use completable plans may include conditional outputs (possibly in addition to other conditional actions). Soar conditionally decodes actions using the same computa-

tional processes that it uses for justified belief maintenance. The instantiated completion of an action is analogous to the automated elaboration of beliefs. Each framework supports methods for executing completable plans; some depending on plan language choices. Soar specifies what the plan language has to be, and therefore also specifies how plan completion occurs.

Processing Requirements

Unsurprisingly, each data element that appears in an agent also requires associated processing that the agent framework uses to activate (or select) and deactivate (or retract) that type of representation. To remain consistent with our goal of unifying the discussion across frameworks, we have examined each representational element in terms of how the framework manages *commitment to* and *reconsideration of* an associated data structure. Generalizing the notion of commitment and reconsideration across representational elements allows us to adopt a similar abstract-level view of processing for each framework, but focus on the aspects of processing in which each framework differs.

Justified Belief Maintenance

Justified beliefs receive no commitment from an agent beyond logical entailment. A set of justified beliefs must always be logically consistent with the elements from which the beliefs are deduced, in so far as the long-term logical rules that produce the beliefs are logically sound. In the BDI framework, this is where rational logic plays a key role. Soar realizes the encoding process with a reason maintenance system that automatically computes entailments from ground perceptions, assumptions, and previously computed entailments. Because reason maintenance is essentially a computational implementation of logic, it would make sense for BDI agents to use a similar implementation. As mentioned previously, augmented FSMs and GOMS may use variables to maintain and store justified beliefs, but the generic frameworks do not specify any particular approach or algorithm.

Assumption Maintenance

In contrast to justified beliefs, assumptions receive a very high level of commitment from agents. An assumption essentially remains in the belief set until the agent explicitly decides to remove it (or, in hierarchical representations, until the agent achieves or gives up the goals associated with the assumption). In BDI terms, this requires explicit intentions both to create and to remove the assumption. Because it would be dangerous to create assumptions without some consideration, Soar demands that assumptions only be created as the result of deliberate intentions (whereas justified beliefs can be created by a more automatic process).

As this analysis demonstrates, the choice between justified beliefs and assumptions essentially boils down to the type of commitment or reconsideration necessary to activate or deactivate the beliefs. For rational agents, it may make sense to use justified beliefs as much as possible, in order to maintain a logically consistent belief set. This implies the use of a reason maintenance system, as Soar includes. Although BDI researchers have taken the question of commitment very seriously, they have mostly done so in terms of committing to (what they call) intentions and not to beliefs. The most popular BDI implementations do not include reason maintenance systems, even though logic and soundness are key parts of the BDI framework. GOMS and FSMs do not give the same prominent role to logic, and their implementations also do not generally include reason maintenance. In implementations that do not include reason maintenance, *all* beliefs are implemented as assumptions, and it is up to the agent developer to implement belief maintenance in the agent's knowledge base (for example, as part of each plan in a JAM agent).

Desire Maintenance

If an agent includes explicit representations of desires, there also needs to be a mechanism for maintaining desires. BDI again accomplishes this through a logical process. Presumably, modifications to support desires in the other frameworks could also mirror the processes that support belief maintenance. Soar includes a preference mechanism that allows the explicit proposal of actions that may not get selected, depending on the current context. One potential use for the preference mechanism would be to represent BDI-like desires, but that is not necessarily how the mechanism is used in practice. Similarly, desires could be represented and processed in GOMS or FSMs, but they would have to be maintained using specific knowledge encoded in the framework's action language.

Active Goal Maintenance

Active goals also require a mechanism of selection. In many frameworks, goal maintenance is similar to belief maintenance or arises as a side effect of the selection and execution of plans. For example, in GOMS goal activation occurs

by invoking the plan associated with a particular goal. Although this is a deliberative process, it is not divided into separate steps as in other frameworks. BDI and Soar include explicit processes for deliberate goal activation, distinct from the step of selecting a plan. FSMs, because they do not have explicit goals, also generally only have implicitly activated goals. An FSM goal may be considered active when the machine that implements that goal is executing. Therefore, we might say that an FSM activates a new goal by switching execution to a new plan (state machine).

Plan Maintenance

Tied closely to goal maintenance are questions of when an agent should select or abandon a current plan in the context of its current goals. Plan selection in most frameworks is fairly simple, with plans being indexed directly by goals. Under this model, activating a goal leads more or less directly to activating the associated plan. GOMS includes selection rules that allow the agent to consider different possible plans for a particular goal. As we have mentioned, Soar agents do not generally use explicit plan schemas, so it makes less sense to speak of plan selection for Soar. Rather, a "plan" in Soar is an emergent sequence of action selections.

For most of these frameworks the question of plan reconsideration is more interesting that the initial commitment to a plan. The question is when an agent should decide to abandon (or suspend) a plan, possibly even when it has chosen to remain committed to the plan's current goal (Wooldridge 2000). None of the frameworks provide clear advice on when such reconsideration should occur, but this is appropriate because it is a decision that requires knowledge of a particular task. The frameworks that use explicit plans in general must consider whether to provide support for the ability to abandon a plan reactively. The frameworks that do not directly support or insist on explicit plans implicitly support easier plan switching, because there is no separate commitment to a plan in the first place. Thus, in frameworks that do not require explicit plan representations, an agent may commit to one action at a time rather than committing to a whole plan (it is sometimes useful in such situations to view each action as a very fine-grained plan, or each plan as a very complex completable action). The BDI framework lies between the two extremes. While BDI insists that a plan must exist and be selected explicitly, it also dictates that plans should execute one action at a time, allowing time for reconsideration after each step. Some variations of GOMS and FSMs provide similar functions.

The issue of commitment and reconsideration associated with plans has many ramifications for the design and capabilities of intelligent agents. If an agent uses completable plans, it is possible to commit to an abstract plan while allowing adaptations of the plan during execution. Each framework supports methods for executing completable plans, some depending on designer choices of a plan language. Soar specifies what the plan language has to be, and therefore also specifies how plan completion occurs. Agents that do not commit to high-level plans do not have to provide mechanisms for switching plans midstream. But they do have to include mechanisms for remaining committed to a particular course of action when necessary (without the benefit of support from an explicit plan).

Perhaps the most complex form of plan interruption involves switching back and forth between two or more plans in support of multiple goals (perhaps also meaning that the system is switching activation between the goals). This is clearly a capability of humans, and we mark it as a requirement for knowledge-intensive agents in many complex domains. The ability to accomplish task switching depends on the commitment and reconsideration methods for plan selection and execution but adds the problem of recommitting to a suspended plan. Neither BDI nor FSMs explicitly specify how such switching might occur. A model builder would have to encode a number of explicit conditions for when the plan should be abandoned and then taken up again. Additionally, it is not clear how a BDI agent should represent two goals that are actively being pursued, but in a switched manner. Frameworks that do not define these types of commitment and reconsideration leave the choices up to individual agent designers.

Related to plan switching is reentrant execution. It is sometimes desirable for an agent to resume a plan from a suspended point, rather than beginning the plan anew. Similarly, it can be advantageous to commit to portions of plans opportunistically when it appears that part of a plan is suddenly appropriate to a set of goals. Implementations of BDI or FSMs do not appear to support initiating the execution of a plan from somewhere in the middle of it. As suggested earlier, a possible alternative would be not to provide support for monolithic plans, as in Soar, and essentially treat each plan action atomically. Under such a scheme, instead of having explicit plan representations, each action must have appropriate selection conditions based on the belief set and active goals.

Many Soar agents implement task switching, but it is particularly difficult for GOMS and FSMs, since those frameworks explicitly insist on a hierarchical task structure. Switching between tasks in different parts of a task hierarchy requires quite a bit of overhead in continuously constructing and replacing the active hierarchies. In addition, an agent that is switching between two plans for two different goals must make sure that each plan's completions are sensitive to the effects of the other plan. This requirement demands some sort of shared memory in a global store or blackboard. The belief sets in BDI, GOMS, and Soar serve this purpose and enable communication between switching plans. However, even FSMs that allow variables and hierarchical decomposition generally encapsulate the variables within each machine, making such task switching onerous.

Plan Execution

A final issue involves how each framework constrains execution of a selected plan. Plan execution might also be called *action maintenance* because it has to do with the commitment to and reconsideration of the individual actions that make up the plan. The main issue here has to do with whether the process of execution integrates into the overall decision mechanisms of the framework. GOMS and Soar both represent plan actions as operators, which serve the dual purpose of executing primitive actions and activating new goals. Thus, they integrate the basic processes of reasoning and commitment into each execution step of a plan. BDI and FSM systems generally provide a plan language that is a complete programming language, relatively disconnected from the basic processes provided by the framework.

Certainly a plan language should contain methods for updating beliefs. Some implementations also include language primitives for creating desires and activating goals. Other features, such as loops, conditionals, and possibly local variables, provide very powerful execution abilities, but leave them outside of the constraints of the framework. This again imposes a trade-off between ease of programming (where the BDI and FSM implementations generally win) and taking advantage of the uniformity of the framework's built-in processes (where GOMS and Soar generally have an advantage).

Conclusions

The research communities that use agent frameworks continue to explore the issues that limit and inform the development of highly competent intelligent agents *within* their frameworks (for example, Harland and Winikoff [2001]; Jones and Laird [1997]). However, too little attention is being paid to understanding the commonalities and differences across frameworks. Achieving this understanding is exacerbated by the differences in terminology and assumptions across research communities. We have attempted to contribute to this larger discussion by reviewing the directly supported representations and processes in broadly differing agent frameworks and adopting a rational, neutral, and common set of terms to describe each of them.

Each of these frameworks has been applied successfully to enough problems that it is not likely they are "missing" any representations and processes that are functionally necessary. However, from the point of view of creating deployed knowledge-intensive agents, the lack of explicit support from a framework imposes a burden on agent developers. Soar's (lack of) support for plans is a good example. The lack of an explicit plan representation lends flexibility in terms of plan execution (including interleaved execution with other plans). However, it also requires that a model builder create ad hoc solutions to plan commitment in the design of agent knowledge. Clearly this imposes a trade-off on the costs and benefits of using Soar's approach to plan representation. Each framework we have reviewed incorporates similar trade-offs with respect to various aspects of the framework's design.

Continuing to identify and develop representations and processes for agents is an important research activity. Increasingly, researchers are attending to processes necessary for social agents, including normatives, values, obligations, and teamwork. However, there are additional intraagent representations and processes that the frameworks discussed here do not directly support and that may be so widely necessary that they should be considered base-level representations. Examples include deliberate attention (Hill 1999), parallel active goals (Jones et al. 1994; Thangarajah, Padgham, and Harland 2002), and architectural support for managing resource limitations and conflicts (Meyer and Kieras 1997). Learning is also important for long-lived knowledge-intensive agents. The migration of knowledge into (and out of) long-term memory can also be studied in terms of representations, commitment, and reconsideration, resulting in a complex space of potential learning mechanisms (for example, along dimensions of automatic versus deliberate learning, or representations of procedural, declarative, and episodic memories).

This analysis lays the groundwork for extending and unifying the basic level representations and processes needed for knowledge-intensive intelligent agents. Perhaps as important, the analysis provides a potential theoretical framework and common set of terms to fuel future comparative and investigative work in the design of knowledge-intensive agent architectures.

References

Albus, J. 2001. *Engineering of Mind: An Introduction to the Science of Intelligent Systems*. New York: John Wiley and Sons.

Anderson, J. R.; and Lebiere, C. 1998. *Atomic Components of Thought*. Hillsdale, NJ: Lawrence Erlbaum.

Bratman, M. E. 1987. *Intentions, Plans, and Practical Reason*. Cambridge, MA: Harvard University Press

Bratman, M. E.; Israel, D. J.; and Pollack, M. E. 1988. Plans and Resource-Bounded Practical Reasoning. *Computational Intelligence* 4: 349–355.

Card, S.; Moran, T.; and Newell, A. 1983. *The Psychology of Human-Computer Interaction*. Hillsdale, NJ: Lawrence Erlbaum.

Carmel, D.; and Markovitch, S. 1996. Learning Models of Intelligent Agents. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*. Menlo Park, CA: AAAI Press.

Ceranowicz, A., Nielsen, P.; and Koss, F. V. 2000. Behavioral Representation In JSAF. In *Proceedings of the Ninth Computer Generated Forces and Behavioral Representation Conference*. Orlando, Fl: Institute for Simulation and Training, University of Central Florida.

Chong, R. S.; and Wray, R. E. 2005. Constraints on Architectural Models: Elements of Act-R, Soar and Epic In Human Learning and Performance. In *Modeling Human Behavior with Integrated Cognitive Architectures: Comparison, Evaluation, and Validation*, ed. K. Gluck and R. Pew, 237–304. Mahweh, NJ: Lawrence Erlbaum Associates.

d'Inverno, M.; Kinny, D.; Luck, M.; and Wooldridge, M. 1997. A Formal Specification of dMARS. In *Intelligent Agents IV* Lecture Notes on Artificial Intelligence, Volume 1365, ed. M. P. Singh, A. Rao, and M. J. Wooldridge, 155–176. Berlin: Springer Verlag.

Forbus, K. D.; and deKleer, J. 1993. *Building Problem Solvers*. Cambridge, MA: The MIT Press.

Freed, M. A.; and Remington, R. W. 2000. Making Human-Machine System Simulation a Practical Engineering Tool: An Apex Overview. Paper presented at the 2000 International Conference on Cognitive Modeling, Groningen, the Netherlands, 23–25 March.

Gardenfors, P. 1988. *Knowledge In Flux*. Cambridge, MA: The MIT Press.

Georgeff, M. P.; and Ingrand, F. F. 1990. Real-Time Reasoning: The Monitoring and Control of Spacecraft Systems. In *Proceedings of the Sixth IEEE Conference on Artificial Intelligence Applications*. Piscataway, NJ: Institute of Electrical and Electronics Engineers.

Georgeff, M. P.; and Lansky, A. L. 1987. Reactive Reasoning and Planning. In *Proceedings of the Sixth National Conference on Artificial Intelligence*, 677–682. Menlo Park, CA: AAAI Press.

Gervasio, M. T.; and DeJong, G. F. 1994. An Incremental Approach for Completable Planning. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, 78–86. Menlo Park, CA: AAAI Press.

Gray, W. D.; John, B. E.; and Atwood, M. E. 1993. Project Ernestine: Validating a GOMS Analysis for Predicting and Explaining Real-World Performance. *Human-Computer Interaction* 8(3): 237–309.

Harland, J.; and Winikoff, M. 2001. Agents Via Mixed-Mode Computation In Linear Logic: A Proposal. Paper presented at the ICLP'01 Workshop on Computational Logic in Multi-Agent Systems. Paphos, Cyprus.

Hill, R. 1999. Modeling Perceptual Attention In Virtual Humans. Paper presented at the Eighth Conference on Computer Generated Forces and Behavior Representation. Orlando, FL.

Hopcroft, J. E.; and Ullman, J. D. 1979. *Introduction to Automata Theory, Languages, and Computation*. Reading, MA: Addison-Wesley.

Howden, N.; Rönnquist, R.; Hodgson, A.; and Lucas, A. 2001. Jack: Summary of an Agent Infrastructure. Paper presented at the Workshop on Infrastructure for Agents, MAS, and Scalable MAS at the Fifth International Conference on Autonomous Agents. Montreal, Canada.

Huber, M. J. 1999. Jam: A BDI-Theoretic Mobile Agent Architecture. In *Proceedings of the Third International Conference on Autonomous Agents*, 236–243. New York: Association for Computing Machinery.

John, B. E.; Vera, A. H.; and Newell, A. 1994. Toward Real-Time GOMS: A Model of Expert Behavior in a Highly Interactive Task. *Behavior and Information Technology* 13(4): 255–267.

Jones, R. M.; and Laird, J. E. 1997. Constraints on the Design of a High-Level Model of Cognition. In *Proceedings of the Nineteenth Annual Conference of the Cognitive Science Society*. Wheat Ridge, CO: Cognitive Science Society.

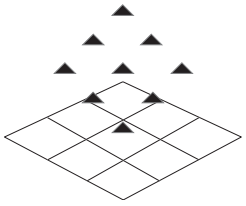
Jones, R. M.; Laird, J. E.; Nielsen, P. E. 1999. Automated Intelligent Pilots for Combat Flight Simulation. *AI Magazine* 20(1): 27–42.

Jones, R. M.; Laird, J. E.; Tambe, M.; and Rosenbloom, P. S. 1994. Generating Behavior in Response to Interacting Goals. In *Proceedings of the Fourth Conference on Computer Generated Forces and Behavior Representation*, 325–332. Orlando, Fl: Institute for Simulation and Training, University of Central Florida.

Kieras, D. E. 1997. A Guide to GOMS Model Usability Evaluation Using Ngomsl. In *Handbook of Human-Computer Interaction*, ed. M. Helander, T. Landauer, and P. Prabhu, 733–766. Amsterdam: North-Holland.

Kieras, D. E.; Wood, S. D.; Abotel, K.; and Hornof, A. 1995. GLEAN: A Computer-Based Tool for Rapid GOMS Model Usability Evaluation of User Interface Designs. In *Proceedings of the ACM Symposium on User Interface Software and Technology*. New York: Association for Computing Machinery.

Laird, J. E.; Newell, A.; and Rosenbloom, P. S. 1987. Soar: An Architecture for General Intelligence. *Artifi-*



AAAI 2007 Spring Symposium Series

The 2007 Spring Symposium Series will be held March 26 through March 38, 2007 at Stanford University. The Call for Participation will be available in July on the AAAI web site (www.aaai.org/Symposia/Spring/sss07.php).

Submissions will be due to the organizers on October 6, 2006.

For more information, please contact Symposium Chair, Alan Schultz, at schultz@aic.nrl.navy.mil or AAAI at sss07@aaai.org.

cial Intelligence 33(1): 1–64.

Lallement, Y.; and John, B. E. 1998. Cognitive Architecture and Modeling Idiom: A Model of the Wicken's Task. In *Proceedings of the Twentieth Annual Conference of the Cognitive Science Society*. Wheat Ridge, CO: Cognitive Science Society.

Langley, P.; Choi, D.; and Shapiro, D. 2004. A Cognitive Architecture for Physical Agents. Technical Report. Institute for the Study of Learning and Expertise, Palo Alto, CA.

Meyer, D.; and Kieras, D. 1997. Epic: A Computational Theory of Executive Cognitive Processes and Multiple-Task Performance: Part 1. *Psychological Review* 104: 3–65,

Newell, A. 1980. Physical Symbol Systems. *Cognitive Science* 4: 135–183.

Newell, A. 1982. The Knowledge Level. *Artificial Intelligence* 18(1): 87–127.

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

Payne, T. R.; Singh, R.; and Sycara, K. 2002. Calendar Agents on the Semantic Web. *IEEE Intelligent Systems* 17(3): 84–86,

Peck, V. A.; and John, B. E. 1992. Browser-Soar: A Computational Model of a Highly Interactive Task. In *Proceedings, SIGChi Conference on Human Factors in Computing Systems*, ed. P. Bauersfeld, J. Bennett, and G. Lynch,165–172. New York: ACM Press.

Rao, A.; and Georgeff, M. 1995, BDI Agents: From Theory to Practice. In *Proceedings of the First International Conference on Multiagent Systems*. San Francisco. Menlo Park, CA: AAAI Press.

Russell S.; and Norvig, P. 1994. *Artificial Intelligence: A Modern Approach*. Upper Saddle River, NJ: Prentice Hall. 1995

Schutt, M. C.; and Wooldridge, M. 2001. Principles of Intention Reconsideration. *Proceedings of the Fifth International Conference on Autonomous Agents*, 209–216. New York: ACM Press.

Thangarajah, J.; Padgham, L.; and Harland, J. 2002. Representation and Reasoning for Goals in BDI Agents. In *Proceedings of the Twenty-Fifth Australasian Conference on Computer Science* — Volume 4, Melbourne. Darlinghurst, Australia: Australian Computer Society.

Veloso, M. M.; Pollack, M. E.; and Cox, M. T. 1998. Rationale-Based Monitoring for Planning in Dynamic Environments. In *Proceedings of the Fourth International Conference on AI Planning Systems*. Menlo Park, CA: AAAI Press.

Wooldridge, M. 2000. *Reasoning about Rational Agents*. Cambridge, MA: The MIT Press.

Wray, R. E.; and Jones, R. M. 2005. An Introduction to Soar as an Agent Architecture. In *Cognition and Multi-Agent Interaction: From Cognitive Modeling to Social Simulation*, ed. R. Sun. Cambridge, UK: Cambridge University Press.

Wray, R. E.; and Laird, J. E. 2003. An Architectural Approach to Consistency In Hierarchical Execution. *Journal of Artificial Intelligence Research* 19: 355–398.

Randolph M. Jones, Ph.D., is a senior scientist at Soar Technology, Inc., and an assistant professor of computer science at Colby College. He has worked with a variety of agent architectures and models in both scientific research and applied intelligent systems. He has more than 20 years of research and development experience with cognitive and agent architectures, intelligent agents, machine and human learning, graphical user interfaces, cognitive modeling, and a variety of related areas. Jones previously held research positions at the University of Michigan, the University of Pittsburgh, and Carnegie Mellon University. He earned a B.S. (1984) in mathematics and computer science at UCLA, and M.S. (1987) and Ph.D. (1989) degrees from the Department of Information and Computer Science at the University of California, Irvine. Contact rjones@soartech.com.

Robert E. Wray is chief scientist at Soar Technology. He received a Ph.D. in computer science and engineering from the University of Michigan. His doctoral research focused on maintaining logical consistency in agent reasoning systems, and his innovations were incorporated in the Soar architecture. At Soar Technology, he leads or has led R&D projects for the U.S. Air Force, Army, and Navy and the Defense Advanced Research Projects Agency. Wray's research encompasses many areas of artificial intelligence including agent-based systems and agent architectures, machine learning, cognitive science, and knowledge

Appears in 21st National Conference on Artificial Intelligence (AAAI06), Boston, MA 2006

Towards a Validated Model of "Emotional Intelligence"

Jonathan Gratch, Stacy Marsella and Wenji Mao

University of Southern California
gratch@ict.usc.edu, Marsella@isi.edu, mao@ict.usc.edu

Abstract

This article summarizes recent progress in developing a validated computational account of the cognitive antecedents and consequences of emotion. We describe the potential of this work to impact a variety of AI problem domains.

Introduction

The last decade has seen an explosion of interest in emotion in both the social and computational sciences. Within artificial intelligence, we see the development of computational models of emotion as a core research focus that will facilitate advances in the large array of intelligent systems that strive for human-level competence in dynamic, semi-predictable and social environments:

- Applications that presume the ability to interpret the beliefs, motives and intentions underlying human behavior can benefit from a model of how emotion motivates human action, distorts perception and inference, and communicates information about mental state. Some tutoring applications already incorporate emotion into user models [1]. Dialogue and collaborative planning systems could also benefit from such an approach.
- Emotions play a powerful role in social influence: emotional displays seem designed to elicit social responses from other individuals. Such responses can be difficult to suppress and the responding individual may not even be consciously aware of the manipulation. A better understanding of this phenomena would benefit applications that attempt to shape human behavior, such as psychotherapy [2], tutoring [3] and marketing.
- Modeling techniques increasingly strive to simulate emotional-evoking situations such as how crowds react in disasters [4], how military units respond to the stress of battle [5], and even large social situations as when modeling the economic impact of traumatic events such as 9/11 or modeling inter-group conflicts [6]).

More generally, an understanding of the cognitive and social function of human emotion complements traditional rational views of intelligence. Debates about the benefit of emotion span recorded history and were prominent in the early days of artificial intelligence. Several have argued that emotional influences that seem irrational on the surface have important social and cognitive functions that are lacking from the individualistic and disembodied view of cognition from which artificial intelligence stems. For example, Simon [7] argued that emotions make us more reactive by interrupting normal cognition when unattended goals require immediate servicing in the world. Frank argues social emotions such as anger reflect a mechanism that improves group utility by minimizing social conflicts, and thereby explains peoples "irrational" choices to cooperate in social games such as prison's dilemma [8]. Similarly, "emotional biases" such as wishful thinking may reflect a rational mechanism that is more accurately accounting for certain social costs, such as the cost of betrayal when a parent defends a child despite strong evidence of their guilt in a crime [9]. Finally, the exercise of accurately modeling emotion can often spur the development of new agent capabilities. For example, Mao's effort to model anger has led to a general mechanism of social credit assignment and a model of social coercion [10].

This article draws on recently published papers to summarize our recent progress in developing a validated computational account of the cognitive antecedents and consequences of emotion [10-13]. Our goal is to create working models that simulate emotional human behavior for a variety of possible applications, but with a focus on virtual reality-based training [14]. Here, we highlight recent progress in validating this model against human performance data, along the way emphasizing differences between "rational" and emotionally influenced information processing.

Emotion Theory

Contemporary emotion research suggests emotion exacts pervasive control over cognitive processes. Emotional state can influence what information is available in working memory [15], the subjective utility of alternative choices [16], and even the style of processing [17]. For example,

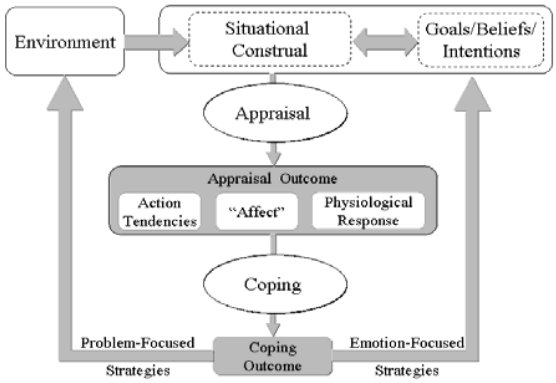


Figure 1: Schematic view of appraisal theory.

people who are angry or happy tend to perform shallower inference and are more influenced by stereotypical beliefs than sad individuals. Neuroscience evidence also underscores the close connection between emotion and centers of the brain associated with higher-level cognition. For example, damage to the connections between emotion and decision-making centers of the brain lead to maladaptive behavior in certain gambling tasks [18]. Collectively, these findings demonstrate that emotion and cognition are closely coupled and suggest emotion has a strong, pervasive and controlling influence over cognition.

There are several theoretical perspectives on the relationship between emotion and cognition. *Appraisal theory* [19] is the predominant psychological theory of emotion (Figure 1). We argue that it is the most fruitful theory of emotion for those interested in the design of symbolic AI systems as it emphasizes the connection between emotion and cognition. Emotion is argued to arise from patterns of individual judgment concerning the *person-environment relationship* (i.e., the perceived relationship between events and an individual's beliefs, desires and intentions). These judgments, formalized as *appraisal variables*, characterize aspects of the personal significance of events (e.g., was this event expected in terms of my prior beliefs? is this event congruent with my goals; do I have the power to alter the consequences of this event?). Patterns of appraisal elicit emotional behavior, but they also trigger stereotypical cognitive responses formalized as qualitatively distinct *coping strategies* (e.g., planning, procrastination or resignation).

Due to its reliance on cognitive judgments and responses, appraisal theory can be recast as a requirement specification for how to build an intelligent system – it claims a superset of the judgments and cognitive strategies considered by most AI systems must be supported in order to correctly detect, classify, and adaptively respond to significant changes to their physical and social environment.

EMA

EMA is a computational model of the cognitive antecedents and consequences of emotion as posited by appraisal theory [11, 13]. In general terms, we characterize a computational model as processes operating on representations. In this case, the processes involve the interpretation (appraisal) and manipulation (coping) of a representation of the person-environment relationship. In realizing this abstract psychological theory, we draw extensively on common artificial intelligent methods of reasoning and representation. To this end, EMA represents the relationship between events and an agent's internal beliefs desires and intentions by building on AI planning to represent the physical relationship between events and their consequences, and BDI frameworks to represent the epistemic factors that underlie human (particularly social) activities.

Appraisal processes characterize this representation in terms of individual appraisal judgments. These extend traditional AI concerns with utility and probability:

- Desirability: what is the utility (positive or negative) of the event if it comes to pass.
- Likelihood: how probable is the outcome of the event.
- Causal attribution: who deserves credit/blame.
- Controllability: can the outcome be altered by actions under control of the agent.
- Changeability: can the outcome change on its own.

Patterns of appraisal elicit emotional displays, but they also initiate coping processes to regulate the agent's cognitive response to the appraised emotion. Coping strategies work in the reverse direction of appraisal, identifying plans, beliefs, desires or intentions to maintain or alter. These include "problem focused" strategies (e.g. planning) directed towards improving the world (the traditional concern of AI techniques) but also encompasses "emotion-focused" strategies that impact an agent's epistemic and motivational state:

- Planning: form an intention to perform some act (the planner uses intentions to drive its plan generation)
- Seek instrumental support: ask someone that is in control of an outcome for help
- Procrastination: wait for an external event to change the current circumstances
- Denial: lower the perceived likelihood of an undesirable outcome
- Mental disengagement: lower utility of desired state
- Shift blame: shift responsibility for an action toward some other agent

Strategies give input to the cognitive processes that actually execute these directives. For example, planful coping generates an intention to act, leading the planning system to generate and execute a valid plan to accomplish this act. Alternatively, coping strategies might abandon the goal, lower the goal's importance, or re-assess who is to blame.

EMA uses an explicit representation of plans, beliefs, desires and intentions to capture output and intermediate results of processes that relate the agent to its physical and social environment. This represents the agent's current view of the agent-environment relationship, which changes

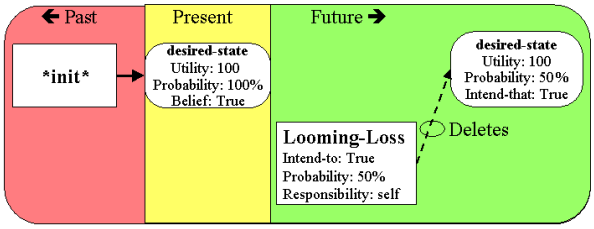


Figure 2: EMA's encoding of the SCPQ loss condition

with further observation or inference. We treat appraisal as a mapping from syntactic features of this representation to individual appraisal variables. Multiple appraisals are aggregated into an overall emotional state that influences behavior. Coping directs control signals to auxiliary reasoning modules (i.e., planning, or belief updates) to overturn or maintain features of the representation that lead to individual appraisals. For example, coping may abandon a cherished desire in response to an uncontrollable threat.

Validation

Our recent efforts have been directed towards validating EMA's effectiveness in modeling the influence of emotion over human judgments. This involves assessing its consistency with human emotional responses (I/O validity). We are further interested in the more challenging test of whether the inferential mechanisms underlying EMA are consistent with human inference (process validity). Rather than using an abstract overall assessment, such as a subject's assessment of "believability," we directly compare the internal variables of the model to human data, assessing emotional responses, but also the value of appraisal variables, coping tendencies, and in particular, how these assessments change in response to an evolving situation.

There is little established methodology for evaluating emotion models and our group spearheaded such efforts. Our current efforts adopt the following approach: identify a corpus of emotional situations used to validate psychological theories of emotion; encode these situations in our model; contrast the predictions of the model with human responses. We have recently completed two major studies based on this approach which we summarize here.

Appraisal and Coping Dynamics: Although human mental processes cannot be observed directly, emotion psychologists assess this information indirectly through interactive questionnaires. The Stress and Coping Process Questionnaire (SCPQ) [20] is one such instrument used to assess coping processes. Subjects are presented stereotypical emotion-evoking episodes and their responses are queried as the episodes evolve. Episodes are constructed from a grammar that encodes prototypical causal relationships between events and goals. For example, in the *loss condition*, a subject might be told of a looming threat to an important goal (e.g. your spouse is threatening a divorce). Subjects are queried on how they would feel in this situation (*emotional response*), how they appraise certain aspects of the current situation (*appraisal variables*) and what strategies they would use to confront the situation (*coping strategies*). They are then presented updates to the situation (e.g., some time has passed and the situation has not improved) and asked how their interpretation changes.

The grammar underlying SCPQ elicits specific patterns of appraisal and coping responses. We use this characteristic to assess the validity of EMA's by comparing these patterns with those produced by the model. Rather than attempting to parse English and use the scale directly, we take advantage of the fact that all of the episodes in the scale correspond to one of four dynamic causal theories. For example, Figure 2 illustrates EMA's encoding of the loss condition. See [12] for details.

Results: The results show strong support for the model. SCPQ identifies nine trends that indicate normal emotional responses. EMA is consistent with eight of these trends. EMA also shows close correspondence with the temporal patterns of appraisal and emotional response across the phases of the dynamic scenarios. One departure from the human data is that people often felt they had more control over situations than are predicted by the model, suggesting people were bringing commonsense knowledge to bear that was not explicitly mentioned in the episode descriptions. Another limitation of EMA concerns its ability to reason about social emotions. This is addressed in the next study.

Causal Attributions: Some emotions such as guilt and anger involve social judgments of blame and responsibility. Although many intelligent systems reason about the physical causes of outcomes, traditional notions of causality are simply inadequate for explaining such social judgments. Instead, social causality, in theory and as practiced in everyday folk judgments, emphasizes multiple causal dimensions, involves epistemic variables, and distinguishes between physical cause, responsibility and blame.

We have begun to model how people form judgments of blame and responsibility, including not only causal factors, but also epistemic variables such as freedom of choice, intention and foreknowledge [10, 21]. As a result, an actor may physically cause an event, but be absolved of responsibility and blame, or conversely, blamed for what she did not physically cause.

Using a variation of our methodology, we contrasted performance of this model against human performance data on hypothetical scenarios, but used the model itself to systematically generate scenario variants that should be appraised differently. As a starting point, we adopt the well-known "company program scenario that involves two corporate executives discussing a policy that may harm the environment [22]. In our study, descriptions of the scenario are organized into separate labeled statements of evidence. We then added, deleted or altered these lines in order to change intermediate inferences made by the model. For example, if our model suggests that a particular line of evidence is

necessary to infer coercion, than an obvious variation would be to eliminate that line of evidence.

Results: A questionnaire followed the presentation of each scenario. Each question was designed to test the belief about one judgment underlying the model (e.g., did agent A intend X). In terms of I/O validity, we measured the agreement of the model and each subject using *Kappa statistic*. The average *Kappa* agreement of the model and subjects is 0.732, indicating substantial agreement. In terms of process validity, we compared subject responses with intermediate inferences of the model. In the model, each belief is derived by a specific inference rule, so each question in the questionnaire corresponds to the firing of one rule. Currently, we have 37 dialogue and causal inference rules in the model. This survey study covers 19 of them. To assess the inference rules, we compare the conditions of each rule with the evidence people use in forming each answer. We derive an accuracy value for each rule based on a confusion matrix built from subjects responses. The results show high accuracy (in the range of 70-90% for each rule).

Conclusion

Our empirical studies to date show strong empirical support for our approach. One concern with our current validation methodology is its reliance on self-reports of imagined situations. Although this is standard in contemporary emotion research and results are generally consistent with those obtained by other means, self-reports are rightly criticized as possibly saying more about how people think about emotion retrospectively rather then how they actually behave in emotional situations. As self-reports are the primary means for assessing appraised emotional state, this is a concern, not just for the present study, but for the field of emotion research in general. The use of virtual humans and virtual environments points to one way to address this concern. Rather than presenting subjects a fixed textual description of a situation, they could be presented with a virtual facsimile of the episode. And rather than asking subject how they might act in such a situation, they could be provided the means of actually acting out in the episode and possibly changing its evolution through their actions.

More generally, we see the modeling emotion as increasingly vital as AI matures beyond simple, static and nonsocial problem solving. Human emotion clearly exacts a controlling influence over cognition and a functional analysis of emotion's impact can contribute to the discourse on how to achieve human-level intelligence. As a theory designed to characterize emotional responses to a wide span of human situations, appraisal theory can suggesting core cognitive functions often overlooked by traditional AI.

References

1. Conati, C. and H. MacLaren. *Evaluating A Probabilistic Model of Student Affect.* in *7th International Conference on Intelligent Tutoring Systems.* 2004. Maceio, Brazil.
2. Marsella, S., W.L. Johnson, and C. LaBore. *Interactive Pedagogical Drama.* in *Autonomous Agents.* 2001. Montreal.
3. Lester, J.C., B.A. Stone, and G.D. Stelling, *Lifelike Pedagogical Agents for Mixed-Initiative Problem Solving in Constructivist Learning Environments.* User Modeling and User-Adapted Instruction, 1999. **9**(1-2): p. 1-44.
4. Silverman, B.G., et al. *Constructing Virtual Asymmetric Opponents from Data and Models in the Literature: Case of Crowd Rioting.* in *CGF-BR.* 2002. Orlando, FL.
5. Gratch, J. and S. Marsella, *Fight the way you train: the role and limits of emotions in training for combat.* Brown Journal of World Affairs, 2003. **X(1)**(Summer/Fall): p. 63-76.
6. Marsella, S., D. Pynadath, and S. Read. *PsychSim: Agent-based modeling of social interactions and influence.* in *International Conference on Cognitive Modeling.* 2004.
7. Simon, H.A., *Motivational and emotional controls of cognition.* Psychological Review, 1967. **74**: p. 29-39.
8. Frank, R., *Passions with reason: the strategic role of the emotions.* 1988, New York, NY: W. W. Norton.
9. Mele, A.R., *Self-Deception Unmasked.* 2001, Princeton, NJ: Princeton University Press.
10. Mao, W. and J. Gratch. *Evaluating a computational model of social causality and responsibility.* in *AAMAS* 2006.
11. Gratch, J. and S. Marsella. *Tears and Fears: Modeling Emotions and Emotional Behaviors in Synthetic Agents.* in *Autonomous Agents.* 2001. Montreal, Canada: ACM Press.
12. Gratch, J. and S. Marsella. *Evaluating the modeling and use of emotion in virtual humans.* in *AAMAS* 2004. New York.
13. Marsella, S. and J. Gratch. *Modeling coping behaviors in virtual humans: Don't worry, be happy.* in *AAMAS* 2003..
14. Swartout, W., et al., *Toward Virtual Humans.* AI Mag, 2006.
15. Bower, G.H., *Emotional mood and memory.* American Psychologist, 1991. **31**: p. 129-148.
16. Lerner, J.S. and D. Keltner, *Beyond valence: Toward a model of emotion-specific influences on judgement and choice.* Cognition and Emotion, 2000. **14**: p. 473-493.
17. Bless, H., N. Schwarz, and M. Kemmelmeier, *Mood and stereotyping: The impact of moods on the use of general knowledge structures.* European Review of Social Psychology, 1996. **7**: p. 63-93.
18. Bechara, A., et al., *Different Contributions of the Human Amygdala and Ventromedial Prefrontal Cortex to Decision-Making.* Journal of Neuroscience, 1999. **19**(13).

This work was sponsored by the U. S. Army Research, Development, and Engineering Command (RDECOM), and the content does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

19. Scherer, K.R., A. Schorr, and T. Johnstone, eds. *Appraisal Processes in Emotion.* Affective Science, ed. R.J. Davidson, P. Ekman, and K.R. Scherer. 2001, Oxford University Press.
20. Perrez, M. and M. Reicherts, *Stress, Coping, and Health.* 1992, Seattle, WA: Hogrefe and Huber Publishers.
21. Mao, W. and J. Gratch. *Social Judgment in Multiagent Interactions.* in *AAMAS* 2004.
22. Knobe, J., *Intentional Action and Side-Effects in Ordinary Language.* Analysis, 2003. **63**: p. 190-193.

Intelligence Assessment for Early-Stage Software Systems Aimed at Human-Level, Roughly Human-Like AGI

Ben Goertzel
Novamente LLC

One of the many difficult issues arising in the course of research on human-level AGI is that of "evaluation and metrics" – i.e., AGI intelligence testing.

It's not so hard to tell when you've achieved human-level AGI -- though there is *some* subtlety here, which I'll discuss below. However, assessing the quality of incremental progress toward human-level AGI is a much subtler matter. In this essay I'll present some thoughts on this issue, culminating in a couple specific proposals:

- *Online School Tests*, in which AGIs are tested via their ability to succeed in existing online educational fora
- of more immediate interest, a series of tests called the *AGI Preschool Tests* (AIP Tests[1], for short), based on the notion of "multiple intelligences" (Gardner, 1983) and also on some novel ideas regarding learning-based intelligence testing.

The AIP Tests suggested here are specifically intended for AGI systems that control agents embodied in 3D worlds resembling the everyday human world, via either physical robots or virtually embodied agents. Very differently embodied AGI systems (e.g. systems to be initially taught purely via text without any simulated human-like or animal-like body) would potentially need qualitatively different testing methdologies.

Apologies and Warnings

Just to set expectations properly, I note up front that I am not going to articulate here any extremely crisp, simple AGI testing method that could easily be used to create some sort of "X Prize" analogue for AGI's. I have thought about the latter possibility a great deal and have come to the conclusion that it probably is not a good direction to follow. General intelligence, by its nature, is complex and multifaceted and doesn't lend itself to simple test scenarios; this is why human intelligence tests come in multiple flavors, all of which are long and contain numerous questions of various types. And, testing for particular sorts of intelligent accomplishments, while easier to do than testing for general intelligence, seems inevitably to lead down the road of encouraging test-focused narrow-AI development rather than AGI development focused on the creation of AGI systems with truly broad and creative intelligence. However (obviously, since I wrote this essay), I do think that AGI intelligence testing is an important area worth of thought and consideration in the AI field – even though the result of work in this area is

[1] Pronounced "ape tests" ;-)

likely to be a "AGI IQ tests" that, like the AIP tests suggested here, are complex and multifaceted rather than simple, crisp and elegant.

Another caveat is that I am not here attempting to approach the problem of assessing "general intelligence" in a truly mathematically broad sense, as addressed e.g. in Legg and Hutter's (2007) formalizations of the intelligence concept. This is the meaning of the qualifiers "Human Level, Human-Like" in the title of the essay. My goal is to explore ways of testing early-stage versions of AGI systems that are aimed at being as smart as humans (and potentially ultimately smarter), and at being smart in ways that are rougly similar to the ways humans are smart. Undoubtedly there are many other ways of being smart, including many that we humans would never recognize as intelligence.

Of course the notions of "human level" and "roughly human-like" are both vague, non-rigorous notions; and this doesn't mean they can't be use to conceptually motivate rigorous tests, but it does mean that human common sense is going to have to be applied to determine the contexts in which the tests proposed here should be applied. For example, an AGI system aimed solely at mathematical theorem-proving would fail most of the tests proposed here, even if it had incredibly general and deep intelligence in the mathematical domain; and this is not surprising because this AGI would manifestly fails to fulfill the "roughly human-like" criterion, even if it is intuitively "human-level" or even superhuman. Similarly an AGI successfully emulating chimp would fail most of the tests proposed here, even though it would constitute an extremely important and impressive achievement; and this is not surprising because this AGI would manifestly fail to fulfill the "human-level" criterion.

Why Not Focus on Testing Individual Components?

One suggestion that is sometimes made, regarding AGI testing, is: "If objectively assessing an AGI's overall functionality is such a complex matter, then why not just test its individual components and validate that they work really well? If an AGI has components that are better than the best-of-breed components in today's narrow-AI systems, this surely tells you something." My position is that this is a fatally flawed approach to assessing AGI intelligence. My focus here will be solely on testing holistic system functionality, not testing the functionality of individual components of AGI systems

Of course, systematic and comparative testing of system components can be valuable, and we have done plenty of it in the AGI projects I've been involved with: for instance, testing the MOSES procedure learning component we use in NCE/OpenCog against other program learning algorithms (see Looks, 2006; and testing the PLN probabilistic logic framework we use in NCE/OpenCog against other probabilistic logic approaches (see the Appendix of Goertzel et al, 2008). However, there is a very deep problem with this sort of testing as approach to AGI IQ assessment, which is that it is generally not meaningful to compare AGI system components against other AGI system components that look similar on the surface, but actually embody radically different theoretical assumptions, related to their different roles in the overall systems in which they are embedded. Apparently similar components may potentially play subtly different roles in the overall AGI systems in which they are embedded. To drive this point home

thoroughly, I will now spend a couple paragraphs recounting in moderate detail one example in which this problem arose in my own work, in the comparative testing of the PLN inference engine used in NCE/OpenCog with the NARS inference engine used in Pei Wang's NARS AGI system. (The details of this example may be slightly opaque to readers not familiar with NARS or PLN, but it seems hard to give a concrete, detailed example that would not be somewhat obscure in a similar way.)

An example of the kind of comparison we did[2] was the following sort of inference: consider

```
Ben is an author of a book on AGI <tv1>
This dude is an author of a book on AGI <tv2>
|-
This dude is Ben <tv3>
```

versus

```
Ben is odd <tv1>
This dude is odd <tv2>
|-
This dude is Ben <tv4>
```

Here each of the English statements is a shorthand for a logical relationship that in the AI systems in question is expressed in a formal structure; and the notations like <tv1> indicate uncertain truth values attached to logical relationships, In both NARS and PLN, uncertain truth values have multiple components, including a "strength" value that denotes a frequency, and other values denoting confidence measures. However, the semantics of the strength values in NARS and PLN are not identical.

Doing these two inferences in NARS you will get

```
tv3.strength = tv4.strength
```

whereas in PLN you will not, you will get

```
tv3.strength >> tv4.strength
```

The difference between the two inference results in the PLN case results from the fact that

```
P(author of book on AGI) << P(odd)
```

and the fact that PLN uses Bayes rule as part of its approach to these inferences.

My initial reaction, on getting these results, was that the NARS results seemed not to make intuitive sense, because I was sure that any intelligent human, on being presented with these inferences, would assign tv3.strength >> tv4.strength. However, when I discussed the issue with Pei Wang, the creator of NARS, he responded by saying,

---

[2] This comparative testing was done by Izabela Freire in 2002, using research funding provided by David Hart

roughly (I'm paraphrasing him loosely, with some risk of unintentional error) that there are other ways of indirectly accounting for the fact that

```
P(author of book on AGI) << P(odd)
```

in NARS, and thus that just feeding NARS the above syllogisms without other background knowledge is not a fair comparative test ... instead you'd need to compare NARS vs PLN on these syllogisms in the context of a rich database of background knowledge, with overall properties similar to those that one would find in a system that had gained its knowledge from life-experience.

This example illustrates the subtlety of comparatively testing inference engines (or AGI system components in general) from an AGI perspective. And it reinforces the notion that the right metrics for AGI systems will almost surely have to do with the overall behaviors of systems controlled by the AGI systems (for example embodied agents like physical or virtual robots), rather than concerning themselves with abstracted, lower-level functionalities like individual inference steps (which, even if they look very similar, may mean different things to different AGI systems or algorithms).

Testing different inference engines on the same formal structures, may not tell you much of anything if these different inference engines interpret these same formal structures differently. However, doing tests involving controlling robots or virtual agents, or holding English conversations, bypasses this problem via referring to an "objective" world whose interpretation is approximatively shared by the humans ultimately doing the evaluating.

Neither PLN nor NARS's inference engine is intended as a whole AGI system -- each one is intended as part of an overall AGI design, in which it receives outputs from certain other system components, and gives outputs to certain other system components. If the other components of NCE/OpenCog control PLN inputs/outputs in a manner that systematically differs from the way the other components of the NARS systems control NARS inference engine inputs/outputs, then this makes it very hard to compare the two inference systems. This is a subtler issue than it may at first seem, because the different manners of controlling inputs/outputs may embody different conceptual and semantic assumptions. It is logically quite possible that both PLN and NARS could work well within the systems they are designed for, but work poorly if swapped and placed into the contexts designed for each other – even if their inputs and outputs have the same syntactif form and closely related (but not identical) semantics.

Another, related, simpler point is that focusing on testing individual system components tends to lead AI developers down a path of refining system components for optimum functionality on isolated, easily-defined test problems that may not have much to do with general intelligence. It is possible of course that the right path to AGI is to craft excellent components (as verified on various isolated test problems) and then glue them together in the right way. On the other hand, if intelligence is in large part a systems phenomenon, that has to do with the interconnection of reasonably-intelligent components in a reasonably-intelligent way (as I have argued e.g. in Goertzel, 2006), then testing the intelligence of individual system components is largely beside the point: it may be better to have moderately-intelligent components hooked together in an AGI-appropriate way, than extremely-intelligent components that are not able to cooperate with other components sufficiently usefully.

Ultimately, studying the functionality of individual system components to assess overall system intelligence makes no more sense than studying the properties of a runner's muscles, heart, lungs etc. to assess how fast they can run. Of course, a runner's internal properties are going to be correlated with their speed, but these correlations are going to be complex and require much research to unravel, in part because of subtle dependencies between body parts. Whereas direct assessments of a runner's speed, or an AGI system's behaviors, are far less theory-laden and hence more appropriate as approximatively "objective" measures.

Online School Tests: A Pragmatic Replacement for the Turing Test

The classic approach to assessing whether an AI has achieved human-level general intelligence is the Turing Test (Turing, 1950), which measures the ability of an AI to fool humans, in a conversational context, into believing it's human.

However, the Turing Test has proved a singularly poor guide for the development of early-stage AGI systems. The Loebner Prize, which is given each year to the AI system that comes the closest to passing the Turing Test, has in practice had very little to do with real work toward general intelligence. An early-stage AGI is almost inevitably going to be far worse at holding humanlike English conversations than a well-crafted chatbot filled with a bunch of stock phrases but no real understanding.

On top of the "chatbot" problem, the Turing Test also has additional issues: it is obviously problematic for AGI approaches that are oriented toward making human-level and roughly *but not strictly* human-like AI systems. For an AI system like this, impersonating a human may not necessarily be a fair nor useful test. Is it really fair to demand that an AI be able to believably describe the feeling of a stomachache, a hangover, or warm rain falling lightly on the back of one's neck? This seems roughly as fair as demanding that a human be able to believably describe the particular psychological sensation of a hard drive failure, or the exquisite combination of joy and disturbance resulting from an overly rapid increase in the polygon resolution of one's fellow agents in a virtual world. Of course, Turing did not intend his test as a necessary criterion for human-level intelligence nor as a practical goal for AI development; in his original conception it was more of a challenge to those who conceive intelligence in non-functional terms.

But if we don't want to use the Turing test, what is the alternative for assessing achieved human-level, roughly human-like AGI? One approach, I suggest, is the "Online University Test." If an AI can get a BA degree at a real university, via online coursework only (assuming for simplicity courses where no voice interaction is needed, only textual and mouse-based communication), then I suggest we should consider that AI to have human-level intelligence. Note that the coursework spans multiple disciplines, and the details of the homework assignments and exams are not known in advance (otherwise students would be able to cheat too easily). Some basic social interaction and natural language communication are needed here, as well as understanding of course material, ability to do online research, and ability to solve problems. However, there is no requirement to be strictly humanlike in order to pass university classes.

There are also online high schools and even elementary schools[3], so one can also postulate related Online Highschool Test and Online Elementary School Tests -- though it is unclear how much easier these tests would be for AI systems, as for many AI systems, the hard parts will not be the course material itself, but rather the social and linguistic aspects of the online education (i.e., "figuring out what are the problems to be solved," rather than solving the problems). We may group all these possibilities under the heading of an "Online School Test" methodology.

The Online University Test is fine as a criterion for what it means to create a "human level, roughly humanlike AGI." But it isn't much use as a guide for incremental development towards this goal. Arguably, by the time one has a system that can pass the Online Elementary School Test, one has already passed the most difficult phases of AGI design and engineering. Thus, the really tricky question regarding evaluation and metrics regards how to measure the development of AI systems that haven't yet achieved the functionality of a human elementary school student. We may formulate this problem as the challenge of creating an appropriate series of Preschool AI Tests, with a goal of measuring an AI's incremental progress toward Elementary School Intelligence (ESI).

Challenges in Creating Preschool AI Tests

One of the major challenges in creating a Preschool AI Test is that different approaches to AGI may naturally be evaluated by different sorts of tests. Any set of tests one creates, with the view of measuring an AI system's incremental progress toward elementary-school intelligence, is naturally going to favor some paths to ESI over others. In spite of this inevitable bias, however, it seems important to articulate Preschool AI Tests anyway. If different approaches to AGI come along with different tests, this is not ideal, but is by no means an insuperable obstacle to progress. Competitive comparison of different approaches is one purpose of testing, but not the only one: well-crafted tests are also valuable simply for helping AGI developers to understand what their systems are capable of.

The specific Preschool AI Test approach I'll suggest in this essay is oriented toward AGI systems that are physically or virtuall embodied, and won't be directly applicable to other sorts of AGI systems. Some parts of the suggested approach will apply to (for instance) purely text-chat-based systems, others will not.

Another major challenge is the problem of "cheating." By this I don't mean cheating on the part of the AI (such as surreptitiously instant-messaging its creator for answers), but rather on the part of the AI designer. Over and over again, in the history of AI, we've seen the danger of "overfitting an AI system" to a specifically, narrowly defined goal or set of goals. Over and over again, it turns out that hacks or narrow-AI cleverness of various sorts can be used to achieve a set of specific goals which at first seemed to require general intelligence ... without really capturing the spirit in which the goals were originally proposed. One can substantially work around this problem by making one's test broad enough in nature, but this isn't as easy as one might think.

Due to these challenges, it seems to me that the most important assessments of intermediate stages of AGI development are necessarily going to be qualitative.

[3] e.g. http://www.e-tutor.com/elementary.php

Objectively measurable milestones are going to be very useful for testing, tuning and tweaking AGI systems -- when they are used in the context of a deep understanding and appreciation of the qualitative goals. But I suggest that Preschool AI Tests should not be used as the primary tool for structuring development of early-stage AGIs; rather, only as a tool for helping guide developers to maximize quantitative progress along lines that are qualitatively sensible in terms of a deep underlying cognitive/AI theory.

Naively, it might seem that creating a variety of different test problems (which is part of the approach I'm going to suggest later on in this essay) could circumvent the "cheating" challenge. However, a moment's consideration shows that diversity is not a cure-all. Suppose one poses 50 different test problems, qualitatively different in nature. One "trivial" approach to passing these tests would be to create a narrow-AI approach to each one of the 50 problems separately, and then wrap up these 50 specialized solutions inside a common external interface.

Furthermore, it's not wholly clear where the boundary between this trivial "cheating-based" approach and serious AGI design lies. For instance, suppose two of the 50 problems in one's test set involve navigation in complex environments. Is it "cheating" to create a specialized navigation process within one's AGI system, or not? Eric Baum (author of "What Is Thought?"; Baum, 2004) is one serious AGI thinker who believes that hard-wiring navigation into an AGI system is the correct thing to do: he strongly feels that the human brain has an in-built navigation module, and that an engineered AGI system should have one too. In my own work with the Novamente Cognition Engine and OpenCog Prime, we have implemented a hard-wired navigation system for practical applications, but for future development are leaning toward a middle path, in which certain high-level spatial-movement functions useful for navigation are exposed to the AI's learning algorithms primitives, but the AI system must learn to compose these functions into a real navigation algorithm. I think this can lead to a more flexible and adaptive navigation algorithm than directly providing the AI with navigation algorithms ... but, whether this difference would be apparent on a simple navigation-based test problems, is not clear. Quite possibly, hard-wired navigation algorithms could be humanly-tweaked to do very well on a couple narrow classes of navigation-based test problems, and a learning-based approach might have trouble competing. After all, not all humans are all that good at navigating, either. However, if an AI system had to learn to navigate in an unfamiliar sort of environment, then the learning-based approach would obviously be more powerful.

One obvious, partial solution to the cheating challenge is not to reveal to the AI nor the AI designer too many specifics of the tests, in advance. The general nature of the tests should be revealed, but not the details. For instance: Perhaps the testers could reveal that some tests will require moving around in crowded environments, but not the specifics of what sort of navigation testing will be done.

Another partial workaround is to test, not just what an AI system can do, but what it can learn based on certain types of feedback. For instance, one could test an AI's ability to navigate in a certain environment, then give it some lessons on navigation, and then see how well it is able to navigate after that. This is by no means an ironclad defense against cheating, because an AI designer could always program an AI with both the knowledge of navigation, and the propensity to pretend not to know how to navigate until navigation lessons have been received. One can work around this problem as well

to an extent, by using the Randomized Learning Based Test method that I'll describe below – but even this is not a complete solution. Ultimately, we are brought back to the point that qualitative assessment is going to be the most important thing at the AI preschool level. The purpose of tests and metrics, at this stage, is going to be to guide qualitative assessment, rather than to replace it.

Randomized Learning Based Testing Methodology

Let us define a Learning-Based Test as consisting of three parts:

1. a pre-test
2. some (generally interactive) instruction
3. a post-test, that measures how well the learner has learned from the instruction

For instance, one might

1. test an AI's ability to correctly identify the emotion associated with a gesture
2. give it some interactive instruction on identifying emotions associated with gestures (e.g. by explicitly telling it "When I do this I'm happy" while smiling; or else by giving it positive and negative reinforcement signals when it makes correct vs. incorrect judgments)
3. then re-test its ability

As noted above, the problem with this sort of test is that (to continue with the above example) an AI designer could potentially pre-program their AI system with the capability to associated emotions with gestures, and also with the propensity to feign ignorance about this until instructed. We may call this the "Nintendogs problem," as the popular virtual-pets game involves animated dogs that are preprogrammed to "act as if they're learning" various behaviors – when in fact the code for the behaviors is supplied in advance, along with code telling them to do these behaviors correctly only after receiving a certain amount of reinforcement.

A partial workaround for the Nintendogs problem is what I call Randomized Learning Based Testing Methodology, or RLB testing for short. RLB testing takes advantage of the fact that with AI's, unlike human children, it is possible to create multiple copies of the same AI and give each of them different instructions. However, it only works for the teaching of things that are in some sense arbitrary, rather than "natural." The idea is as follows:

1. Give an AI a pre-test
2. Then, create N copies of the AI, and place them out of communication with each other
3. Give each of the copies of the AI a separate instructional experience, aimed at teaching a somewhat different set of specific skills (but of the same general nature)
4. Give each of the copies of the AI a post-test, that measures how well the learner has learned from the instruction

So, for example, to continue with the emotion/gesture identification task, in the RLB method one of the copies might be taught that smiling indicates happiness, whereas another might be taught that it indicates anger, and another might be taught that it indicates sadness. The problem of course is that if the AI has been watching movies or studying images of people, it may have already learned that smiling really indicates happiness – so that some of the copies are being asked to learn plainly artificial, fake information, whereas others are being asked to learn information that is accurate in the context of the real world. On the other hand, any AI subjected to the test would be subjected to the same protocol, so there's nothing unfair about it.

Teaching an AI the rules of a game like baseball is another good example for this sort of methodology. The rules of baseball are fairly arbitrary, so that there should be no problem teaching different copies of an AI different variants of the rules. Furthermore, there are many different variants so it's not very likely that a clever, nefarious AI designer is going to preprogram their AI with a knowledge of all the variants that the clever, nefarious test designers are likely to cook up.

On the other hand, this sort of methodology seems less likely to be effective in contexts like language learning. Yet, even here there are some tests one could easily apply RLB too. For instance, one could make up fake words of different types – one could teach copy 1 of the AI the proper use of "fnorbulate", teach copy 2 of the AI the proper use of the word "gttrbuckular", and so forth. This would certainly test the ability of the AI to learn usage of different words of different sorts. Making up new grammatical rules to teach different copies is harder because we don't know as much about what makes a "psychologically natural" grammar rule, and it's harder for us as teachers to effectively and naturalistically use a made-up grammar rule, as opposed to a made-up word.

RLB is not a sufficiently powerful idea to fully overcome the cheating challenge associated with AGI testing, but, it does seem a worthwhile addition to the arsenal of AGI testing methodologies.

The Multiple Intelligences Approach

The specific approach I suggest for an AGI Preschool Test, for the case of AGI systems with roughly humanlike physical or virtual embodiment, is based on the learning based and RLB testing methodologies introduced above, combined with the psychological notion of multiple intelligences.

"Multiple intelligences" is a psychological approach to intelligence assessment based on the idea that different people have mental strengths in different high-level domains, so that intelligence testing should contain tests that focus on each of these domains separately. My suggested use of the multiple intelligences framework for AGI is not particularly tied to the value (or otherwise) of the framework for assessing human intelligence. The value of the framework for assessing AGI intelligence lies in its explicit attention to the broad, general scope of human intelligence.

The following table[4] summarizes the key intelligences posited within the theory:

Intelligence	Aspects	Tests
Linguistic	words and language, written and spoken; retention, interpretation and explanation of ideas and information via language, understands relationship between communication and meaning	write a set of instructions; speak on a subject; edit a written piece or work; write a speech; commentate on an event; apply positive or negative 'spin' to a story
Logical-Mathematical	logical thinking, detecting patterns, scientific reasoning and deduction; analyse problems, perform mathematical calculations, understands relationship between cause and effect towards a tangible outcome or result	perform a mental arithmetic calculation; create a process to measure something difficult; analyse how a machine works; create a process; devise a strategy to achieve an aim; assess the value of a business or a proposition
Musical	musical ability, awareness, appreciation and use of sound; recognition of tonal and rhythmic patterns, understands relationship between sound and feeling	perform a musical piece; sing a song; review a musical work; coach someone to play a musical instrument; specify mood music for telephone systems and receptions
Bodily-Kinesthetic	body movement control, manual dexterity, physical agility and balance; eye and body coordination	juggle; demonstrate a sports technique; flip a beer-mat; create a mime to explain something; toss a pancake; fly a kite; coach workplace posture, assess work-station ergonomics
Spatial-Visual	visual and spatial perception; interpretation and creation of visual images; pictorial imagination and expression; understands relationship between images and meanings, and between space and effect	design a costume; interpret a painting; create a room layout; create a corporate logo; design a building; pack a suitcase or the boot of a car
Interpersonal	perception of other people's feelings; ability to relate to others; interpretation of behaviour and communications; understands the relationships between people and their situations, including other people	interpret moods from facial expressions; demonstrate feelings through body language; affect the feelings of others in a planned way; coach or counsel another person

[4] This table is borrowed with minor modifications from www.businessballs.com/howardgardnermultipleintelligences.htm

Whether all the intelligences in this table are necessary to consider from an AGI perspective is not clear. The necessity of the linguistic, interpersonal, spatio-visual and logico-mathematic intelligences is obvious: without these, there is no way an AI will pass the Online Elementary School Test, for example. Musical intelligence can potentially be ignored for the purpose preparing an AGI for Online School Tests; but the situation with Bodily-Kinesthetic intelligence is less clear: it may be that achieving some measure of Bodily-Kinesthetic intelligence is going to be critical for the understanding of linguistic metaphors related to bodily-kinesthetic activity, which are rampant in ordinary language.

My specific suggestion for testing preschool-level AGI systems is to create a number of test categories based on each of the multiple intelligences listed above (and the phrases in the third column are examples of potential test categories). Then, for each of these categories, multiple specific tests may be generated, using learning-based and RLB testing whenever possible. To have a good testing methodology, AGI's and their developers shouldn't know the specific tests to be used in advance anyway, but only the general categories. Specific examples of tests within each category should be provided for guidance, but the actual tests given should not rigidly imitate the specifics of the example tests.

I will here give examples of possible tests within each of the five types of intelligence mentioned above (excluding only musical). In the examples I'll use the case of an AI controlling agents in virtual worlds, for sake of concreteness, but the same examples obviously apply to physical robotics.

A Linguistic Test

An example test of linguistic intelligence is the task of writing a set of instructions. Suppose we have two human-controlled avatars, A and B, and one AI-controlled avatar. And, suppose A shows the AGI how to carry out some task X, and then leaves. The AI's job is then to show B how to do that same task X.

This has many variants, including cases where the best way to describe X is purely verbal, and others where the best way to desribe X involves a combination of words and actions.

A concrete example would be teaching someone how to assemble a piece of furniture, similar to the furniture kits one buys at K-mart or Staples ... or a bicycle. Of course, the specific type of item to be assembled would not be known to the AGI or AGI designer prior to the test being given.

Using the RLB methodology, the AGI's could be given a period of feedback regarding how well they gave instructions ... and then after the feedback period, could be tested on how well they absorbed the instructions.

A Logico-Mathematical Test

Example tests of logico-mathematical intelligence are the task of creating a process to measure something difficult, or compare two or more entities regarding some quantity that is difficult to measure. One paradigm that can be used here is that of indirect comparisons. To quote extensively from (Reece et al, 2001)

Indirect comparisons require the ability to make two kinds of mental relationships - transitive reasoning and unit iteration - which are best explained by the following task. Piaget et al. asked children in individual interviews to build a tower having the same height as a model 80 cm tall (Piaget et al., 1948/1960). The child's tower was to be built on a table 90 cm lower than the base of the model. The child was given smaller blocks than the ones used in the model so that one-to-- one correspondence was impossible. Long strips of paper, as well as a ruler and three sticks were provided - a stick 80 cm long, one that was longer, and one that was shorter than 80 cm.

Before the age of about seven, children did not use the sticks or ruler; and when 3-, 4-, and 5-year-olds were asked if a stick or a ruler might be useful, they answered "No." Five-year-olds consistently wanted to bring the two towers together for direct comparison, but the interviewer did not allow this action. Children then used various body parts in an attempt to compare the two towers as precisely as possible.

Finally, around the age of seven, children began to use one of the longer sticks as a third term. (Here, the model tower and the copy were the first two terms, and the stick was the third.) Seven-year-olds marked the height of the model on the stick, took the stick to their tower, and made their tower as tall as the height indicated on the stick. Piaget et al. explained this use of a stick as a manifestation of transitive reasoning that becomes possible when the child's logic has developed.

Transitivity refers to the ability to deduce a third relationship from two (or more) other relationships of equality or inequality. The child who can reason transitively can deduce that if the height of the model tower and the length marked on the stick are equal (a direct comparison), and this length marked on the stick is equal to the height of his or her tower (another direct comparison), then the height of the two towers must be the same (a deduction). Most children before the age of seven cannot understand this logical reasoning, even if it is explained to them.

Piaget et al. went on to show a small block to the children and asked if it could be used to compare the height of the two towers. The children who had demonstrated transitive reasoning responded in one of two ways: The less advanced group said that the small block was too small to be of any use, but the more advanced group used it as a unit to iterate and count. These children placed the small block at the bottom of the tower, marked the upper end of the block on the tower, moved the block up until its lower end was exactly on the mark, and repeated the same procedure upward to the end of the tower, without any gaps or overlappings. These actions showed that the child thought

about the length of the block as a part of the tower's height. Recent research indicates that a majority of children become able to iterate a unit of length around third grade (Kamii & Clark, 1997). Note that when children have developed the logic of unit iteration, their measurement becomes exact.

It is easy to see how one could create a variety of indirect comparison tests along the lines of the above. Furthermore, one could easily use an RLB approach in this case, giving different sorts of structures and measuring implements to different replicates of an AI, to avoid the risk of the AI being specialized to some particular type of structure or measuring implement.

A Bodily-Kinesthetic Test

An example bodily-kinesthetic test would be the ability to communicate observed activities using mime. The AGI would watch human agent A carry out a certain action involving other agents or objects; then these agents or objects would be removed from the scene, and the AGI would need to do a mime for human agent B, indicating to A the activity in question. This is basically the game of "charades."

Another example would be the ability to teach another agent a dance. Human agent A teaches the AGI a dance, and then goes away; and the AI is then supposed to teach human agent B the dance. The AGI will have to demonstrate the dance, but then also correct B if B does it wrong, and explain the right way to do it. This particular example would be hard to do given current virtual world technology, but would be easy in near-future virtual worlds enabled with better haptic devices and finer-grained avatar control.

A Spatial-Visual Test

An example test of visuospatial intelligence is the creation of a room layout. Suppose an AGI is told what people are going to live in a house, and a few things about them, including what their tastes and occupations are. Then the AGI has to figure out what furniture they need and how to arrange it. The human occupants then rate the room layout based on how much they would like to live in the room. This lends itself well to RLB since different people may have very different tastes.

Another test would be the ability to draw "cave paintings" – i.e., given a simple marker or paintbrush, to create images that evoke particular objects. The AGI would be shown an object, and would then need to draw a picture conveying the object to a human viewer, the accuracy being judged by whether the human could correctly identify the object (from among a long list of choices). To make the test more interesting, using an RLB approach one could have the final test involve a variety of different artistic media: pens, paint ... rocks arranged on the ground, etc.

An Interpersonal Test

An example test of interpersonal intelligence is the recognition of feelings through body language or tone of voice. Recognition of feelings is an interesting test task

because of how well it fits in with the RLB methodology: one can easily have human testers express their feelings in odd ways (different ways to each AGI copy) and see how well the AGI adapts. Also, one can use testers from different cultures, who habitually express feelings in different ways. The measurement of accuracy is easy here of course: one simply asks the AGI what the human it's interacting with is feeling.

Virtual worlds are fairly weak for expression of feeling except through voice, but use of haptic interfaces and cutting-edge virtual-world technology would make this sort of testing possible. One could also simply use videos of human faces for this sort of task, though this requires AGIs with strong vision processing components.

Another interpersonal intelligence test, not requiring so much on the perception side, is listening to a conversation between people (preferably in an embodied context where the AGI can see the world the people are talking about) and telling when they are joking. Again accuracy assessment is pretty easy here: one just needs the AGI to report when it thinks the people are joking. This lends itself very well to an RLB approach because different people can have such different senses of humor: the "humor teachers" may well have different senses of humor than the conversors that the AI listens to during the final testing phase.

Conclusion

While I have not specified a concrete, usable "AGI IQ test" here, I believe I have laid out a direction along which such a test could practicably be constructed. The next step would be to make the ideas of the prior section more concrete, and create a variety of conceptually similar tests embodying different test categories probing the multiple intelligences.

Ideally, one would like to see a number of different researchers, proponents of different designs aimed at human-level, roughly human-like AGI, agree to a common testing approach, such as a specific incarnations of the Online School Test and AGI Preschool Test proposed above. Such agreement could be a valuable step in terms of crispening the focus of the AGI research community.

References

- Baum, Eric (2004). What is Thought?, MIT Press
- Gardner, Howard. (1983) "Frames of Mind: The Theory of Multiple Intelligences." New York: Basic Books.
- Goertzel, Ben (2006). The Hidden Pattern. Brown Walker Press.
- Goertzel, Ben, Matthew Ikle', Izabela Freire Goertzel and Ari Heljakka (2008). Probabilistic Logic Networks. Springer.
- Legg, Shane and Marcus Hutter (2007). Universal Intelligence: A Definition of Machine Intelligence. In Minds and Machines, pages 391-444, volume 17, number 4, November 2007.
- Looks, Moshe (2006). Competent Program Evolution. PhD Thesis in Computer Science Department, Washington University, St. Louis.
- Reece, Charlotte Strange, Kamii, Constance (2001). The measurement of volume: Why do young children measure inaccurately?, School Science and

Mathematics, Nov 2001, http://findarticles.com/p/articles/mi_qa3667/is_200111/ai_n9009076

- Turing, Alan (October 1950), "Computing Machinery and Intelligence", Mind LIX(236): 433–460, http://loebner.net/Prizef/TuringArticle.html
- Wang, Pei (2006). Rigid Flexibility: The Logic of Intelligence. Springer.

Running Head: ABILITY, BREADTH, PARSIMONY

ABILITY, BREADTH AND PARSIMONY IN COMPUTATIONAL MODELS OF HIGHER-ORDER COGNITION

Nicholas L. Cassimatis
Rensselaer Polytechnic Institute
cassin@rpi.edu
Telephone: 518-276-3853
Fax: 518-276-3015

Paul Bello
Office of Naval Research
paul.bello@navy.mil
Telephone: 703-696-4318
Fax: 703-696-1212

Pat Langley
Arizona State University
pat.langley@asu.edu
Telephone: 480-965-8850
Fax: 480-965-2751

Keywords: higher-order cognition, human-level intelligence, cognitive models

Abstract

Computational models will play an important role in our understanding of human higher-order cognition. How can we evaluate a model's contribution to this goal? We argue that three important aspects of a model of higher-order cognition to evaluate are (a) its ability to reason, solve problems, converse and learn as well as people do, (b) the breadth of situations in which it can do so and (c) the parsimony of the mechanisms it posits. We argue that fits of models to quantitative experimental data, though valuable for other reasons, do not address these criteria. Further, using analogies with other sciences, the history of cognitive science and examples from modern-day research programs, we identify five activities that have been demonstrated to play an important role in our understanding of human higher-order cognition. These include modeling within a cognitive architecture, conducting artificial intelligence research, measuring and expanding a model's ability, finding mappings between the structure of different domains and attempting to explain multiple phenomena within a single model.

1. Understanding higher-order cognition

Computational modeling is a particularly important part of understanding higher-order cognition. One reason for this is that precise models can help clarify or obviate often troublesome theoretical constructs such as "representation" and "concept". A second is that the characteristics of human intelligence appear to be so different from other topics of scientific research as to call into question whether a mechanistic account of human intelligence is possible. Being instantiated in a computational model would resolve doubts about whether a theory was implicitly presupposing an intelligent "homunculus" and would make the possibility of intelligence resulting from natural phenomena more plausible.

In this paper, "higher-order cognition" refers to inference, problem solving and language use that goes beyond immediate sensations and memories. Such cognition is often studied in subfields such as reasoning, problem solving, analogy, syntax, semantics and pragmatics. We are particularly concerned with a challenging puzzle: how do mechanisms that do not exhibit higher-order cognition (such as retrieval from long-term memory and access to short-term memory buffers) combine to produce higher-order cognition? How do mechanisms that do not reason, use language, or have goals combine into a system that does? No known modeling approach, including production systems, dynamical systems or neural networks, currently exhibits the full range of higher-order human cognition. Our aim is to eliminate the gap between current approaches to cognitive modeling and the abilities of human intelligence.

Two desired traits of cognitive models are *ability* and *fidelity*. That is, we attempt to identify mechanisms that have the power and flexibility of human intelligence. We also wish to

confirm that these mechanisms are at least similar to those that underlie human cognition. How should we evaluate a cognitive model's progress towards these goals?

Like Newell (1973), we argue for the need to move beyond models of isolated cognitive phenomena. We claim that, although fitting model behavior to human data can be an important activity, it is but one method for evaluating cognitive models. We then stress the importance of ability, breadth and parsimony in model evaluation. Finally, we use ongoing research programs to illustrate how cognitive models can be motivated and evaluated using these criteria.

There are many discussions that touch on ability, breadth and/or parsimony. For example, Anderson and Lebiere (2003) use Newell's (1990) criteria for cognitive models to compare the ACT-R and connectionist approaches to modeling. Many of these criteria relate primarily to ability, although the range of abilities is broad and parsimony is briefly mentioned. This paper differs from previous discussions in relating ability to model fitting, giving examples of concrete and precise measures of these criteria and by motivating the discussion explicitly from the goal of understanding higher-order cognition.

2. The model fit bias

The observe-hypothesize-test view of the scientific method is often applied to evaluating cognitive models thus: first one collects quantitative human data in an experimental setting, then one develops a cognitive model that reproduces this behavior and predicts unobserved behavior, after which one conducts further experiments to confirm these predictions. Although such work can play an important role in evaluating models and we do not claim it is unnecessary or somehow undesirable, we argue that over-emphasizing model fits relative to other criteria does

not address all the goals of cognitive science and can impede progress towards models that provide general accounts of higher-order cognition.

2.1. Modeling the fundamental data

People can solve problems in new situations, carry out complex chains of reasoning, interpret what they see and perceive, engage in extended conversations about a wide range of topics and learn complex structures that support these abilities. There are no existing cognitive models that can reason, solve problems, converse or learn as well as people. Although there are models that predict reaction times or error rates in specific situations, they do so only on one or a few tasks and thus do not generally account for any of the aforementioned human abilities.

The fact that models with exemplary, near-perfect fits to quantitative data are possible to construct without making significant progress towards expanding the situations and domains over which cognitive models can deal with illustrates that the model fits alone are not sufficient for evaluating models of higher-order cognition. For example, consider two hypothetical sentence processing models. Model A fits eye movement and reaction time data but makes no inferences about the sentence meaning. Model B makes many correct inferences about sentences (measured, for example, by answering questions about them) at a level that far exceeds the state of the art in computational linguistics or cognitive modeling. Model B does not predict reaction times or eye movements. Model B would clearly be an important contribution because it advances our ability to provide computational accounts of the inferential power evident in human language use. However, if we primarily emphasize model fits, then model B does not count as much of a contribution and model A must be favored. Our claim is not that model B is better

than model A. Model A is very likely to account for processes that B does not. Rather, our claim is that over-emphasizing model fits can under-emphasize certain kinds of modeling efforts that make progress towards solving some very difficult computational questions about higher-order cognition.

This point can also be made in terms often used to motivate the importance of model fits. The role of models (and scientific theories generally) is to explain and predict observations. The number and range of observations a model explains and predicts is often used to test how accurately the model characterizes reality. These observations can include data carefully collected in a laboratory as well as readily-observable facts such as the daily rising of the sun and the solubility of salt in water. One of the most unique observerable facts about humans is their ability to reason, solve problems, converse and learn. Thus, when evaluating the cognitive plausibility of a model, among the observations that should be considered are those that pertain to ability. In short, ability is part of the data on human cognition and the extent to which a model has this ability is an important part of evaluating its plausibility as a model of human cognition.

2.2. Enabling ability before fitting models

In order fit a model to data about performance in a task, one must first have a model that can perform the task. In the case of higher-order cognition, however, it is often the case that no computational methods are known that can exhibit the ability to perform many tasks. Discovering computational methods with this ability is therefore important and has many characteristics that distinguish it from model fitting work.

In cognitive modeling research, there is often more than one mechanism that produces a particular kind of behavior. For example, there are both neural network (McClelland & Patterson, 2002) and rule-based (Pinker & Ullman, 2002) accounts of past-tense morphological processing and there are both mental model (Johnson-Laird, 1983) and mental logic (Braine & O'Brien, 1998; Rips, 1994) accounts of behavior in many reasoning tasks. In these cases, research often attempts to determine which mechanisms are actually involved in these tasks by observing behavior corresponding to the differing predictions each model makes.

In much of higher-order cognition, however, there are no known mechanisms that exhibit the kind of behavior we seek to understand. We know of no computational processes that can learn, use language, reason or solve problems in as wide and complex a set of situations as people. In such cases, it is impossible to fit models because there are no candidate models to fit.

Since finding mechanisms that enable such models is such a difficult problem, it cannot merely be treated as a preliminary stage of model fitting research. Finding computational methods with human-level cognitive abilities will likely involve several steps of progress along the way, each of which will need to be evaluated in some manner. To the extent that quantitative model fits are not well-suited to measuring ability, additional criteria will be required. Since finding computational methods with human-level ability is a large task, it will require many projects evaluated primarily according to those criteria. The remainder of this paper proposes some such criteria and illustrates their use.

3. Evaluating the ability, breadth and parsimony of a cognitive model

Although we have argued that quantitative model fitting, as typically defined, is not alone sufficient for evaluating accounts of higher-order cognition, the field still requires guidelines for measuring progress. In this section, we propose some additional criteria for evaluating cognitive models. For each criterion, we discuss its analogues in other fields and the ways in which it has played an important role in the history of cognitive science. Unfortunately, just as there is no all-encompassing precise definition or procedure for model fitting, we must content ourselves for now with general descriptions of these criteria and specific methods of using them in certain situations. In subsequent sections, we will provide examples of such methods.

3.1 Ability

We have argued that one of the most interesting and important facts about cognition is that people can solve novel problems, make nontrivial inferences, converse in many domains and acquire complex knowledge structures, and that they can do so at a level beyond the reach of currently-known computational methods. Thus, it is important to ask how much a model advances the ability of computational methods to explain and produce higher-order cognitive phenomena.

Many important contributions to cognitive science have involved ability. Chomsky's (1959) arguments against associationist models of language relied on the claim that they did not have the ability to model the hierarchical and recursive nature of human language syntax. His argument focused on linguistic competence rather than the details of performance. Similarly, Newell, Shaw and Simon's (1958b) Logic Theorist was an advance because it demonstrated that a computational mechanism, search through a problem space, could prove the same logic

theorems that humans could. The degree of match to human data, quantitative or otherwise, was far less important than the demonstration that a certain class of mechanism could explain some kinds of human problem solving. Finally, back propagation in neural networks (Rumelhart, Hinton, & Williams, 1986; Werbos, 1974) was viewed as a significant achievement not because it fit human data on learning patterns like the exclusive-or function, but because, counter to impressions generated by Minsky and Papert (1969), it demonstrated their ability to learn some of these functions.

The above efforts each caused a genuine revolution within cognitive science because they helped advance the ability of formal or computational methods to explain human behavior. Although none of these efforts endeavored to fit detailed data initially, each resulted in a framework that enabled more precise accounts of specific behavior and ultimately led to empirically rich bodies of research. These efforts suggest that, when faced with a choice between increasing a modeling framework's cognitive abilities and improving fits against already-modeled phenomena, there are clear benefits if some researchers choose to work on ability without immediate concern for quantitative model fits.

As a cautionary tale regarding the dangers of narrowly characterizing ability, we consider the history of computer chess playing. Researchers as early as Turing (1946) believed that chess playing would be a good demonstration of the power of computational accounts of human intelligence. Early efforts at successfully programming computers to play chess aimed to give them the ability to win games and were not concerned with fitting detailed human performance data. Simon's boast (recounted in (Crevier, 1993)) about progress in chess was that "a computer would be chess champion of the world within ten years" and not that say, high-quality eye-

movement predictions would be soon possible. The reason for this was that the question at the time pertained to how any mechanical process could produce intelligent behavior such as chess playing. The first approach to produce good results was heuristic lookahead search (Newell, Shaw, & Simon, 1958a). This constituted progress at the time because it showed that computational processes could lead to some forms of intelligent behavior (and thus an advance according to the ability criterion) and because in fact people actually carry out some lookahead search when playing chess.

Subsequent computational investigations into chess focused almost exclusively on improving the chess rating of computer chess programs (one measure of ability). The general approach was to exploit programming improvements and growing computational power to increase the numbers of possible moves a computer could explore. The result was that computer chess programs matched, and in many cases exceeded, human ability but did so by performing inhuman amounts of lookahead.

It is common to conclude from the history of chess research that ability is a flawed criterion for cognitive models. There are, however, three problems with this conclusion. First, even when most aspects of a model are implausible, some of them may be similar in some way to actual cognitive mechanisms. For example, although humans do not perform "brute-force" search, there is significant evidence (Dingeman, 1978), e.g., from verbal protocols, that they do perform some search. Further, much work into human chess playing has investigated specific heuristics people use to perform search.

This history is consistent with the sequence, discussed in the last section, from modeling ability to modeling specific mechanistic details. Although chess programs have from the

beginning generally used implausible amounts of search, they did reflect the fact that humans used some search and have established the framework within which many aspects of human chess playing have been understood.

Second, models can help explain human intelligence, even when they do not use *any* mechanisms that it is implausible to believe humans have. Some cognitive modeling research efforts do not specifically address mechanisms at all. For example, creators of many Bayesian cognitive models use stochastic simulation mechanisms such as Gibbs Sampling(Geman & Geman, 1984), which require implausibly large numbers of simulations of an event, because they are not attempting to model the mechanisms of human cognition, but instead to identify the constraints or knowledge those mechanisms are using. In human chess playing research, for example, it is common to investigate how humans formulate aspects of a game (Dingeman, 1978). One could imagine using search mechanisms quite different from human search to show that certain representations of chess playing yield to human-like playing patterns and use this as evidence that humans use those representations. These examples illustrate how models whose mechanisms are not faithful to human cognition can nevertheless help explain aspects of it.

A third problem with using chess research as an argument against ability concerns the sense of ability being considered. Human beings are not only able to play chess, but they are also able to learn the rules of chess, speak about their chess playing strategies, adapt to changes in rules and play many other games. The algorithms used in computer chess today do not have any of these capabilities and thus, when ability is construed to include them, they are not only failures under the fidelity criterion, but also according to the ability criterion. Thus, as we discuss in the next section, when breadth and not just level of ability is considered, existing

chess-playing systems are not a good counterexample to the importance of ability as a criterion, because they are lacking in (some important aspects of) ability.

3.2. Breadth and Parsimony through Unifications

As just mentioned, one of the characteristics of human intelligence we wish to explain is how it is able to succeed in such a broad array of situations. We further wish to do so with a single theory that posits as few mechanisms as possible. There are several advantages to explaining a set of phenomena with one theory rather than many. First, so long as the single account is internally consistent, one can be confident that its explanation of those phenomena is consistent, whereas explaining them with many theories may rely on inconsistent assumptions. Thus, when Newton provided one theory that explained the phenomena covered by Galileo's law of uniform acceleration and Kepler's laws of planetary motion, he demonstrated that those accounts were consistent with one another.

Theory unifications are important scientific contributions because they serve Occam's razor by increasing the ratio of phenomena explained to theoretical principles posited. Several important achievements in the history of science have involved unifications. Newton's three laws and one gravitational force subsumed Kepler's laws of planetary motion and Galileo's mechanics, providing a unified account for a variety of phenomena. Others immediately recognized it as an important achievement because of the unification itself rather than any new empirical results.

Unification is particularly important in cognitive science for several reasons. Newell (1990) lists several. Pertaining specifically to higher-order cognition is the fact that much of the

progress of science has been to provide naturalistic explanations of phenomena previously explained by vital or goal-directed forces. For example, the theory of evolution and the economic theory of markets both show how globally purposeful behavior can emerge from local interactions. Biology has shown how much of what was once attributed to vital forces can be explained through physical and chemical interactions. To the extent that cognitive modelers are successful, cognition would be accounted for using principles that are as un-goal-directed as, for example, gravitational or electromagnetic forces and thus unified with the rest of our understanding of the natural world. In the case of higher-order cognition, which superficially seems to be governed by principles so different from those regulating such physical forces, the unification would be especially dramatic.

Further, although human cognitive mechanisms are likely to be to some extent heterogonous, the generality of cognition implies that that there must be common elements (and corresponding theoretical unifications) in many forms of cognition. Since entities such as automobiles, parliaments and interest rates did not exist when humans cognition evolved, the mechanisms used to reason about them must be the same as those used to reason about aspects of the world that humans did evolved to deal with, for example physical and social events and relations. Thus, many of the same principles must govern cognition in all of these domains.

The history of cognitive science confirms the importance of unifications. Several achievements in cognitive science have been significant primarily because they unified previous results rather than because they predicted or explained new phenomena. Examples include Soar's (Laird, Newell, & Rosenbloom, 1987) account of many different weak methods in terms of problem-space search and impasses, the REM model (Shiffrin & Steyvers, 1997) of multiple

memory phenomena in terms of the probability of storage errors and Chomsky's (1981) unification of constraints in transformational grammar under a small set of principles and parameters.

Cognitive architectures (Newell, 1990) have been a particularly important contribution to breadth and parsimony in theories of higher-order cognition. Key abilities that support higher-order cognition include carrying out multi-step reasoning, solving novel problems, conversing about many topics and learning complex structures. A cognitive architecture provides a theory of the memories, representational formalisms and mental processes that are invariant in human cognition across domains and that enable adaptation to new domains. Thus, cognitive architectures are particularly well-suited for implementing theories about the generality and adaptability of human cognition.

Cognitive architectures are further important because, like other kinds of theory unification, they can increase the impact of individual pieces of work. The broader a theory's coverage of a phenomena, the greater the number of potential ramifications for individual efforts that elaborate on it. For instance, Hamilton's (1833) elaboration of Newtonian mechanics had a wider impact than if it had been merely an elaboration of Galileo or Kepler's theory. In cognitive science, when the mechanisms of an architecture account for multiple phenomena, revisions in those mechanisms can have an impact on theories of each of those phenomena. The more phenomena modeled within an architecture, the broader the potential explanatory value of revisions or extensions to it. For example, since ACT-R's production system and theory of memory are used in models of analogy, sentence processing and problem solving, revisions of

those systems will alter our understanding of all of those processes and will also be constrained by what is already known about them.

These reflections on breadth and parsimony suggest that, although there are benefits to refining models of specific phenomena, there is also great value in creating computational frameworks that provide unified accounts for a wide array of cognitive activities.

To summarize, the history of cognitive science and other disciplines suggests there are many ways to evaluate the field's progress towards explanations of higher-order cognition. Hypothesis testing and model fits are appropriate in some cases for gauging how accurately a model approximates particular aspects of cognition, but they do not measure whether a model or architecture has the power to explain the breadth and complexity of the human mind. Quantitative model fits measure success on only one criterion: how faithful a model is to reality. Other important issues concern whether a theoretical framework has the basic ability to predict observed phenomena and the range of phenomena it can cover with a small number of principles. Criteria such as ability, breadth and parsimony therefore have a crucial role to play in evaluating candidate theories of higher-order cognition.

4. A modern example of model comparison

We have shown how efforts to increase the ability, breadth and parsimony of computational approaches to modeling human higher-order cognition have played seminal roles in the development of cognitive science. In this section, we provide examples of model

comparison in the domain of language. Rather than selecting a "winner", our goal in making these comparisons is to demonstrate how these criteria can play an important role in modern cognitive science, how progress on them can be measured quantitatively and how they can be used to evaluate cognitive models.

4.1. Evaluating models of language use

Language use is a quintessential example of higher-order cognition. It has been studied in many subfields of cognitive science (most notably in artificial intelligence, cognitive psychology and, of course, linguistics) with very different methodologies. We will use three specific research efforts based on different methodologies to illustrate how the criteria of ability, breadth and parsimony can be used to compare and evaluate cognitive models. We will first briefly describe each model and discuss its contribution along these dimensions. We will then discuss the tradeoffs among the approaches and use them to illustrate the relationship between artificial intelligence and cognitive science.

4.1.1. The limited capacity model

We will first consider Lewis and Vasishth's (Lewis & Vasishth, 2005) sentence processing model, which we refer to as the "limited capacity model". It attempts to explain how humans can quickly infer the complex grammatical structure of sentences despite verbal memory that does not include information about the order of words and a severely limited focus of attention. Since order is an essential part of a sentence's syntactic structure, it is surprising that there is little evidence for explicit encoding of word order in short-term memory.

The limited capacity model accounts for the fact that human sentence processing conforms to order constraints in syntax (e.g., that subjects normally precede verbs in English) without any explicit representation of order. These can be illustrated using the sentence "Mary knew John appreciated her gift." When "appreciated" is focused on, the model attempts to retrieve a noun phrase subject. Since there is no explicit order information in memory, "Mary" is not immediately ruled out for retrieval and the subject can predict that there will be subsequent noun phrases (i.e., "the gift) in the sentence. However, three factors favor "John" being retrieved from among the other noun phrases in the sentence. First, "John" is remembered when "appreciated" and "her gift" are perceived. Thus, the fact that memories are always in the past implicitly leads to order constraints in processing. Second, Mary is already encoded as being the subject by "knew" and thus cannot also be the subject of "appreciated" (because, in part, it was the only previous noun phrase when "knew" was perceived). Finally, memory decays through time so that, other things being equal, more recent items (i.e., "John") are more likely to be retrieved than less recent items ("Mary"). Factors such as these explain how sentence processing conforms to syntactic ordering constraints without explicitly representing order in memory.

Lewis and Vasishth evaluate their model with fits to reading times. Although these fits are impressive and many aspects of the model are original, their dependent measures and methods of fitting reading data are standard. We now consider how the model contributes to ability, breadth and parsimony.

By explaining sentence processing effects in terms of memory and attention mechanisms with no explicit syntactic representations, the limited capacity model extends the verbal memory and attention literature's breadth of impact. It also contributes to parsimony of cognitive theory

by reducing the need to posit separate mechanisms for syntactic and other forms of processing. Also, since the model is created within a cognitive architecture, ACT-R, it expands the explanatory scope of that architecture and thus brings more unity to the field insofar as ACT-R can explain a wide variety of phenomena. These contributions would all have been significant with less accurate model fits or even just qualitative predictions.

Regarding ability, Lewis and Vasishth concede that there are several kinds of syntactic constructions, such as "self-embeddings", that their model does not parse, e.g., "The rat the cat ate died." This is consistent with the fact that people often find such sentences difficult to parse, but it does not reflect the fact that people nevertheless often can overcome this difficulty and parse these sentences. More generally, modern, wide-coverage syntactic theories in formal linguistics involve formal structures and operations (such as type hierarchies with default inheritance (Pollard & Sag, 1994) and empty categories (Radford, 1997)) that are not included in the limited capacity model. Without such mechanisms or structures, it remains to be seen whether the assumptions of the model are able to achieve broad grammatical coverage.

Although these remarks do not make use of any mathematical or statistical methods, they nevertheless illustrate how ability, breadth and parsimony can be discussed with some precision. The list of grammatical constructions the limited capacity model explains (a measure of breadth) can be specifically enumerated and the precise number of mechanisms the model shares with other ACT-R models (a measure of parsimony) is easy to determine by inspection. This is an example of how the precision generated by computationally instantiating a theory of cognitive architecture makes work within that framework easier to discuss and evaluate.

4.1.2. The corpus-driven approach

The corpus-driven approaches into language that we discuss here would not typically be considered cognitive modeling, although we will argue that, on the basis of our criteria, it makes a significant contribution to our understanding of language use. A central goal of this work is to avoid the difficulties of hand-crafting a broad-coverage human language grammar by automatically learning grammatical rules from an annotated corpus. Hand-crafting grammars has been difficult because of the large number of grammatical constructions that must be covered and the many exceptions to these constructions.

This work relies heavily on annotated corpora. The existence of corpora such as the Penn Tree Bank (Marcus, Santorini, & Marcinkiewicz, 1994) that include the syntactic structure for thousands of sentences has led to much activity (summarized in (Lease, Charniak, Johnson, & McClosky, 2006)) in artificial intelligence and computational linguistics. The Penn Treebank includes sentences from sources (such as the Wall Street Journal) paired with their grammatical structures. Given this corpus, researchers design algorithms that attempt to infer the probabilistic context-free grammar that generated the sentences. A probabilistic context-free grammar rule is a context-free rule associated with a conditional probability. For example, S → (.3) NP VP is interpreted as asserting that when a sentence (S) occurs, it is in 30% of cases generated by a noun phrase (NP) followed by a verb phrase (VP). By combining the rules that generated a parse of a sentence, one can compute the probability of that parse having been generated. Parsers for probabilistic context-free grammars output the most likely parse. Learning methods are often evaluated by the "precision" and "recall" of the grammars they produce. Precision is the proportion of phrases generated by the learned grammar that exist in the corpus and recall is the

proportion of phrases in the corpus that the learned grammar generates. Precision and recall rates in the mid-90% range have been reported.

These results have had a significant impact on the field. This can be illustrated with the "prepositional phrase attachment problem". For example, the sentence "I saw the bird with the telescope" has multiple possible syntactic structures. In one, "with the telescope" is a prepositional phrase that modifies "the bird", implying that the bird has the telescope. In another, "with the telescope" modifies "see", implying that the telescope was used to see the bird. Resolving these "attachment ambiguities" seems to require reasoning about the world (e.g., that birds do not typically use telescopes") in addition to purely syntactic processing. That the corpus-driven approach resolves a surprisingly high proportion of (though by no means all) attachment ambiguities, together with other successes using corpora, has been counted as evidence that statistical or "distributional" information is much more potent than has been implied by those arguing (Chomsky, 1976) that the language children hear is not sufficient for them to learn grammatical structure and thus significant aspects of syntax must be innate.

This work contributes to the breadth and parsimony of cognitive theory in several ways. The corpus-driven approach can potentially increase the parsimony of theories of language development and use because it reduces (though not necessarily eliminates) the need to posit learning and sentence processing mechanisms beyond those involved in processing statistical information. Although most learning algorithms operate over all sentences at once (as opposed to incrementally, as in human language learning), they do demonstrate the power of the statistical information latent in human language use. This work also contributes to breadth because of the wide range of sentence types and corpora upon which it has shown success.

Sentence processing models in psycholinguistics and syntactic theories from formal syntax cannot claim to correctly identify large fractions of phrases in large-scale corpora. Finally, this work also advances the ability of computational models of language use since there were previously no known computational methods (in psycholinguistics or computational linguistics) for inducing such wide-coverage grammars.

Since precision and recall are measures of ability and since the number of sentences and corpora used reflects breadth, this work illustrates that there are contexts were breadth and ability can be precisely quantified and enable price model comparisons. A review (Lease et al., 2006) of some past research in this field lists several approaches that make clear claims about their superiority over previous methods based on the improvements in precision and recall they generate.

All these contributions have been made despite the fact that most of the algorithms used in this research would fail entirely on many of the eye-tracking or reading time tests often used to evaluate theories of sentence processing. More out of convenience than necessity, these algorithms parse sentences in parallel, bottom-up manner (i.e., by considering all words simultaneously and computing the phrases they can form) rather than in the incremental order in which humans process them.

Two criticisms of this approach concern ability. First, most current corpora, including the Penn Treebank, are based on context-free, or even less powerful, grammars. Although the relative simplicity of these grammars makes them more amenable to statistical methods, there are many regularities (for example, gender agreement and head/argument ordering uniformities) in language that context-free grammars do not naturally capture. Second, there are many aspects of

sentence processing (and language use in general) that cannot be studied in corpora as they are currently constituted. For example, consider the following two sentences. "The couple went for a walk. He held her hand." "he" and "she" clearly refer to the male and female members of the couple respectively. But existing corpora only encode coreference between words in the corpus. Thus, since the actual antecedents of "he" and "she" are not written or spoken and only inferred, statistical inferences made on purely what is written or said cannot determine these coreference relationships. Further, ellipsis provides an example in which the corpus includes the antecedent but the reference to it is not actually spoken. For example, in "Mary likes ice cream and so does John", it is clear that John likes ice cream. However, since John liking ice cream is not specifically mentioned in the corpus or its annotation, we cannot test whether a parser associates the appropriate action with "so does John". This makes ellipsis very difficult to study within a corpus. Of course, these are problems with the corpus-based approach as it is today. They do not preclude richer corpora from being developed to study such phenomena.

4.1.3. Mapping syntactic structure onto physical structure

The final research effort we study attempts to show how syntactic structure can be mapped onto the structure of physical reasoning problems so that a physical reasoning model can thereby process sentences (Cassimatis, 2004; Murugesan & Cassimatis, 2006). There are several reasons to attempt to map syntactic structures to physical structures. The concepts involved in syntax (e.g., anaphora, bindings, gaps) superficially appear much different from the concepts used in the physical reasoning. Like a similar mapping found between social and physical cognition (Bello, Bignoli, & Cassimatis, 2007), finding a mapping between these two domains

would make it more plausible that other surprising mappings between domains exist and that architectural approaches to cognitive modeling based on small numbers of principles and mechanisms can have wide explanatory breadth. Also, as we describe below, these mappings have a direct impact on views about the modularity, learnability and innateness of language.

The mapping between language and physical structures is based on the fact that verbal utterances[1] are actions taken by people. Like physical actions, verbal utterances occur over temporal intervals, belong to categories, combine into larger actions and are normally taken to achieve some goal. This mapping enables the most powerful grammatical theories to be reformulated using representations for physical actions and events. To formally confirm that

[1] We consider spoken utterance here. Text and other forms of nonspoken language can be mapped onto hypothetical spoken utterances.

such mappings are possible, Cassimatis and Murugesan used Head-Driven Phrase Structure Grammar (HPSG) (Pollard & Sag, 1994). HPSG was used because its coverage is competitive with other major theories while being very amenable to computational implementation. The following examples illustrate how structures used by HPSG can be represented using physical structures.

Constituency and linear order as parthood and time. Almost all grammars contain some rules such as: S → NP + VP. In physical terms, we represent this using part, temporal and category relationships. For example, this rule says that an action of category noun phrase utterance followed immediately by an action of category verb phrase utterance can combine to form an action of category sentence utterance.

Feature unification and identity. HPSG and many other grammars use feature information, for example, to say that the gender of a verb and its subject should agree. Features of physical objects must also "agree". For example, the gun should be loaded with cartridges of the same caliber as the gun and the octane of gas put into the car should be the same as the octane of the gas that the car requires. Agreement in both the physical and verbal world can be thought of as an identity relation between features.

Type hierarchies. Many grammars rely (heavily in the case of HPSG) on hierarchies of categories . These also exist in the physical world (e.g., iron objects are metal objects) and which physical laws apply to an object depends on its category.

Co-reference. The fact that two phrases share the same reference (e.g., as with "Mary" and "herself" in "Mary likes herself") can be encoded as an identity relation. In this case, the reference (R1) of one phrase is said to be identical to the reference (R2) of another phrase: *R1 =*

R2. Identity relations and the need to resolve identity ambiguity are an important aspect of physical reasoning, as when one must infer whether an object that has emerged from an occluder is the same object one saw go behind the occluder or merely a different object with the same appearance.

Figure 1 illustrates this mapping. It depicts a parse tree for a sentence where each element of the parse is encoded using physical relation.

----------Insert Figure 1 about here ----------

Even seemingly obscure and language-specific constraints can be easily reformulated using this mapping between syntactic and physical structures. For example, Radford (1997) formulates "the c-command condition on binding" thus: *A bound constituent must be c-commanded by an appropriate antecedent*. He defines c-command by stating that *a node X c-*

. Since c-command is a constituency relationship, it can be reformulated using the parthood relation thus:

X c-commands

This mapping has several consequences. First, since (as mentioned earlier) human syntax and the physical relations used in the mapping are so superficially different, the possibility of finding other mappings between dissimilar domains increases. Second, language is often neglected (sometimes explicitly (Newell, 1990)), in cognitive architecture research and thought by many (e.g., (Chomsky, 1976)) to be a module "encapsulated" from the rest of cognition. The syntax mapping raises the possibility that human language processing involves the same mechanisms studied in other domains. At a minimum, the mapping demonstrates that domain-general mechanisms can generate and process language whose structure is as rich as human syntax. The mapping therefore reduces the need to posit a special language faculty or module to explain human verbal behavior. Finally, many arguments (e.g., Laurence & Margolis, 2001; Lidz & Waxman, 2004) for the innateness of language are based on the presumption (Chomsky, 1980) (called the *poverty of the stimulus)* that children have too little verbal input to infer the grammatical structure of English on their own. However, if language processing shares many or all of the mechanisms of physical reasoning, then the amount of experience children can bring to bear on language development includes their physical interaction with the world and is therefore much larger, thus reducing the potency of poverty of the stimulus arguments.

This mapping primarily makes contributions to ability and unity. First, it enables a model of syntactic parsing to be built using the same mechanisms as an updated version of a model of physical reasoning, the mapping clearly demonstrates that physical reasoning mechanisms (with only the addition of a representation of the sound of words) are sufficient to explain syntactic parsing. Second, because computational methods in AI and cognitive modeling for processing

syntax and those for processing other forms of information have often been so different and difficult to integrate, the mapping increases the ability of computational methods to integrate pragmatic, semantic and syntactic information during parsing. However, the mapping and the model based on it do not yet have the ability to learn a grammar even though it potentially has a role in explaining human language acquisition.

4.2. Tradeoffs and model evaluation methods

Having illustrated how the criteria of ability, breadth and parsimony can be used to evaluate the contribution of individual cognitive models, we now use these criteria to compare these models. Specifically, we will show that even though the evaluation methods their creators used lead to certain tradeoffs and different research directions, their results can nevertheless inform each other's work. Many of the tradeoffs we discuss are not logically entailed, but since there are only finite resources for any research project, they often do become practical necessities in the short-term.

Not being bound by the necessity to fit quantitative data on sentence processing has enabled the mapping model to create more grammatical coverage and has allowed the corpus-driven model to provide insight into language learning over a very wide range of corpora. The mapping approach presupposed a temporal reasoning capacity, which is not as well characterized psychometrically as the aspects of verbal memory and attention that the limited capacity model assumes. To have first collected the relevant psychometric data would have required a large effort that would have made the project infeasible. Likewise, although there is a considerable body of work on language acquisition, the measures, methods, subject pools and often even the

results vary so dramatically that it would have been impractical to attempt to regularize all the data and then relate it to the corpora used to test grammar learning algorithms. Of course, one consequence of accepting these tradeoffs is that, unlike the limited capacity model, the mapping model and corpus-driven approach do not make detailed claims about the specific mechanisms involved in sentence processing.

The mapping model and the corpus-driven approach illustrate a tradeoff between different kinds of abilities. Since the mapping model involves a more expressive grammatical formalism than the grammars used in most corpus-driven work, it can capture more of the deep, cross-linguistic regularities in grammar. Further, the mapping from grammatical to physical structure enables reasoning about nonlinguistic items to constrain sentence processing, while context-free grammars (which capture only category, order and parthood relations) are ill-suited for such nonlinguistic reasoning (which involves many other kinds of relations). Statistical learning methods for more complex grammatical formalisms are much less-well understood and more difficult to implement. Further, a complex grammar would require better-trained human corpus-annotators and much more of their time. In the short-term, it would therefore be difficult (or at least prohibitively expensive) for the mapping approach to achieve the breadth, precision and recall rates of the corpus-driven approach. There is a thus a tradeoff between the corpus-driven approach's ability to learn and the mapping model's ability to explain deep linguistic regularities and the integration of syntactic and other forms of information in processing.

That choosing a certain methodology leads these efforts to certain tradeoffs in the short-term does not preclude each research program from benefitting considerably from the others. For example, ACT-R's memory retrieval system operates in part on the basis of statistical

information from past experience. If the operation of this subsystem could be meaningfully related to the statistical methods used to induce probabilistic context-free grammars in the corpus-driven work, then the ACT-R model would gain significant language learning abilities while the corpus learning approach would be constrained by and perhaps exploit insights from what is known about human learning and memory. The mapping model suggests directions for both research programs. For example, since the mapping relies heavily on temporal and identity relations, work incorporating methods for learning these relations could potentially increase the ability of the other two approaches to deal with more complex grammars. Such work would in turn add a learning ability now lacking in the mapping model. We believe that these potential synergies between research efforts would happen, not in spite of the fact that each research program does not subject itself to the others' evaluation regimes, but because ignoring certain constraints in the short-term frees each approach to develop certain ideas in sufficient depth to make them broadly relevant.

4.3. Cognitive modeling and artificial intelligence

Our example model comparisons illustrate that, with regard to ability, parsimony and breadth, artificial intelligence and cognitive modeling are highly interrelated endeavors.

It is clear that advances in computational models of higher-order cognition can be significant contributions to artificial intelligence research. As mentioned earlier, one of the greatest challenges to creating such models is that there are no known computational methods that exhibit many aspects of human higher-order intelligence. Finding such methods is one of

the goals of research in artificial intelligence. An advance in the abilities of cognitive models of higher-order intelligence is thus also a contribution to artificial intelligence.

Further, work that increases the power of known computational methods can also be of interest to cognitive modeling, even when it does not aim to do so. For example, in the early literature on the corpus-based approaches (typically associated with artificial intelligence or computational linguistics rather than cognitive modeling), there was little discussion of the actual mechanisms used by people in language understanding. However, as outlined in the last section, this work had a significant impact in our understanding of and research into human language use and development because it demonstrated that statistical information had more grammatical information latent within it than many had previously suspected. As another example, even though the early work into chess-playing machines made few if any attempts to relate the mechanisms they used to actual human mechanisms, our discussion of chess showed that this work nevertheless did contribute meaningfully, in part by establishing a framework within which to ask research questions, to subsequent studies of human chess playing.

Examples such as these illustrate that when model evaluation is broadened beyond model fits to include ability, breadth and parsimony, results in artificial intelligence (operationally defined, for example, by the journals and conferences to which it is disseminated) are often in fact a contribution to our understanding of human cognition. It also illustrates that, as in the early chess and corpus-drive language research, systems that use mechanisms that differ extensively from the mechanisms of human cognition and that do not precisely fit or predict empirical data (for example about reaction times and error rates) can make a significant contribution to our understanding of human cognition.

5. Conclusions

Computational models play several roles in explaining higher-order cognition in humans and consequently there are several different ways to evaluate them. Researchers would like to know whether a model posits mechanisms that (a) at least approximate those which implement human cognition and (b) are capable enough to explain the broad range of human cognitive abilities. They should also prefer alternatives that are parsimonious and that provide a unified account of these phenomena. Although ability, parsimony and breadth are very difficult to define formally and measure precisely in general, we have demonstrated that, in specific contexts, it is possible to precisely characterize the contribution of a research result to achieving these ends.

We have argued that quantitative model fits are only one of many activities that can contribute to a computational understanding of higher-order cognition. Our discussion about ability, parsimony and breadth, together with the example model comparisons, suggest a number of approaches to evaluating and driving progress in cognitive models for higher-order cognition.

Breadth and cognitive architectures. Cognitive architectures are theories of structures and processes that are invariant across much or all of human cognition. When an architecture is used to model cognition in a new domain, the breadth and parsimony of cognitive theory is extended because more phenomena are explained using the same set of mechanisms. As the next point amplifies, these benefits are increased to the extent that multiple models within an architectural framework make the same assumptions.

Parsimony through a single model. One aspect of higher-order cognition that is especially difficult to replicate computationally is people's ability to function in a wide variety of situations. Most computational frameworks become intractable as the amount of world knowledge becomes larger. The common practice of modeling behavior only on one specific task does not demonstrate that an architecture scales as needed. By evaluating a framework's ability to support a single model that operates across a broad range of tasks, we can assure that research within that framework does not evade the difficult computational challenges of higher-order cognition. This research agenda would move the field toward theories that reproduce the broad functionality and adaptability observed in humans.

Increasing ability. A model that is unable to reason, solve problems, converse and learn in situations where humans clearly are able to do so is an as-of-yet incomplete theory of cognition. Improving a model's cognitive abilities is thus an important step towards developing it into a comprehensive and unified account of higher-order cognition. There is no single approach to evaluating ability, but once it has been recognized as an important goal, it is often straightforward to measure. Examples of precise characterizations of ability mentioned in this paper include Chomsky's language hierarchy, chess ratings, precision and recall rates in corpus parsing and formal demonstrations that one set of concepts (e.g., those involved in physical reasoning) are sufficient to characterize the structure of other domains (e.g., the syntax of human language).

Unity through mappings. Theories of the cognitive architecture posit that certain basic mechanisms are involved throughout the range of human cognition and thus, at least implicitly, assume that there are many common underlying structures to domains in which people operate.

Finding mappings between the structure of domains supports claims for parsimony and breadth made by a particular architecture, but can also play an important architecture-independent role. For example, as more domains whose structures are mapped to those described above (e.g., time, identity, parts and categories), any cognitive model that explains reasoning with those structures gains in explanatory power.

Artificial intelligence. It is common to characterize models that do not quantitatively fit data as being "AI" and not "Cognitive Science". We have argued that this distinction is historically inaccurate (Langley, 2006) and that both fields today, especially insofar as ability is a priority, have overlapping goals. Cognitive modelers need to develop computational methods with more ability, which is also the goal of artificial intelligence. Further, our discussion of the contributions of the corpus driven approach show that, even when work in artificial intelligence is conducted without attention to psychological constraints, it can still significantly advance our understanding of human cognition.

Both the history of cognitive science and ongoing research demonstrate that concentrating on ability, breadth and parsimony can generate results which have significant implications for the field. These include issues such as how language interacts with the rest of cognition and how people learn grammar. Many more such outstanding questions regard how various forms of learning integrate with each other and with reasoning, how people are often able to retrieve relevant facts and knowledge from a very large memory store in less than a second and how emotion interacts with reasoning and problem solving. We believe that the results to date demonstrate that the approaches to evaluating computational models described herein can help drive progress towards theoretical frameworks with greater ability, parsimony and breadth,

and therefore lead to significant progress on many challenging and important questions in cognitive science.

References

Anderson, J., & Lebiere, C. (2003). The Newell test for a theory of cognition. *Behavioral and Brain Sciences*, 587-601.

Bello, P., Bignoli, P., & Cassimatis, N. L. (2007). *Attention and Association Explain the Emergence of Reasoning About False Belief in Young Children.* Paper presented at the 8th International Conference on Cognitive Modeling, Ann Arbor, MI.

Braine, M. D. S., & O'Brien, D. P. (1998). *Mental logic*. Mahwah, NJ: Lawrence Erlbaum Associates.

Cassimatis, N. L. (2004). *Grammatical Processing Using the Mechanisms of Physical Inferences.* Paper presented at the Twentieth-Sixth Annual Conference of the Cognitive Science Society.

Chomsky, N. (1959). A review of Skinner's "Verbal Behavior". *Language, 35*, 26-58.

Chomsky, N. (1976). On the nature of language. In S. R. Harnad, H. D. Steklis & J. Lancaster (Eds.), *Annals of the New York Academy of Sciences* (Vol. 280, pp. 46-57).

Chomsky, N. (1980). *Rules and representations*. Oxford: Basil Blackwell.

Chomsky, N. (1981). *Lectures on Government and Binding*. Holland: Foris Publications.

Crevier, D. (1993). AI: The Tumultuous History of the Search for Artificial Intelligence. New York: Basic Books.

Dingeman, A. (1978). *Thought and choice in chess*: Walter de Gruyter.

Geman, S., & Geman, D. (1984). Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 7*, 721–741.

Hamilton, W. R. (1833). Introductory Lecture on Astronomy. *Dublin University Review and Quarterly Magazine, 1*.

Johnson-Laird, P. (1983). *Mental models*. Cambridge, MA: Harvard Unversity Press.

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

Langley, P. (2006). *Intelligent Behavior in Humans and Machines*: Computational Learning Laboratory, CSLI, Stanford Universityo. Document Number)

Laurence, S., & Margolis, E. (2001). The Poverty of the Stimulus Argument. *The British Journal for the Philosophy of Science, 52*(2), 217-276.

Lease, M., Charniak, E., Johnson, M., & McClosky, D. (2006). *A Look at Parsing and Its Applications.* Paper presented at the American Association for Artificial Intelligence

Lewis, R. L., & Vasishth, S. (2005). An activation-based model of sentence processing as skilled memory retrieval. *Cognitive Science, 29*, 375-419.

Lidz, J., & Waxman, S. (2004). Reaffirming the Poverty of the Stimulus Argument: A Reply to the Replies. *Cognition, 93*(2), 157-165.

Marcus, M. P., Santorini, B., & Marcinkiewicz, M. A. (1994). Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics, 19*(2), 313-330.

McClelland, J. L., & Patterson, K. (2002). Rules or Connections in Past-Tense inflections: What does the evidence rule out? *Trends in Cognitive Sciences, 6*(11), 465-472.

Minsky, M. L., & Pappert, S. A. (1969). *Perceptrons*: MIT Press.

Murugesan, A., & Cassimatis, N. L. (2006). *A Model of Syntactic Parsing Based on Domain-General Cognitive Mechanisms.* Paper presented at the 8th Annual Conference of the Cognitive Science Society, Vancouver, Canada.

Newell, A. (1973). You can't play *20 questions* with nature and win. In W. G. Chase (Ed.), *Visual Information Processing*: Academic Press.

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

Newell, A., Shaw, J. C., & Simon, H. A. (1958a). Chess-playing programs and the problem of complexity. . *IBM Journal of Research and Development, 2*, 320-335.

Newell, A., Shaw, J. C., & Simon, H. A. (1958b). Elements of a theory of human problem solving. *Psychological Review, 65*, 151-166.

Pinker, S., & Ullman, M. T. (2002). The past and future of the past tense. *Trends in Cognitive Sciences, 6*, 456-463.

Pollard, C., & Sag, I. A. (1994). *Head-Driven Phrase Structure Grammar*. Chicago, IL: University of Chicago Press and Publications.

Radford, A. (1997). Syntactic theory and the structure of English: A minimalist approach. Cambridge: Cambridge University Press.

Rips, L. J. (1994). The psychology of proof: Deduction in human thinking. Cambridge, MA: MIT Press.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature, 323*, 533-536.

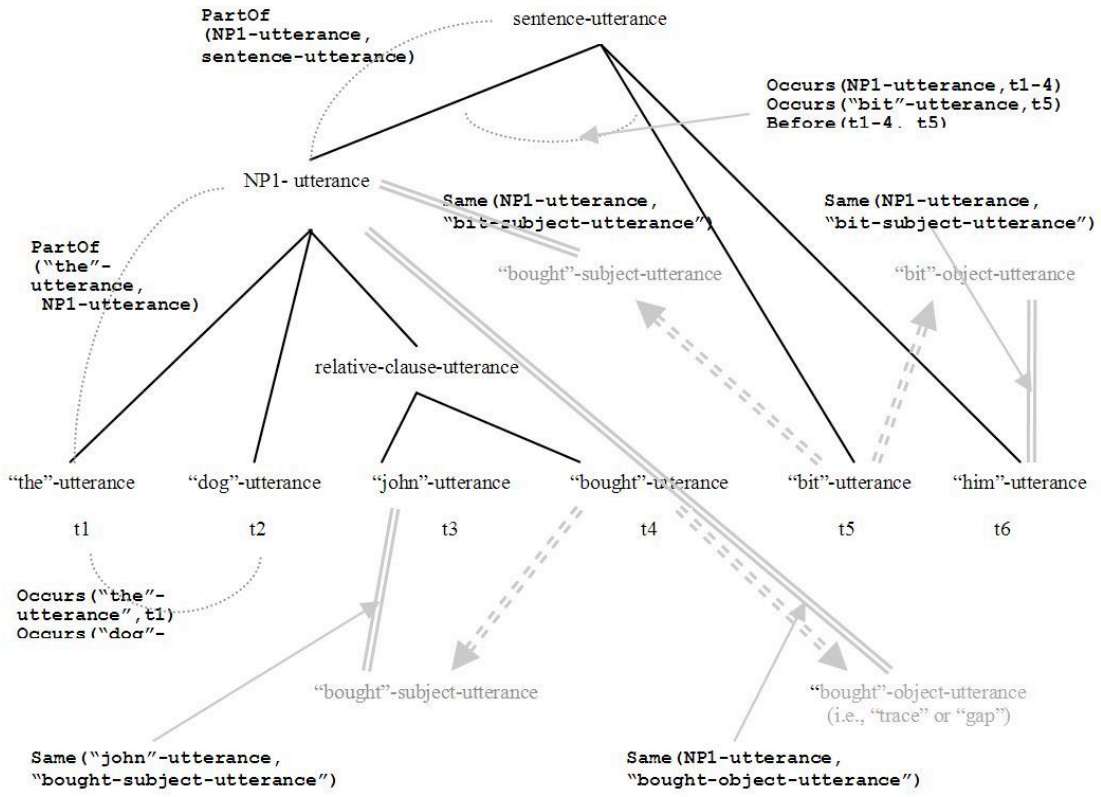
Shiffrin, R. M., & Steyvers, M. (1997). A model for recognition memory: REM: Retrieving Effectively from Memory. *Psychonomic Bulletin & Review, 4*(2), 145-166.

Turing, A. M. (1946). *Proposed electronic calculator*: National Physical Laboratory, Teddingtono. Document Number)

Werbos, P. (1974). *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences.* Unpublished doctoral dissertation, Harvard University, Cambridge, MA.

Figures

Figure 1. The syntactic structure of the sentence represented using relations from physical reasoning.



Experimental Studies of Integrated Cognitive Systems

Pat Langley
Computational Learning Laboratory
Center for the Study of Language & Information
Stanford University, Stanford, CA 94305 USA
langley@csli.stanford.edu

Elena Messina
National Institute of Standards & Technology
100 Bureau Drive, Mail Stop 8230
Gaithersburg, MD 20899 USA
elena.messina@nist.gov

Abstract | In this paper, we examine the issues that arise in the experimental study of integrated cognitive systems. We review the reasons why such artifacts are diŽcult to evaluate, then consider some dependent measures that can be used to characterize their behavior. Next we discuss independent variables that can in uence this behavior, in particular features of the domain and characteristics of the system, including its knowledge and experience. We then turn to domains and testbeds that support experiments with such systems, giving examples of some promising candidates. We conclude with a discussion of the scientiþc goals of experimentation, which involve understanding the mapping from domain and system characteristics onto behavior.

I. Introduction and Motivation

For more than a decade, research in artiþcial intelligence has relied on experimentation as a key element in evaluation. Machine learning was perhaps the þrst subdiscipline to adopt systematic experiments (e.g., Kibler & Langley, 1988), but their use has spread throughout the broader community (e.g., Cohen, 1995). Today, experiments are the primary means by which AI researchers evaluate their methods, and the experimental techniques as mature and well understood.

However, the experimental study of integrated cognitive systems is less well established and clearly needs more development. The reasons should be clear from the phrase itself, which re ects the nature of the intelligent artifacts being constructed. First, it is inherently more diŽcult to evaluate systems than component algorithms, since they are harder to construct and analyze. Second, it is more challenging to run experiments with cognitive systems, since they rely on complex, multi-step reasoning rather than simple classiþcation or reactive control. Finally, evaluating claims about integrated systems is problematic because it involves the examination of interactions among their components. Together, these distinctive factors have slowed the development of an experimental method for such complex entities.

In this paper, we propose an experimental framework that is appropriate for the study of integrated cognitive systems. In the next section, we discuss basic and higher-order dependent measures that can arise in such experiments. After this, we consider three main classes of independent factors that can in uence system behavior, then turn to domains and testbeds that would support the experimental evaluation of such systems. In closing, we discuss the broader scientiþc goals of experimentation, which aim not to show superiority but to identify reasons for observed behaviors.

II. Dependent Measures of System Behavior

As scientists, we are concerned with understanding the behavior of integrated cognitive systems, which in turn means that we require ways to observe and characterize this behavior. In this context, it is important to distinguish between between metrics and dependent measures. These terms are closely related, but the þrst is typically associated with prescriptive benchmarks that are used to determine one system's superiority other another, whereas the second is generally associated with systematic experiments that aim at scientiþc understanding. The comments that follow are relevant to both approaches to evaluation, but our focus here is on the latter, which we think is far more appropriate for the current stage of the þeld. We organize our treatment into three broad categories: basic measures, averaged metrics, and higher-order variables.

A. Basic Measures of System Behavior

The existing literature reports a variety of basic measures that are relevant to integrated cognitive systems. These provide the simplest ways to describe the observed behavior of an intelligent construct. We should clarify that behavior always occurs in the context of some task, whether provided externally or generated by the agent itself, and some situation, whether it involves the agent's physical environment or its mental state. We will refer to this context informally as the problem that the agent is attempting to solve.

Perhaps the most straightforward behavioral measure concerns whether the agent succeeds or fails at handling a given problem. For example, a cognitive system may prove or fail to prove a geometry theorem, it may or may not solve a novel puzzle, it may or may not deliver a package to a speciþed address, and it may win or lose a given game. This measure oPers only one bit of information, but it may still be valuable when combined with other results, as we will see shortly.

However, other problem-related measures provide more detail. One such metric is the eŽciency or speed with which the cognitive system handles a given problem.

For instance, one can count the number of states in a problem space considered during a geometry proof, the time it takes a UPS driver to deliver a package, and the number of moves until checkmate in a chess game. Such a dependent variable gives information about the cognitive or physical eŽciency with which the agent handles a particular problem.

Of course, some paths to success are more desirable than others, so we may also want to measure the quality of the cognitive system's solution to a problem. For example, a geometry proof may have few or many steps and thus be more or less elegant, a package deliverer may drive safely and politely or dangerously and impolitely on his way to an address, and a chess player may lose only a few unimportant pieces or many important ones in defeating an opponent. Metrics of this sort oþer details about the desirability of the cognitive agent's behavior in accomplishing a given task.

B. Combined Measures of Behavior

The þeld of statistics tells us we should not draw conclusions from individual cases, but rather that we should rely on multiple samples. We can then combine the results from these samples and calculate a more robust dependent variable. Taking the average of sampled measurements is the most common and obvious combination scheme, but calculating cumulative scores is another possibility. The important thing is that, by combining measures for diþerent samples, we can partly cancel out variation due to unknown or unavailable factors, and thus increase the chance of meaningful results.

Naturally, this approach requires some population from which to draw samples, typically diþerent problems from within a single domain, although sampling from across domains is also possible. For instance, we might present the cognitive system with diþerent geometry theorems to prove, ask it to deliver packages to distinct addresses or even in diþerent cities, and confront it with diþerent chess opponents or even chess-like games with alternative rules. The population from which one draws samples determines the generality of one's conclusions about the cognitive system's behavior. We may suspect that the agent can prove theorems not only in geometry but also in algebra, but sampling from the former domain provides no evidence for the latter. An empirical study should state clearly the population being sampled, ideally in formal terms but always in enough detail that others can replicate the sampling process.

We should note that combined measures of behavior oþer more than guards against unknown factors and random noise. This approach also lets one convert qualitative measures, such as success or failure on a problem, into quantitative ones, such as the percentage or total number of problems solved. This makes them especially useful for researchers who want to make claims about new functionality, which at þrst glance appear to involve only qualitative evidence, but which can be handled in quantitative terms with averaged, cumulative, or other combined measurements of system performance.

C. Higher-Order Measures of Behavior

Although combined measures guard against unknown in uences and oþer quantitative variables, they still present only a small window into often complex behavior. Metrics that average across domains improve the situation, since they provide information about a cognitive system's broader generality, but more sophisticated responses are certainly possible.

For instance, we might plot the dependent measure for a novel system against the same measure for a baseline or control system, with each point summarizing the two systems' behaviors on a distinct problem. We can then use regression to þt a line to the points, which gives both a slope and an intercept as higher-order measures. A positive intercept means the novel system does better than the control even on easy problems, whereas a slope greater than one means it scales to diŽculty better than the baseline system.

Another example, which we will discuss more later, involves learning curves, in which one plots a behavioral measure like eŽciency or quality against the number of training cases a learning system has encountered. Such curves typically have either an exponential or sigmoid shape, so that linear regression is not appropriate, but we can þt them with other parametric forms. These produce higher-order measures for the system's performance at the outset, its rate of improvement as a function of experience, and its asymptotic performance.

Both of these examples involve some form of variation, though this need not be systematic. In general, whenever one collects simple measures of a cognitive system's behavior under a number of distinct conditions, these can be used to calculate higher-order measures that summarize its behavioral characteristics across the conditions from which the samples were taken.

III. In uences on System Behavior

A scientiþc experiment should do more than measure a system's behavior under one or more condition. The goal of experimentation is to understand the factors that in uence the behavior, which means one should measure the dependent variables in multiple situations that diþer along some dimension. Such a factor is often referred to as an independent variable, since one can typically vary it independently of others. As with dependent measures, diþerent independent variables can reveal diþerent facets of the system under study. In this section, we examine three broad classes of controllable factors that are appropriate for the experimental evaluation of integrated cognitive systems.

A. Characteristics of the Task and Domain

One important type of independent variable concerns aspects of the problem domain and the tasks which occur within it. The simplest version of this idea involves collecting multiple samples for an experimental condition, which we have already discussed above. For studies with an intelligent system, this means running the system multiple times on diþerent problems from a domain, and then combining the results in some fashion. For this purpose, one draws sample tasks from some distribution over the problem domain. This may involve specifying a þxed set of problems or tasks, but another strategy involves creating a generator that can produce sample problems. In either case, one should state the relation between these samples and the broader class of problems over which one hopes to generalize.

An important variation on this idea involves running the system on problems from diþerent domains to ensure its generality. If we are interested in this central issue, then it is essential to demonstrate successful behavior not only across diþerent tasks within the same domain, but across a variety of distinct domains. For instance, most AI work on game playing has focused on a single game like chess, which Pell (1996) argues has produced systems that are optimized for that domain but do not demonstrate general intelligence. Instead, he deþned an entire class of chess-like games and developed a system that plays reasonably when given information about their board, pieces, and rules.

Such studies ensure generality, but they do not by themselves reveal the reasons for variations in system behavior. For this, we must examine the relation between problem diŽculty and response. We can order problems by the results they produce on some behavioral measure like problems solved or eŽciency of solutions, but this does not provide much insight. Ideally, one should vary experimentally the problem diŽculty and examine its eþects on system behavior. This in turn requires an analysis of the domain that suggests what factors in uence the diŽculty of problems.

Kibler and Langley (1998) provide an early domain analysis for machine learning. They propose a number of factors that aþect the diŽculty of induction tasks, including the complexity of the target concept, the number of irrelevant features, and the amount of noise in the training data. Their analysis focused on classiþcation, but they mention analogous diŽculty factors for other areas, such as the regularity of problem spaces and the structure of target grammars. One factor they overlooked was the rate of environmental change, which can pose a challenge for any learning system.

Studies that vary problem diŽculty typically rely on synthetic domains to control this factor, but Langley (1996) warns against their casual use. Synthetic problems give one þne-grained control over domain characteristics, which can let one determine how these factors in uence behavior. But one must be careful to ensure that these problems are suŽciently similar to ones which arise in natural domains that they remain relevant. Nor should one utilize synthetic problems except to support the systematic variation of domain features. In general, a well-balanced experimental program includes studies with both synthetic domains, to provide insight, and natural ones, to ensure relevance.[1]

B. Characteristics of the System

If we want to understand why a cognitive system behaves well or poorly, then we must vary characteristics of that system. The simplest version of this idea involves replacing the entire system with another, as typically occurs in competitions. Unfortunately, even when one system behaves uniformly better than another, which seldom happens, such comparisons provide no insight into the reasons for their behavioral diþerences.

One form of þner-grained study involves varying the parameters associated with the cognitive system and measuring the eþect on its behavior. For instance, one might alter the depth to which search occurs in a system that proves geometry theorems, the utility function used to guide a driving system's choices, and the relative values of pieces in a chess player. Such experiments can lead to conclusions about the importance of a parameter to system behavior, which may be unchanged across a wide range of parameter values, change slowly as the parameter varies, or produce sudden shifts at certain threshold values. Parametric studies may also detect interactions among settings that indicate nonlinear eþects.

Another experimental approach compares the basic system's behavior with that when one or more of its modules has been removed. For example, one might compare a driving agent with and without a component for planning routes. Similarly, one might examine a geometry theorem prover with and without a module that learns from previous proofs or a chess player that can or cannot analyze its opponent's strategy. Such lesion studies let one draw conclusions about the contribution of the removed components to the system's overall behavior. They can be especially useful in understanding integrated cognitive systems, since they can reveal interactions among modules. For instance, inclusion of planning and learning abilities in a driving system may provide beneþts greater than their sum when used alone.

C. Knowledge and Experience of the System

Cognitive systems rely centrally on knowledge about a domain to make inferences and generate candidate solutions to the problems they encounter. Knowledge is just as important a determinant of behavior as the domain and system characteristics. However, the precise impact of

[1] Unfortunately, this mixture is quite rare in the literature, presumably because it requires extra eþort from experimenters, but this does not reduce its importance for the study of intelligent systems.

knowledge on a speciþc intelligent system is an open issue that can be studied experimentally.

The methodology of lesion studies, which we discussed above in the context of system components, can be adapted easily to knowledge. We can run a geometry theorem prover with and without access to lemmas, we can ask a driver to deliver packages with and without a cognitive map of the city, and we can provide or not provide a chess player with a library of opening moves. In some cases, such lesion studies are equivalent to experiments with system modules, since certain components may be included only to utilize a speciþc type of knowledge. But the modules of many cognitive systems have more general abilities, so that running them with and without access to knowledge can uncover its importance independent of the component processes themselves.

Of course, the knowledge utilized by a cognitive system does not usually come in large packages, but rather in small, modular knowledge elements. As a result, one can also vary systematically the amount of knowledge available to the agent of a given type. For instance, a theorem prover may have access to many or few lemmas, a driver responsible for delivering packages may have a more or less complete cognitive map, and a chess player may know about diÞerent numbers of opening moves. Experiments that treat knowledge in this manner produce graphs that plot behavioral measures like eŽciency and quality against knowledge. These can also provide higher-order metrics that describe the rate of improvement per knowledge element, as we discussed earlier.

For cognitive systems that learn, we can examine the eÞects of experience in a similar manner. Here one relates the number of problems solved, the time spent by the agent, or other measures of experience to the standard behavioral variables. For example, one can graph the percentage of geometry theorems proved as a function of the number of previous eÞorts, the eŽciency of package delivery against the number of earlier trips, and the number of chess pieces lost against the number of games played. As mentioned earlier, such learning curves also provide higher-order information about the rate of improvement and asymptotic behavior.

IV. Repositories for Cognitive Systems

As we have noted, experimental studies of intelligent systems require some class of problems on which to measure behavior, but developing such tasks can be time consuming and expensive. The natural response is to develop a common repository of domains and problems for use by the research community. The earliest example was the UCI Machine Learning Repository (Blake & Merz, 1998), launched by David Aha in the late 1980s. This provided a variety of well-documented data sets for the evaluation of supervised learning systems, and within a few years it became so popular that most papers on machine learning utilized it in their experimental studies. Another model came from computational linguistics, where the annual TREC competitions came to drive many research eÞorts and has been imitated by other þelds, such as the AI planning community.

Unfortunately, despite their advantages, repositories and competitions also have negative aspects. Their very ease of use can encourage a community to focus only on the technical issues they represent. For example, the UCI repository encouraged increased learning research on classiþcation domains at the expense of work on problem-solving tasks. Moreover, many learning researchers have adopted a `bake-oÞ' mentality that is concerned only with improving performance scores over earlier systems, and competitions like TREC have much the same eÞect. To the extent that the contents of repositories come to be viewed as benchmark problems, they lose their usefulness for genuine scientiþc studies.

A. Desirable Characteristics of Testbeds

Nevertheless, a common repository is an obvious means to encourage and support research on integrated cognitive systems, so we should consider what characteristics would make it most useful. Like the UCI repository, it must include a variety of distinct domains to ensure the generality of experimental results. Moreover, its contents must be well documented and it must be easy for researchers to use, with a standardized format or interface to simplify interaction with diÞerent cognitive systems. These are key characteristics of existing repositories that are well worth replicating in new ones.

However, the repository should support experiments with integrated cognitive systems in ways that previous ones have not. For example, it should not contain data sets like the UCI site or the TREC competitions, or even sets of problems, like the planning competitions. Instead, it should provide the community with environments or testbeds in which researchers can evaluate their creations. Unlike many component AI algorithms, a cognitive system exists over time and requires some environment in which to operate. This environment need not be a physical one, but embodied cognitive systems are perhaps the most interesting variety, so the repository should contain some testbeds that support the study of physical agents.

A testbed provides supporting or enabling infrastructure for work on a given problem domain. Each testbed must include a deþnition of the tasks or missions that arise in its domain, stated in terms of initial situations and the desired states or objectives. Each domain should support a range of such tasks and, ideally, come with a problem generator that researchers can use to produce novel ones. A testbed provides infrastructure that facilitates experimentation by the community and thus can lead to insights about alternative approaches. Examples of infrastructural support include: external databases, such as geographic information systems, and the means to connecting to

these resources; the controlled capture, replay, halting, and restart of scenarios; and methods for capturing relevant performance measures via application programming interfaces, access to variables and parameters, and external physical instrumentation.

A well-designed testbed for cognitive systems eases their experimental evaluation, which follows naturally from certain desirable attributes of the infrastructure and problem set. To assist researchers in evaluating high-level behavior, it should provide an environment that has little or no dependence on actuation or sensor processing. In addition, the infrastructure and problem domain should oPer a rich operating environment, with the ability to model and control various entities. The testbed should let researchers vary, in quantiþable ways, the diŽculty or complexity of the environment or mission. Moreover, although the study of integrated systems is crucial, a testbed should also support evaluation of component subsystems, such as reasoning and learning methods, through parametric and lesion studies.

For domains that involve an external setting, one can certainly create a physical testbed to support evaluation, but another option is to develop a realistic simulated environment that can be used by many more research groups at much lower cost. For example, JacoP, Messina, and Evans (2001) describe a physical testbed for evaluating robot search and rescue, whereas Balakirsky and Messina (2002) report a simulated environment to support research on the same problem. Simulated testbeds have an additional advantage in that they allow easy variation of domain parameters, ranging from details of the environmental layout to noise in the agent's sensors. Moreover, they let one record detailed traces of the intelligent system's physical behavior and its mapping onto cognitive state, which in turn supports detailed analyses and replay starting from any point along the agent's behavioral trajectory.

However, as we noted above, testbeds that rely on synthetic domains also come with the danger of irrelevance. Whenever possible, they should be based closely on a physical testbed and provide simulations of suŽciently high þdelity. Wang (2003) describes one such simulated domain that incorporates models, based on a gaming engine that supports kinematics and dynamics, of the physical NIST arenas for urban search and rescue. To further ensure relevance for intelligent systems that sense their environment, a testbed may provide data sets collected from real sensors in analogous locations (e.g., Shneier, 2003). Such additions can help retain the advantages of physical environments while oPering the aPordability and ease of simulated ones.

B. Promising Domains and Testbeds

We can clarify the desirable features of testbeds with some examples. We have already mentioned the search and rescue domain, for which NIST has developed both physical and simulated testbeds. The primary task involves searching for survivors in an urban area after an earthquake or similar disaster. This domain requires the combination of sensing, planning, and action in an integrated cognitive system that can recognize humans, þnd routes through dangerous areas, and execute its plans successfully. The testbeds have been in place for a number of years and have been used ePectively in a number of international competitions.

Another candidate domain involves ying a simulated aircraft in a military setting. Keeping an airplane aloft can be a challenging control task, but by itself this does not require much cognitive activity or integration of diPerent capabilities. However, Jones et al. (1999) report a complex environment in which an agent must y a jet þghter, distinguish friendly from enemy aircraft, respond according to established doctrine, and communicate with other pilots. Their intelligent agent operated within the ModSAF environment, which was populated by other aircraft, some controlled by programs and others by humans. A related set of problems would involve ying an unmanned reconnaissance vehicle over enemy territory to gather information while avoiding dangerous areas.

A third challenging domain involves in-city driving. This raises few problems at the control level, since keeping a car upright, on the road, and within its lane does not require much intelligence. But the presence of buildings, sidewalks, traŽc signs and signals, moving and parked vehicles, and pedestrians make for a very rich environment that requires the allocation of perceptual attention and other resources. Moreover, driving can support many distinct high-level tasks, such as delivering packages, tailing another car unobtrusively, and pulling over vehicles for moving violations. These all require the integration of cognitive, perceptual, and motor components in a complex dynamical setting.

There already exist many simulated driving environments, but few have been developed with the intention of evaluating intelligent systems. Moriarty and Langley (1998) report a simulator for highway driving, but this environment had low þdelity and agents had limited options. More recently, Choi et al. (2004) describe an in-city driving environment, which they have used to evaluate a cognitive driving agent, that includes many more objects and a broader range of activities. Balakirsky, Scrapper, and Messina (in press) are developing another infrastructure, Mobility Open Architecture Simulation and Tools, that provides well-deþned interfaces to the various driving subsystems and rich visualization at various levels of resolution. Several organizations are using this system to test subsystems for vehicle control, but it remains to be seen whether the environment meets all the requirements for evaluating an integrated cognitive system.

Both driving and ying involve control of an individual agent, but an equally important class of domains involve managing a large set of other agents. Commanding troops in a battleþeld scenario is one example that requires capa-

bilities like monitoring, situation assessment, planning and scheduling of activities, and allocation of resources. However, interactive strategy games like Civilization have similar characteristics and complexity, and they are familiar to more people. Aha and Molineaux (2004) are constructing a framework that simpliþes the interface to such games, and thus will provide a set of related testbeds for the experimental study of integrated cognitive systems. Michael Genesereth (personal communication, 2004) is developing a diÞerent infrastructure to support an annual competition in generalized game playing (http://games.stanford.edu/), with the intent of fostering research eÞorts on exible approaches to intelligent behavior.

V. Concluding Remarks

In the preceding pages, we have considered the dependent measures and independent factors that arise in studying integrated cognitive systems, along with characteristics of repositories and testbeds to support such experiments. Before closing, we should situate these comments in the broader context of scientiþc experimentation. As in other þelds, the aim of systematic experiments is not to show that one approach is superior to another but rather to increase our understanding of complex systems. Such understanding may also lead to improved artifacts, but the overriding goal is to produce replicable and interpretable results that add to our scientiþc knowledge about intelligent behavior.

To this end, researchers should not carry out unmotivated comparisons between diÞerent systems or environments. In most cases, one should have a clear question in mind or a speciþc hypothesis that one wants to test, and the experimental design should re ect this intention. Simple demonstrations of functionality and generality are reasonable when one þrst develops a cognitive system, but they should quickly give way to scaling studies that reveal its ability to handle complexity and to lesion studies that identify the roles that its components play in determining overall behavior.

Whenever possible, experimental results should be utilized to test such hypotheses. Because most studies involve averaging across samples, one should be careful about drawing conclusions. Statistical tests can be useful for this purpose, but they are overrated, in that one can sometimes obtain `signiþcant' diÞerences between experimental conditions even when they are not substantial. Nor are statistical tests required when diÞerences are large, although reporting conþdence intervals is crucial for conditions with high variance.

Results that agree with an hypothesis lend it evidence, though they do not `conþrm' it; science can never draw þnal conclusions about any situation. Results that diverge from one's expectations count as evidence against a claim, and thus require additional explanation. Negative results need not imply failure, since they can lead one to alter assumptions about system behavior and suggest new ways to test them. The iterative loop of hypothesize and test is as central the study of intelligent systems as to other experimental disciplines.

Nevertheless, integrated cognitive systems pose special challenges that require creative adaptation of standard experimental methods. We must develop testbeds that exercise the full capabilities of such systems, rather than emphasizing tasks that can be handled by simple classiþcation or reactive control. We must study behavior at the system level, rather than focusing on component algorithms. Finally, we must design experiments that illuminate the manner in which the modules of such systems interact to produce exible and robust behavior. Taken together, these steps should let us transform the study of integrated cognitive systems into a dynamic and well-balanced experimental science.

Acknowledgements

This research was funded in part by Grant HR0011-04-1-0008 from Rome Laboratories. Discussions with David Aha, Michael Genesereth, and Barney Pell contributed to the ideas presented in this paper.

References

Aha, D. W., & Molineaux, M. (2004). Integrating learning in interactive gaming simulators. Proceedings of the AAAI-2004 Workshop on Challenges of Game AI. San Jose, CA: AAAI Press.

Balakirsky, S., & Messina, E. (2002). A simulation framework for evaluating mobile robots. Proceedings of the Performance Metrics for Intelligent Systems Workshop. Gaithersburg, MD.

Balakirsky, S., Scrapper, C., & Messina, E. (in press). Mobility Open Architecture Simulation and Tools Environment. Proceedings of the International Conference Integration of Knowledge Intensive Multi-Agent Systems. Boston.

Blake, C. L. & Merz, C. J. (1998). UCI repository of machine learning databases [http://www.ics.uci.edu/¾mlearn/MLRepository.html]. Irvine, CA: University of California, Department of Information and Computer Science.

Choi, D., Kaufman, M., Langley, P., Nejati, N., & Shapiro, D. (2004). An architecture for persistent reactive behavior. Proceedings of the Third International Joint Conference on Autonomous Agents and Multi Agent Systems (pp. 988{995). New York: ACM Press.

Cohen, P. R. (1995). Empirical methods for artiþcial intelligence. Cambridge, MA: The MIT Press.

JacoÞ, A., Messina, E., & Evans, J. (2001). Experiences in deploying test arenas for autonomous mobile robots. Proceedings of the Performance Metrics for Intelligent Systems Workshop. Mexico City, Mexico.

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

Kibler, D., & Langley, P. (1988). Machine learning as an experimental science. Proceedings of the Third European Working Session on Learning (pp. 81{92). Glasgow, Scotland: Pittman.

Langley, P. (October, 1996). Relevance and insight in experimental studies. IEEE Expert, 11{12.

Moriarty, D., & Langley, P. (1998). Learning cooperative lane selection strategies for highways. Proceedings of the Fifteenth National Conference on Artiþcial Intelligence (pp. 684{691). Madison, WI: AAAI Press.

Pell, B. (1996). A strategic Metagame player for general chess-like games. Computational Intelligence, 12, 177{198.

Shneier, M., Chang, T., Hong, T.H., Cheok, G., Scott, H., Legowik, S., & Lytle, A. (2003). A repository of sensor data for autonomous driving research. Proceedings of the SPIE Aerosense Conference. Orlando, FL.

Wang, J., Lewis, M., & Gennari, J. (2003). A game engine based simulation of the NIST USAR arenas. Proceedings of the 2003 Winter Simulation Conference (pp. 1039{1045). New Orleans, LA.

Metrics for Cognitive Architecture Evaluation

Robert Wray, Soar Technology, Inc., wray@soartech.com
Christian Lebiere, Carnegie Mellon University, cl@cmu.edu

Introduction

The problem of evaluating general architectures is a difficult one (Newell, 1990). Comparative evaluations that focus on performance alone are especially problematic. It is usually feasible to develop a specialized solution for any particular problem that will outperform a general solution, such as one developed within a cognitive architecture. Thus, an evaluation of the architectural approach derives from the power of the primitives of the architecture, the generality and flexibility of these primitives in providing solutions across a range of tasks, and the resulting ability to (relatively) rapidly develop an architectural solution via a common computing framework. While this power is assumed within the cognitive architecture community (and there is significant anecdotal evidence to support it), today the community lacks a *scientific* foundation for measuring and evaluating these claims.

This paper identifies a number of metrics that could be important for assessing cognitive architectures across a range of applications domains. The metrics are organized according to a taxonomy of requirements for intelligent systems developed by Anderson & Lebiere (2003). We focus only on the functional aspects of their analysis, rather than those non-functional requirements specific to human cognition, which are also detailed in the Anderson and Lebiere evaluation approach.

These metrics together reflect our attempt to capture and measure many necessary components of general intelligent behavior, rather than solely performance metrics, which are often the primary means of evaluating intelligent systems in AI. We introduce two metrics novel to cognitive-architecture research, incrementality and adaptivity, which may prove to be useful for capturing and expressing the cumulative value of cognitive-architecture-based solutions across multiple tasks within a domain and across multiple application domains. Our approach is far from complete, in that several requirements include only notional metrics. However, this approach provides at least an empirical foundation for comparing work within and across the development cognitive architectures that can provide more objective measures of a cognitive architecture's capabilities and utility as a platform for general intelligence.

One of the primary factors that makes achieving general, intelligent behavior such a difficult problem is the complexity and variation found in the environment. For general intelligence, agents must be able to cope effectively with this complexity. However, while complex, the environment (usually) is not chaotic. It operates according to laws and general properties and can be characterized according to its complexity. Russell and Norvig's (1995) influential textbook introduces a number of contrasting dimensions that can be used for characterizing domains. For example, accessible domains provide complete access to the state of the environment while, in inaccessible domains, information relevant to a good choice at some point in time may not be available to the agent via direct perception. Different problems will have different complexity profiles based on these dimensions. Many of the metrics introduced below will also interact with these dimensions, so that quality of the overall solution and the problem complexity define a functional space for the metric for some class of related problems. A significant qualification to the work presented here is the lack of consistent measures of complexity across different application domains. Some problems and domains provide simple measures of complexity (such as the number of cities in a Traveling Sales Man tour) which can be used to evaluate the performance of a system with increasing complexity but, to-date, there are no domain-general characterizations of complexity that enable cross-domains comparisons. This will limit the utility of some of the proposed metrics to comparison within particular domains.

The remainder of the paper introduces metrics for each category of the general requirements of Anderson & Lebiere (2003). Many performance-based metrics are reused from one category to the next, suggesting specific ways in which base performance characteristics can be systematically explored to provide a more complete, multi-dimensioned characterization of performance results. While, in most cases, we propose objective, quantitative metrics, in some cases, only qualitative and/or subjective evaluation is possible today. Perhaps the workshop discussion will lead to ideas and insights for more objective approaches to evaluation in these areas as well.

Behave as an (almost) arbitrary function of the environment

Environments generally will change independently of an intelligent system ("dynamic" in the Russell and Norvig properties) and the actual state of an environment may not be known or directly perceivable (inaccessible). Thus, the intelligent system must be able to act in the situation it finds itself in (and even if it is different than the one it expected to be in). This flexibility implies a breadth of capability, meeting the complexity of the environment with appropriate responses.

Taskability is the ability of a system to adapt to new/novel problems without human (programmer) intervention. Taskability is difficult to measure because there is no "absolute" notion of taskability -- a particular quantitative measure for one domain might represent the best one could achieve, while in another, it might be a baseline.

Researchers in AI have generally evaluated taskability by adopting a set of benchmark tasks against which a system is developed, and then introduced novel tasks within the same domain and tested system performance on these new tasks (Hanks, Pollack, & Cohen, 1993). This approach provides a reasonable qualitative measure of taskability within a domain.

Incrementality is the ability to extend a cognitive-architecture-based system from one set of tasks to another set, which can be either a superset of the original set (as an example of generalization) or reflect a different set of requirements (an example of the robustness and taskability of the architecture). Incrementality could possibly be measured by the degree of overlap between the solutions to the two sets of problems. For instance, if some cognitive system provides a quite general capability, a small task-specific addition at the knowledge representation level might be sufficient to tackle a new task. On the other hand, if a system is overly specific to a particular problem, then significant reworking for the new task might be necessary and the resulting incrementality will be poor. The AAAI General Game Playing (GGP) competition is representative of this attempt to realize a more general capability than just effective play of some specific game.

Measuring incrementality will help expose excessive benchmark-driven task specialization and thus help ensure the generality of an architectural framework. To our knowledge, incrementality is a novel dimension that has not been previously evaluated, although it is consistent with the general notion of a "cumulation of results" as discussed in Newell (1990). The primary difference is that rather than a qualitative accumulation of results across different tasks, as suggested by Newell, incrementality is proposed as a quantitative measure that expresses the actual overlap at the source code level between different applications of an architecture.

Figure 1 illustrates a notional approach to measuring incrementality in the evaluation of an architecture. Here, we propose a very simple measure of incrementality. Incrementality is the ratio of the unchanged lines of source code to total lines of code needed for the solution of some problem, in comparison to the source code used to solve previous problems. One of the key points of this definition to incrementality is that it makes no distinction between the architecture source code (typically written in a standard high level language, such as Java, C, or LISP) and the knowledge representations encoded in the language(s) an architecture defines for specifying content.

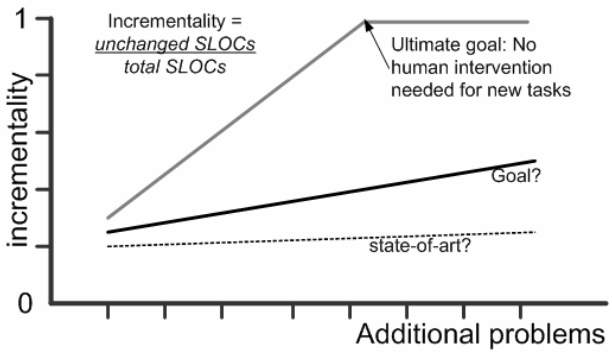


Figure 1: Measuring incrementality.

The simplicity of this definition has two advantages. First, because it does attempt to distinguish or weigh the relative contribution of different elements in the architecture-based application, it provides a direct analog to reuse metrics in software engineering generally and allows comparison to non-architecture-based approaches and their level of reuse. Second, tools to measure incrementality could be readily developed using existing source control revision comparison tools (e.g., "diff").

Today, typically, at best the source code of an architecture is reused from one application to the next, resulting in a modest but not trivial baseline. This baseline suggests some of the inherent value in cognitive architectures generally, since the incrementality of non-cognitive-architecture-based intelligent systems development is near nil (i.e., systems development begins near *de novo* for new applications).

In the ideal case, which is a long-term, not near-term goal, for any existing cognitive architecture, full incrementality is reached: no new changes to the architecture or its encoded knowledge are needed for a new task. In the meantime, as shown in the middle line, incrementality could be used as an explicit tool for understanding at a gross level how much of an architecture and its application are reusing the previous results over time. While full incrementality is likely infeasible, this approach would provide at least a coarse measure of incrementality for cognitive systems and give insight to other scientists about the level of reuse and cumulation within a particular architectural paradigm.

An obvious drawback of this definition is its coarse-grained nature. For example, the Soar source code is very roughly 50,000 source lines of code (SLOCs) of C. An obvious parallel to a SLOC in the Soar language is a production. However, most Soar systems have only a few hundred productions; the largest Soar application system, TacAir-Soar (Jones, et al, 1999), today has about 10,000 productions. Thus, significant changes in the kernel level of Soar are likely to mask reuse at the production-knowledge level of representation; similarly, the lack of reuse at the production-knowledge level will be masked by the SLOCs. We expect approaches that take incrementality seriously will provide a number of more fine-grained measures to highlight these effects.

Another possible limitation of incrementality proposed here is that it measures "latent" capability, rather than the capabilities actually used in the execution of a task. For example, using another Soar example, one could likely include the productions from a number of distinct applications into a single Soar system. In such a situation, the numerator of the incrementality metric would increase with each new application, but little reuse would be guaranteed. We propose that incrementality be used in conjunction with knowledge utilization (described below) to make explicit the actual use of the knowledge within an incrementally increasing source repository.

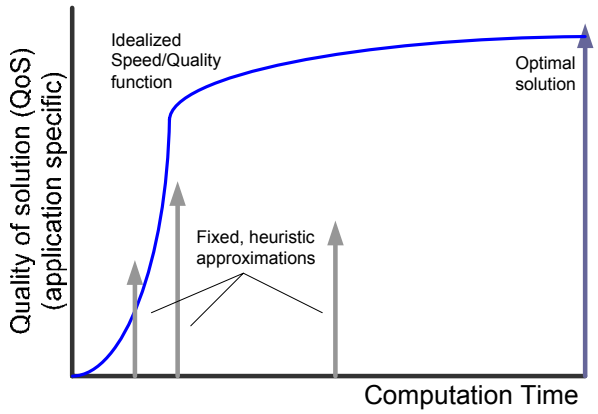


Figure 2: The relationship between computational investment and quality of solution.

Operate in real time

For an intelligent system to behave intelligently, it must able to recognize a situation that it cares about, determine an appropriate response, and then act. However, the world in which an agent operates may be continually changing. Thus, the agent must perform its internal processes quickly, relative to the speed of change in the environment, or its chances of survival/success diminish in a ling-term-existence. Many reflexes and instincts can be viewed as evolutionary solutions to the problem of fast reaction in the animal world, "hard wiring" responses to specific situations. The important requirement is that the speed of response is sufficient for the demands of the problem, rather than being as fast (in absolute terms) as possible.

There is typically also a trade-off between performance measures and the quality of solution (QoS). As an example, consider the range of potential relationships between the time taken to compute or generate some response or behavior, and the quality of the resulting solution, as illustrated in Figure 2. There may a fixed computation time that is needed before any solution or response is produced, as shown by the gray, pulse arrows. Computing an optimal solution typically provides an upper limit on compute time. While heuristics can be used to shorten the delay, except in special cases (such as admissible heuristics), more computational effort does not necessarily result in improving QoS. In the ideal case, behavior generation and reasoning has an *anytime* property, in which additional computation leads to improvement in the overall quality of solution. However, many real-world problems are difficult to formulate in terms of anytime response characteristics. The main lesson from Figure 2 is that there is typically a relationship between the time invested in generating a solution and its overall quality, meaning that absolute performance comparisons should be normalized or cast against QoS.

Metrics for real-time operation should likely include:

Response time is the time between the onset/assignment of a task (including specific subtasks) and its resolution. As noted above, response time should be accompanied by a Quality of Solution associated metric, to distinguish between satisficing/non-satisficing solutions, and to demonstrate trade offs between solution quality and response time. Also, response time should distinguish between soft and hard real time responses.

Cognitive cycles/cognitive operations per second (COPS) measures the cycles (or %age of total CPU time) devoted to cognitive-architecture operations. This metric is a poor stand-alone metric because decreases or increases cannot be evaluated in an absolute sense, similar to the way comparing operations per second in RISC vs. CISC architectures is also marginally informative. Although it has limited utility as an absolute measure, cognitive operations/time is a good *relative* metric, allowing one to assess improvements against a baseline or benchmark. Scalability (as discussed further below) can be partially evaluated according to such metrics.

Extended operation/longevity: Intelligent systems must be able to persist over long durations. What is the uptime of the cognitive architecture in a particular application? How do other performance metrics change as uptime increases? For example, does system performance degrade as uptime increases (this is often observed in some learning systems, where the addition of knowledge via learning leads to significant degradation in knowledge retrieval performance)? While it may be true that particular architectures exhibit poor uptimes due to implementation issues (a common problem is the implementation of inherently parallel processes on serial machines), the engineering-oriented reportage of these measures would give potential users insights into the maturity of current implementations and a better understanding of their actual potential value in a proposed application.

Exhibit rational, effective adaptive behavior

An intelligent system must not only respond to its environment, but it should respond in a manner appropriate for the situation. In particular, as the world changes, the agent should adapt its behavior to the situation such that it continues to make progress on its long-term goals (i.e., it cannot just be reactive). Because environments have consistent (or slowly changing) dynamics, an agent can make predictions about future states and attempt to act to effect the environment in ways that meet its goals. This capability is useful both in domains that are deterministic and non-deterministic. Different sources of knowledge available in the environment can be used to formulate goals, to act to achieve them, and to recognize when goals are met or unreachable. Adaptation to the specific environment is important because the agent's existence may span a long period of time (as above), and non-adaptive behavior may influence the survivability or viability of the agent in an application domain.

Adaptivity corresponds to both short-term adaptivity (changes in how the agent responds within an on-going episode of behavior) and long-term adaptivity (acquiring new general knowledge to improve long-term performance). Adaptivity may be a consequence of learning but is not a learning metric per se; instead, adaptivity is focused on measuring the ability of a system to respond appropriately to variation in its environment. The analog of adaptivity in control systems is the region of stability in a non-linear control problem.

Adaptivity can be quantified by comparing the performance gain of an adaptive version of an agent system, to a non-adaptive system. This approach is somewhat related to robustness (below) but does not reduce to it. Both robustness and adaptivity are necessary: robustness provides acceptable performance in unforeseen situations, adaptivity suggests a cognitive architecture simplifies a priori engineering and provides a more efficient solution-authoring process than one in which complete capability has to be specified by the designer, in addition to allowing the system to adapt on its own to the changing dynamics and task requirements of a domain.

Figure 3 illustrates a possible, albeit notional approach to a domain-specific measure of adaptivity. In this approach, adaptivity is the ratio of the size of perturbation in the environment (measured in terms of problem complexity) to the difference in the resulting quality of solution. As the difference in quality of solution increases (presumably, via a poorer quality solution), adaptivity decreases. Similarly, if quality of solution remains constant while the perturbation increases, adaptivity also increases.

In the current state-of-the-art, intelligent systems are generally designed for a given problem complexity and quality of solution, so that any perturbations, even ones in which complexity decreases, quality of solution is likely to decrease (with very sharp decreases as complexity increases, as suggested by the dotted lines in the figure). The solid lines intersecting the baseline complexity point in the figure illustrate a "minimum" standard for adaptivity. In this case, decreasing complexity preserves the baseline QoS and, as problem complexity increases, solution decreases somewhat more gracefully than the state-of-art.

The figure also suggests how robustness and learning (discussed below) can impact overall adaptivity. The stars in the figure represent a particular point on the problem complexity curve. In comparison to a non-adaptive, baseline system, processes in the agent systems that enable robustness to perturbation could result in an improved Quality of Service. In this case, we assume there are many decisions and actions an agent takes to achieve some result. Robustness processes can produce somewhat better intermediate results, leading to improvement in overall QoS at the baseline level of complexity.

Agent learning should result in an improved quality of service as well (whether resulting in improving performance or solution quality). Thus, if learning is effective, it will shift the systems resulting adaptivity up and out, providing improved quality of solution for any particular point in the problem complexity space, as suggested by the third line in the figure.

Additional metrics for rational, effective, adaptive behavior include the performance metrics introduced previously, but with different emphases and points of comparison:

Response time: What is the response time to an unexpected event? Is the system able to resume execution of an existing plan once an interrupting event to which the agent has responded? As one example, Wray and Laird (2003) developed a domain specific metric to illustrate the reaction time of a system in response to a triggering event.

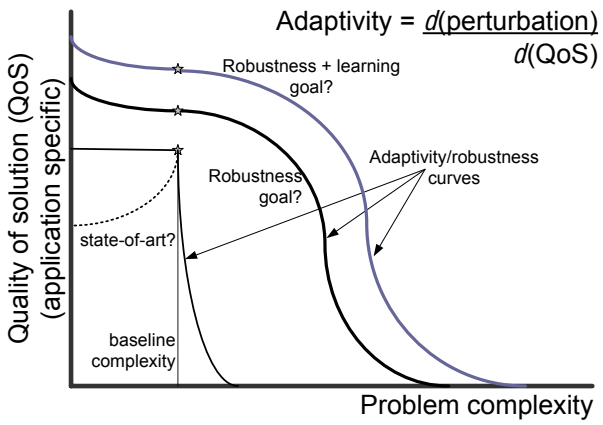


Figure 3: An adaptivity metric.

However, there is no domain general metric for evaluating this aspect of response time.

Scalability in this case is the ability to handle increasingly complex problems. Scalability in the context of this requirement is the ability to adapt to the requirements of specific problems. For example, a system might choose an analytic solution for Traveling Salesman problems up to some system-determined size *n*, then switch to a heuristic method for problems of size greater than *n*. The Traveling Salesman problem represents a problem in which the problem complexity can be systematically scaled; however, collective, subjective judgment is currently the only available approach to ranking domain problems generally along each dimension of complexity.

Use vast amounts of knowledge

There are many different objects in the environment, including other agents. Most objects obey consistent, predictable dynamics, although the agent may not have complete or correct knowledge about these laws and dynamics. These attributes of the environment make "knowledge" a fundamental requirement for intelligent systems. "Knowledge" here really means nothing more than having the means to predict future states of the environment; it does not necessarily imply deep, first-principals knowledge (e.g., the Three Laws of Thermodynamics). However, as the agent's environment becomes more complex (in terms of the objects and interactions it must manage to succeed), it will need increasingly large stores of knowledge to cope with the complexity. Potential metrics for this requirement include:

Knowledge capacity: How much "knowledge" is represented in a cognitive-architecture-based agent? Within a particular symbolic cognitive architecture, it should be relatively straightforward to characterize the size of a knowledge base. However, this metric is difficult to quantify generally. Knowledge content will be especially difficult to quantify in non-symbolic systems. Although enumerating knowledge representations is trivially simple in symbolic systems, simple enumeration can also be misleading (a Soar rule and ACT-R rule correspond to different grain-sizes of cognitive operations; how should ACT-R rules be combined with ACT-R chunks, etc.?)

Knowledge capacity may also have little meaning in architectures (and other intelligent systems) where there is no distinct line between architectural processes and knowledge content. As was observed for many of the performance metrics, knowledge capacity is better employed as a relative measure, than an absolute one.

Figure 4 illustrates one possible use of knowledge capacity. In the ideal case, across a range of tasks and domains, increasing stores of knowledge should result in overall improvements in the average quality of solution. However, the state-of-art today is often that increases in knowledge capacity actually reduce the quality of solution, chiefly due to the additional costs of storing more knowledge encodings and having to search them for retrieval.

Rather than the ideal, in the near-term, a goal of cognitive-architecture-based applications should be to be able to demonstrate that increasing knowledge capacity does not degrade quality of solution. For example, Doorenbos (1994) showed that performance (which is only one measure of quality of service) did not decrease substantially when a research application of Soar was scaled to a million productions. However, in practice, knowledge engineers within the Soar community often do attempt to limit the size of Soar knowledge bases because storing and matching any additional knowledge has some incremental cost, even if small, when implemented on serial hardware. Other architectures have similar limitations, although, as discussed above, it may be that decreasing QoS with increasing knowledge capacity represents more of an engineering issue than a theoretical one.

Knowledge utilization: Knowledge capacity reflects the total content of knowledge that *could* be applied in some situation. However, that capacity may largely be latent for any particular application (especially when increasing incrementality is an explicit goal, as described previously). Knowledge utilization reflects the knowledge that is actually used in the execution of a set of tasks. A straightforward way to measure knowledge utilization is to count the unique instances of each knowledge representation activated or applied in the course of performing a representative sampling of tasks within a domain.

A representative sampling is necessary because any single task instance may only activate part of the utilized knowledge store. TacAir-Soar offers another good example. The TacAir-Soar knowledge base spans many different missions a military pilot might fly; everything from fighter intercepts to flying air refueling missions. For any particular mission, knowledge utilization is likely to be low, but, across the span of all missions, we would assume knowledge utilization would be close to 100% within this single application.

Learning complicates knowledge utilization measures. For example, both Soar and ACT-R include mechanisms of compiling / composing production firings (Laird, et al, 1986; Taatgen & and Lee, 2003). In similar, future situations, the newly-created productions will likely supplant the original ones. Naively, a simple way to avoid this problem would be to "start counting" with the original

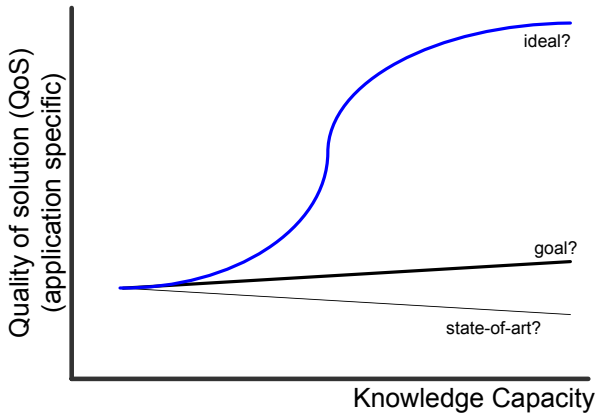


Figure 4: Knowledge capacity.

knowledge base. However, from the point of view of an agent who's knowledge base in increasing in capacity within and across domains, it may be undesirable (or infeasible) to perform some task without the learned knowledge representations.

As suggested, performance metrics also interact strongly with the notion of knowledge capacity and knowledge utilization:

Response time: How does response time change when across orders-of-magnitude differences in the knowledge capacity of a particular system? Does response time change as knowledge utilization changes?

Cognitive cycles/cognitive operations per second: How does the basic cycle time change across orders-of-magnitude differences in the encoded "knowledge" in a particular system? In Figure 4, the limitation of the state-of-art is assumed to derive from a decrease in the cycles/second with increasing knowledge.

System memory footprint: How do memory requirements scale with knowledge capacity?

Behave robustly in the face of error, the unexpected, and the unknown

An agent will always have incomplete or partially incorrect knowledge of the many objects and other agents that appear in its environment. Yet, in order to thrive, it must overcome these limitations and complexities in the environment and behave robustly. The environment itself provides some important aid in this respect. First, the environment usually has structure and abstractions that alleviate the unpredictability of the situation. A shopping agent may not have experience with the specific website just encountered in the execution of a product search, but previous experience with and knowledge of similar websites makes the situation more predictable and provides suggestions for courses of action that increase the likelihood of a successful transaction. Second, the environment provides many sources of knowledge including direct experience, observation of others, instruction, etc. These sources of knowledge can be draw on (and learned) in order to encounter the inherent uncertainty in the environment more readily.

Robustness is the ability to successfully (autonomously/dynamically/safely) withstand perturbations in expected events and tasks. There are no existing, general metrics for robustness, although domain specific metrics have been developed (Nielsen, Beard et al, 2002). One possibility would be to cast robustness as the ratio of the degree of success vs. the degree of perturbation, which is a specialization of the adaptivity metric discussed above. However, both of these measures will be domain and task dependent.

Robustness has a direct relationship with the notion of taskability introduced earlier. The primary difference is that taskability focuses on the ability of the system to handle variation in tasks, while robustness primarily focuses on success in environments where the expected dynamics are changing.

Stochastic assimilation is the ability of the system to capture and reflect in behavior the stochastic character of the environment. Where robustness reflects the ability to recover from unexpected events, in any real application an agent will also need to make rational choices in an environment where those choices are governed by probability distributions. As an example, ACT-R has been applied to a range of non-deterministic games (West, et al, 2006) and has demonstrated its ability to learn the stochastic dynamics of these games. An obvious approach to measuring stochastic assimilation within a domain is to measure the change in QoS over time. We have not yet considered a domain-general formulation of this measure.

Integrate diverse knowledge.

The objects and other agents in the environment result in many different sources of knowledge. To act appropriately, the agent must integrate its knowledge of these different objects to act appropriately in a situation. For example, in an evidence marshalling task, such as "detective's helper," the system must integrate knowledge from "scene of the crime" reports, draw on past experience, be able to reason deductively as well as by abduction and analogy, use general knowledge (language, ontology, etc.) as well as domain specific knowledge (such as the typical etiologies of particular crimes). While the task could possibly be accomplished without multiple sources of knowledge, the assumption is that the introduction of a much larger branching factor in both knowledge search and problem search is offset by the ability to reach a conclusion in just a few steps.

A key aspect of this requirement is the ability to integrate *diverse* sources of knowledge effectively. As another example, consider an agent that must conduct a TSP tour over an actual landscape, with latitude and longitude coordinates of "cities," roads, obstacles, varying constraints (e.g., the fastest tour, vs. the shortest). A common approach to a problem like this is to attempt to map these diverse constraints into an edge-cost that facilitates an algorithmic solution, such as the application of Dijkstra's algorithm. Cognitive architectures typically enable a more open-ended approach to the problem, where individual aspects of the problem can be encoded directly (and with comparatively little information loss) and then combined at run-time to provide a solution.

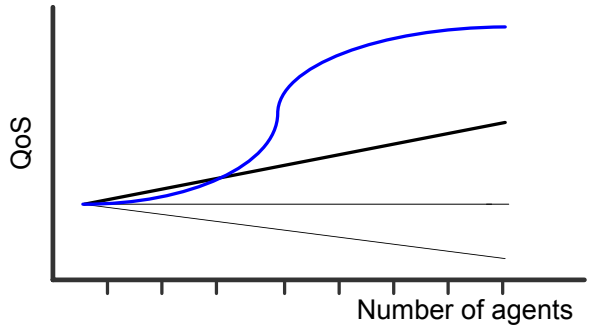


Figure 5: Benefits of social agency.

At present, we have not yet developed metrics for knowledge integration. Knowledge capacity and knowledge utilization express coarse aspects of the general capability, but are not themselves sufficient to express this requirement.

Behave autonomously in a social environment

Over a long life of continual behavior in the environment, an agent will pursue its own success and act on its own. However, the agent may live in a social environment with other agents and actors. Other agents can complicate ultimate success (competitors), but may also be the source of additional knowledge and cooperation. Other actors require that the system be knowledgeable of them (able to predict actions and evaluate intents) as well as the ability to communicate with the other actors.

Scalability: How many "cognitive agents" can interact together, given some baseline performance measure? Ideally, overall performance costs will increase at most linearly with the addition of multiple agents.

Figure 5 suggests some of the potential impacts of multiple agents. In the degenerate case, represented by the dotted line, the addition of additional agents decreases the quality of solution. In the neutral case, represented by the horizontal line, the addition of more agents does not affect the resulting quality of solution. In this case, additional agents are not providing benefit to the agent performing its task. The heavier, straight line suggests a linear benefit of the addition of more agents. In the ideal case, there is a synergistic benefit, with the addition of more agents significantly increasing to overall quality of solution. This latter effect is a common goal of many multiagent systems technologies, such as swarming (e.g., Brueckner and Parunak, 2002). Demonstrating such benefits with cognitive-architecture-based agents has not generally been undertaken.

Exhibit self-awareness and a sense of self

Because existence is long term, an agent will have many opportunities to recognize deficiencies in its knowledge of the environment, and can utilize the many different sources of knowledge in the environment to address the deficiencies. "Self-awareness" is the capability to recognize these opportunities to reflect on the state of one's self and one's behavior and to improve future action

by evaluating the efficacy of actions taken in the current situation. Self-awareness can also include notions of performance monitoring and fault localization within the overall system (i.e., extending beyond the cognitive components of the system).

Adaptivity and **Robustness** are a result of meta-cognitive capabilities, but we have not yet developed a metric that would reflect meta-cognitive capabilities generally. A subjective approach would be to enumerate and define the kinds of processes available in the system for meta-cognitive activity. For example, Soar's automatic subgoaling / impasse mechanism are assumed to provide the basis for meta-cognitive capabilities in Soar, although the basic architectural processes must be completed by encoded knowledge as well.

Learn from its environment

Can the system produce a breadth of different types of learning and improve its function? If the world is consistent and the agent's knowledge is incomplete (as will almost always be the case), then an obvious requirement for long-term success in the environment is learning. Learning, which will draw from the many sources of knowledge in the environment, re-shapes behavior. In the ideal case, learning improves outcomes of future experiences in comparison to past, similar ones.

An intelligent system must not only respond to its environment, but it should respond in a manner appropriate for the situation. In particular, as the world changes, the agent should adapt its behavior to the situation such that it continues to make progress on its long-term goals (i.e., it cannot just be reactive). Because environments have consistent (or slowly changing) dynamics, an agent can make predictions about future states and attempt to act to effect the environment in ways that meet its goals. Different sources of knowledge available in the environment can be used to formulate goals, to act to achieve them, and to recognize when goals are met or unreachable. As discussed above, learning is reflected in improving **adaptivity** within a specific environment. Non-adaptive behavior may influence the survivability or viability of the agent in an application domain

Performance measures also interact strongly with learning. The agent's existence may span a long period of time and the cognitive architecture must strike a solution to the utility problem (Holder, 1990), such that learning increases the quality of solution with experience, rather than decreasing it.

Conclusions

Evaluating cognitive architectures has proven difficult, because both their theoretical and practical value is often only emergent from a breadth of application demonstrations. Cognitive-architecture benchmarks tend to look especially poor in performance-oriented benchmarking, because they typically include an integrated collection of processes and mechanisms, some of which may not significant value in a given benchmark task. We have proposed a number of specific metrics, organized according to a general list of requirements for intelligence, which could be used to measure some of the (assumed) utilitarian advantages of cognitive architectures as general tools for building intelligent systems.

References

1. Anderson, J. R., & Lebiere, C. (2003). The Newell test for a theory of cognition. *Behavioral and Brain Science, 26*, 587-637.
2. Brueckner, S. A., and Parunak, H. V. D. (2002). Swarming Agents for Distributed Pattern Detection and Classification. *In* "AAMAS Workshop on Ubiquitous Computing", Bologna, Italy.
3. Doorenbos, R. B. (1994). Combining left and right unlinking for matching a large number of learned rules. *In* "Twelfth National Conference on Artificial Intelligence (AAAI-94)". AAAI Press, Seattle, Washington.
4. Hanks, S., Pollack, M. E., & Cohen, P. R. (1993). Benchmarks, Test Beds, Controlled Experimentation, and the Design of Agent Architectures. *AI Magazine, 14,* 17-42.
5. Holder, L. B. (1990). The General Utility Problem in Machine Learning. *In* "Machine Learning: Proceedings of the Seventh International Conference", pp. 402-410. Morgan Kaufmann Publishers, San Mateo, CA.
6. Jones, R. M., Laird, J. E., Nielsen, P. E., Coulter, K. J., Kenny, P. G., and Koss, F. V. (1999). Automated Intelligent Pilots for Combat Flight Simulation. *AI Magazine* **20**, 27-42.
7. Laird, J. E., Rosenbloom, P. S., and Newell, A. (1986). Chunking in Soar: The anatomy of a general learning mechanism. *Machine Learning* **1**, 11-46.
8. Newell, A. (1990). *Unified Theories of Cognition.* Cambridge, Massachusetts: Harvard University Press.
9. Russell, S., & Norvig, P. (1995). *Artificial Intelligence: A Modern Approach.* Upper Saddle River, NJ: Prentice-Hall.
10. Taatgen, N. A., and Lee, F. J. (2003). Production Compilation: A simple mechanism to model Complex Skill Acquisition. *Human Factors* **45**, 61-76.
11. West, R. L., Lebiere, C. & Bothell, D. J. (2006). Cognitive architectures, game playing and human evolution. In Sun, R. (Ed) Cognition and Multi-Agent Interaction: From Cognitive Modeling to Social Simulation. NY, NY: Cambridge University Press
12. Wray, R. E., & Laird, J. E. (2003). An architectural approach to consistency in hierarchical execution. *Journal of Artificial Intelligence Research*, 19, 355-398.

BEHAVIORAL AND BRAIN SCIENCES (2003) 26, 000–000
Printed in the United States of America

The Newell Test for a theory of cognition

John R. Anderson
Department of Psychology–BH345D, Carnegie Mellon University, Pittsburgh, PA 15213-3890.
ja @cmu.edu http://act.psy.cmu.edu/ACT/people/ja.html

Christian Lebiere
Human Computer Interaction Institute, Carnegie Mellon University, Pittsburgh, PA 15213-3890.
cl@cmu.edu http://www.andrew.cmu.edu/~cl

Abstract: Newell (1980; 1990) proposed that cognitive theories be developed in an effort to satisfy multiple criteria and to avoid theoretical myopia. He provided two overlapping lists of 13 criteria that the human cognitive architecture would have to satisfy in order to be functional. We have distilled these into 12 criteria: flexible behavior, real-time performance, adaptive behavior, vast knowledge base, dynamic behavior, knowledge integration, natural language, learning, development, evolution, and brain realization. There would be greater theoretical progress if we evaluated theories by a broad set of criteria such as these and attended to the weaknesses such evaluations revealed. To illustrate how theories can be evaluated we apply these criteria to both classical connectionism (McClelland & Rumelhart 1986; Rumelhart & McClelland 1986b) and the ACT-R theory (Anderson & Lebiere 1998). The strengths of classical connectionism on this test derive from its intense effort in addressing empirical phenomena in domains like language and cognitive development. Its weaknesses derive from its failure to acknowledge a symbolic level to thought. In contrast, ACT-R includes both symbolic and subsymbolic components. The strengths of the ACT-R theory derive from its tight integration of the symbolic component with the subsymbolic component. Its weaknesses largely derive from its failure, as yet, to adequately engage in intensive analyses of issues related to certain criteria on Newell's list.

Keywords: cognitive architecture; connectionism; hybrid systems; language; learning; symbolic systems

1. Introduction

Allen Newell, typically a cheery and optimistic man, often expressed frustration over the state of progress in cognitive science. He would point to such things as the "schools" of thought, the changes in fashion, the dominance of controversies, and the cyclical nature of theories. One of the problems he saw was that the field had become too focused on specific issues and had lost sight of the big picture needed to understand the human mind. He advocated a number of remedies for this problem. Twice, Newell (1980; 1990) offered slightly different sets of 13 criteria on the human mind, with the idea (more clearly stated in 1990) that the field would make progress if it tried to address all of these criteria. Table 1 gives the first 12 criteria from his 1980 list, which were basically restated in the 1990 list. Although the individual criteria may vary in their scope and in how compelling they are, none are trivial.

These criteria are functional constraints on the cognitive architecture. The first nine reflect things that the architecture must achieve to implement human intellectual capacity, and the last three reflect constraints on how these functions are to be achieved. As such, they do not reflect everything that one should ask of a cognitive theory. For example, it is imaginable that one could have a system that satisfied all of these criteria and still did not correspond to the human mind. Thus, foremost among the additional criteria that a cognitive theory must satisfy is that it has to correspond to the details of human cognition. In addition to behavioral adequacy, we emphasize that the theory be capable of practical applications in domains like education or therapy. Nonetheless, while the criteria on this list are not everything that one might ask of a full theory of human cognition, they certainly are enough to avoid theoretical myopia.

While Newell certainly was aware of the importance of having theories reproduce the critical nuances of particular experiments, he did express frustration that functionality did not get the attention it deserved in psychology. For instance, Newell (1992) complained about the lack of attention to this in theories of short-term memory (STM) – that it had not been shown that "with whatever limitation the particular STM theory posits, it is possible for the human to function intelligently." He asked, "why don't psychologists address it (functionality) or recognize that there might be a genuine scientific conundrum here, on which the conclusion could be that the existing models are not right?" A theory that predicts the correct serial position curve in a particular experiment, but also says that humans cannot keep track of the situation model implied by a text they are reading (Ericsson & Kintsch 1995), is simply wrong.

So, to repeat: we are not proposing that the criteria in Table 1 are the only ones by which a cognitive theory should be judged. However, such functional criteria need to be

 0140-525X/03 $12.50

given greater scientific prominence. To achieve this goal, we propose to evaluate theories by how well they do at meeting these functional criteria. We suggest calling the evaluation of a theory by this set of criteria "The Newell Test."

This target article reviews Newell's criteria and then considers how they apply to evaluating the various approaches to the study of human cognition. We focus on evaluating two approaches in detail. One is classical connectionism, as exemplified in publications like McClelland and Rumelhart (1986), Rumelhart and McClelland (1986b), and Elman et al. (1996). The other is our own ACT-R theory. To be concrete, we suggest a grading scheme and issue report cards for the two theoretical approaches.

2. Newell's criteria

When Newell first introduced these criteria in 1980, he devoted less than two pages to describing them, and he devoted no more space to them when he described them again in his 1990 book. He must have thought that the criteria were obvious, but the field of cognitive science has not found them all obvious. Therefore, we can be forgiven if we give a little more space to their consideration than did Newell. In this section, we will try to accomplish two things. The first is to make the case that each is a criterion by which all complete theories of cognition should be evaluated. The second is to try to state objective measures associated with the criteria so that their use in evaluation will not be hopelessly subjective. These measures are also summarized in Table 1. Our attempts to achieve objective measures vary in success. Perhaps others can suggest better measures.

John Anderson received his B.A. from the University of British Columbia in 1968 and his Ph.D. from Stanford University 1972. He has been at Carnegie Mellon University since 1978, where he is a professor of psychology and computer science. His current research is on three enterprises involved in the testing of various aspects of the ACT-R theory of cognitive architecture: (1) to model the acquisition of cognitive skills, particularly those involving dynamic problem solving; (2) the application of the architectures to developing intelligent tutoring systems and cognitive agents for training; and (3) research on brain imaging to identify the neural correlates of the cognitive architecture.

Christian Lebiere is a Principal Research Scientist at the Micro Analysis and Design Unit, School of Computer Science, Carnegie Mellon University. He received his B.S. in Computer Science from the University of Liege (Belgium), and his M.S. and Ph.D. from the School of Computer Science at Carnegie Mellon University. During his graduate career, he worked on the development of connectionist models, including the Cascade-Correlation neural network learning algorithm. Since 1990, he has worked on the development of the ACT-R hybrid architecture of cognition. His main research interest is cognitive architectures and their applications to psychology, artificial intelligence, human-computer interaction, decision-making, game theory, and computer-generated forces.

2.1. Flexible behavior

In his 1990 book, *Unified Theories of Cognition,* Newell restated his first criterion as "behave flexibly as a function of the environment," which makes it seem a rather vacuous criterion for human cognition. However, in 1980 he was quite clear that he meant this to be computational univer-

Table 1. *Newell's Functional Criteria for a Human Cognitive Architecture: Proposed Operationalizations and Gradings*

1. Behave as an (almost) arbitrary function of the environment
   –Is it computationally universal with failure?
   Classical Connectionism: Mixed; ACT-R: Better
2. Operate in real time
   –Given its timing assumptions, can it respond as fast as humans?
   Classical Connectionism: Worse; ACT-R: Best
3. Exhibit rational, i.e., effective adaptive behavior
   –Does the system yield functional behavior in the real world?
   Classical Connectionism: Better; ACT-R: Better
4. Use vast amounts of knowledge about the environment
   –How does the size of the knowledge base affect performance?
   Classical Connectionism: Worse; ACT-R: Mixed
5. Behave robustly in the face of error, the unexpected, and the unknown
   –Can it produce cognitive agents that successfully inhabit dynamic environments?
   Classical Connectionism: Mixed; ACT-R: Better
6. Integrate diverse knowledge
   –Is it capable of common examples of intellectual combination?
   Classical Connectionism: Worse; ACT-R: Mixed
7. Use (natural) language
   –Is it ready to take a test of language proficiency?
   Classical Connectionism: Better; ACT-R: Worse
8. Exhibit self-awareness and a sense of self
   –Can it produce functional accounts of phenomena that re-reflect consciousness
   Classical Connectionism: Worse; ACT-R: Worse
9. Learn from its environment
   –Can it produce the variety of human learning
   Classical Connectionism: Better; ACT-R: Better
10. Acquire capabilities through development
   –Can it account for developmental phenomena?
   Classical Connectionism: Better; ACT-R: Worse
11. Arise through evolution
   –Does the theory relate to evolutionary and comparative considerations?
   Classical Connectionism: Worst; ACT-R: Worst
12. Be realizable within the brain
   –Do the components of the theory exhaustively map onto brain processes?
   Classical Connectionism: Best; ACT-R: Worse

sality, and that it was the most important criterion. He devoted the major portion of the 1980 paper to proving that the symbol system he was describing satisfied this criterion. For Newell, the flexibility in human behavior implied computational universality. With modern fashion so emphasizing evolutionarily-prepared, specialized cognitive functions, it is worthwhile to remind ourselves that one of the most distinguishing human features is the ability to learn to perform almost arbitrary cognitive tasks to high degrees of expertise. Whether it is air-traffic control or computer programming, people are capable of performing with high facility cognitive activities that had no anticipation in human evolutionary history. Moreover, humans are the only species that show anything like this cognitive plasticity.

Newell recognized the difficulties he was creating in identifying this capability with formal notions of universal computability. For example, memory limitations prevent humans from being equivalent to Turing machines (with their infinite tapes), and their frequent slips prevent people from displaying perfect behavior. However, he recognized the true flexibility in human cognition that deserved this identification with computational universality, even as the modern computer is characterized as a Turing-equivalent device despite its physical limitations and occasional errors.

While computational universality is a fact of human cognition, it should not be seen in opposition to the idea of specialized facilities for performing various cognitive functions – even a computer can have specialized processors. Moreover, it should not be seen in opposition to the view that some things are much easier for people to learn and do than others. This has been stressed in the linguistic domain where it is argued that there are "natural languages" that are much easier to learn than nonnatural languages. However, this lesson is perhaps even clearer in the world of human artifacts, like air-traffic control systems or computer applications, where some systems are much easier to learn and to use than others. Although there are many complaints about how poorly designed some of these systems are, the artifacts that are in common use are only the tip of the iceberg with respect to unnatural systems. While humans may approach computational universality, it is only a tiny fraction of the computable functions that humans find feasible to acquire and perform.

Grading: If a theory is well specified, it should be relatively straightforward to determine whether it is computationally universal or not. As already noted, this is not to say that the theory should claim that people will find everything equally easy or that human performance will ever be error free.

2.2. Real-time performance

It is not enough for a theory of cognition to explain the great flexibility of human cognition, it must also explain how humans can do this in what Newell referred to as "real time," which means human time. As the understanding of the neural underpinnings of human cognition increases, the field faces increasing constraints on its proposals as to what can be done in a fixed period of time. Real time is a constraint on learning as well as performance. It is no good to be able to learn something in principle if it takes lifetimes to do that learning.

Grading: If a theory comes with well-specified constraints on how fast its processes can proceed, then it is relatively trivial to determine whether it can achieve real time for any specific case of human cognition. It is not possible to prove that the theory satisfies the real-time constraint for all cases of human cognition, so one must be content with looking at specific cases.

2.3. Adaptive behavior

Humans do not just perform marvelous intellectual computations. The computations that they choose to perform serve their needs. As Anderson (1991) argued, there are two levels at which one can address adaptivity. At one level, one can look at the basic processes of an architecture, such as association formation, and ask whether and how they serve a useful function. At another level, one can look at how the whole system is put together and ask whether its overall computation serves to meet human needs.

Grading: What protected the short-term memory models that Newell complained about from the conclusion that they were not adaptive was that they were not part of more completely specified systems. Consequently, one could not determine their implications beyond the laboratory experiments they addressed, where adaptivity was not an issue. However, if one has a more completely specified theory like Newell's Soar system (Newell 1990), one can explore whether the mechanism enables behavior that would be functional in the real world. Although such assessment is not trivial, it can be achieved as shown by analyses such as those exemplified in Oaksford and Chater (1998) or Gigerenzer (2000).

2.4. Vast knowledge base

One key to human adaptivity is the vast amount of knowledge that can be called on. Probably what most distinguishes human cognition from various "expert systems" is the fact that humans have the knowledge necessary to act appropriately in so many situations. However, this vast knowledge base can create problems. Not all of the knowledge is equally reliable or equally relevant. What is relevant to the current situation can rapidly become irrelevant. There may be serious issues of successfully storing all the knowledge and retrieving the relevant knowledge in reasonable time.

Grading: To assess this criterion requires determining how performance changes with the scale of the knowledge base. Again, if the theory is well specified, this criterion is subject to formal analysis. Of course, one should not expect that size will have no effect on performance – as anyone knows who has tried to learn the names of students in a class of 200.

2.5. Dynamic behavior

Living in the real world is not like solving a puzzle like the Tower of Hanoi. The world can change in ways that we do not expect and do not control. Even human efforts to control the world by acting on it can have unexpected effects. People make mistakes and have to recover. The ability to deal with a dynamic and unpredictable environment is a precondition to survival for all organisms. Given the complexity of the environments that humans have created for themselves, the need for dynamic behavior is one of the major cognitive stressors that they face. Dealing with dynamic behavior requires a theory of perception and action as well as

a theory of cognition. The work on situated cognition (e.g., Greeno 1989; Lave 1988; Suchman 1987) has emphasized how cognition arises in response to the structure of the external world. Advocates of this position sometimes argue that all there is to cognition is reaction to the external world. This is the symmetric error to the earlier view that cognition could ignore the external world (Clark 1998; 1999).

Grading: How does one create a test of how well a system deals with the "unexpected"? Certainly, the typical laboratory experiment does a poor job of putting this to the test. An appropriate test requires inserting these systems into uncontrolled environments. In this regard, a promising class of tests looks at cognitive agents built in these systems and inserted into real or synthetic environments. For example, Newell's Soar system successfully simulated pilots in an Air Force mission simulation that involved 5,000 agents including human pilots (Jones et al. 1999).

2.6. Knowledge integration

We have chosen to retitle this criterion. Newell referred to it as "Symbols and Abstractions," and his only comment on this criterion appeared in his 1990 book: "[The] [m]ind is able to use symbols and abstractions. We know that just from observing ourselves" (p. 19). He never seemed to acknowledge just how contentious this issue is, although he certainly expressed frustration (Newell 1992) that people did not "get" what he meant by a symbol. Newell did not mean external symbols like words and equations, about whose existence there can be little controversy. Rather, he was thinking about symbols like those instantiated in list-processing languages. Many of these "symbols" do not have any direct meaning, unlike the sense of symbols that one finds in philosophical discussions or computational efforts, as in Harnad (1990; 1994). Using symbols in Newell's sense, as a grading criterion, seems impossibly loaded. However, if we look to his definition of what a physical symbol does, we see a way to make this criterion fair:

> Symbols provide distal access to knowledge-bearing structures that are located physically elsewhere within the system. The requirement for distal access is a constraint on computing systems that arises from action always being physically local, coupled with only a finite amount of knowledge being encodable within a finite volume of space, coupled with the human mind's containing vast amounts of knowledge. Hence encoded knowledge must be spread out in space, whence it must be continually transported from where it is stored to where processing requires it. Symbols are the means that accomplish the required distal access. (Newell 1990, p. 427)

Symbols provide the means of bringing knowledge together to make the inferences that are most intimately tied to the notion of human intellect. Fodor (2000) refers to this kind of intellectual combination as "abduction" and is so taken by its wonder that he doubts whether standard computational theories of cognition (or any other current theoretical ideas for that matter) can possibly account for it.

In our view, in his statement of this criterion Newell confused mechanism with functionality. The functionality he is describing in the preceding passage is a capacity for intellectual combination. Therefore, to make this criterion consistent with the others (and not biased), we propose to cast it as achieving this capability. In point of fact, we think that when we understand the mechanism that achieves this capacity, it will turn out to involve symbols more or less in the sense Newell intended. (However, we do think there will be some surprises when we discover how the brain achieves these symbols.) Nonetheless, not to prejudge these matters, we simply render the sixth criterion as the capacity for intellectual combination.

Grading: To grade on this criterion we suggest judging whether the theory can produce those intellectual activities which are hallmarks of daily human capacity for intellectual combination – things like inference, induction, metaphor, and analogy. As Fodor (2000) notes, it is always possible to rig a system to produce any particular inference; the real challenge is to produce them all out of one system that is not set up to anticipate any. It is important, however, that this criterion not become a test of some romantic notion of the wonders of human cognition that actually almost never happen. There are limits to the normal capacity for intellectual combination, or else great intellectual discoveries would not be so rare. The system should to be able to reproduce the intellectual combinations that people display on a day-to-day basis.

2.7. Natural language

While most of the criteria on Newell's list could be questioned by some, it is hard to imagine anyone arguing that a complete theory of cognition need not address natural language. Newell and others have wondered about the degree to which natural language is the basis of human symbol manipulation versus the degree to which symbol manipulation is the basis for natural language. Newell took the view that language depends on symbol manipulation.

Grading: It is not obvious how to characterize the full dimensions of that functionality. As a partial but significant test, we suggest looking at those tests that society has set up as measures of language processing – something like the task of reading a passage and answering questions on it. This would involve parsing, comprehension, inference, and relating current text to past knowledge. This is not to give theories a free pass on other aspects of language processing such as partaking in a conversation, but one needs to focus on something in specifying the grading for this criterion.

2.8. Consciousness

Newell acknowledged the importance of consciousness to a full account of human cognition, although he felt compelled to remark that "it is not evident what functional role self-awareness plays in the total scheme of mind." We too have tended to regard consciousness as epiphenomenal, and it has not been directly addressed in the ACT-R theory. However, Newell is calling us to consider all the criteria and not pick and choose the ones to consider.

Grading: Cohen and Schooler (1997) have edited a volume aptly titled *Scientific Approaches to Consciousness,* which contains sections on subliminal perception, implicit learning and memory, and metacognitive processes. We suggest that the measure of a theory on this criterion is its ability to produce these phenomena in a way that explains why they are functional aspects of human cognition.

2.9. Learning

Learning seems to be another uncontroversial criterion for a theory of human cognition. A satisfactory theory of cog-

nition must account for humans' ability to acquire their competences.

Grading: It seems insufficient to grade a theory simply by asking whether the theory is capable of learning because people must be capable of many different kinds of learning. We suggest taking Squire's (1992) classification as a way of measuring whether the theory can account for the range of human learning. The major categories in Squire's classification are semantic memory, episodic memory, skills, priming, and conditioning. They may not be distinct theoretical categories, and there may be more kinds of learning, but these do represent much of the range of human learning.

2.10. Development

Development is the first of the three constraints that Newell listed for a cognitive architecture. Although in some hypothetical world one might imagine the capabilities associated with cognition emerging full blown, human cognition in the real world is constrained to unfold in an organism as it grows and responds to experience.

Grading: There is a problem in grading the developmental criterion which is like that for the language criteria – there seems no good characterization of the full dimensions of human development. In contrast to language, because human development is not a capability but rather a constraint, there are no common tests for the development constraint per se, although the world abounds with tests of how well our children are developing. In grading his own Soar theory on this criterion, Newell was left with asking whether it could account for specific cases of developmental progression (for instance, he considered how Soar might apply to the balance scale). We are unable to suggest anything better.

2.11. Evolution

Human cognitive abilities must have arisen through some evolutionary history. Some have proposed that various content-specific abilities, such as the ability to detect cheaters (Cosmides & Tooby 2000b) or certain constraints on natural language (e.g., Pinker 1994; Pinker & Bloom 1990), evolved at particular times in human evolutionary history. A variation on the evolutionary constraint is the comparative constraint. How is the architecture of human cognition different from that of other mammals? We have identified cognitive plasticity as one of the defining features of human cognition, and others have identified language as a defining feature. What is it about the human cognitive system that underlies its distinct cognitive properties?

Grading: Newell expressed some puzzlement at how the evolutionary constraint should apply. Grading the evolutionary constraint is deeply problematical because of the paucity of the data on the evolution of human cognition. In contrast to judging how adaptive human cognition is in an environment (Criterion 3), reconstruction of a history of selectional pressures seems vulnerable to becoming the construction of a just-so story (Fodor 2000; Gould & Lewontin 1979). The best we can do is ask loosely how the theory relates to evolutionary and comparative considerations.

2.12. Brain

The last constraint collapses two similar criteria in Newell (1980) and corresponds to one of the criteria in Newell (1990). Newell took seriously the idea of the neural implementation of cognition. The timing of his Soar system was determined by his understanding of how it might be neurally implemented. The last decade has seen a major increase in the degree to which data about the functioning of specific brain areas are used to constrain theories of cognition.

Grading: Establishing that a theory is adequate here seems to require both an enumeration and a proof. The enumeration would be a mapping of the components of the cognitive architecture onto brain structures, and the proof would be that the computation of the brain structures match the computation of the assigned components of the architecture. There is possibly an exhaustive requirement as well – that no brain structure is left unaccounted for. Unfortunately, knowledge of brain function has not advanced to the point where one can fully implement either the enumeration or the proof of a computational match. However, there is enough knowledge to partially implement such a test, and even as a partial test, it is quite demanding.

2.13. Conclusions

It might seem reckless to open any theory to an evaluation on such a broad set of criteria as those in Table 1. However, if one is going to propose a cognitive architecture, it is impossible to avoid such an evaluation as Newell (1992) discovered with respect to Soar. As Vere (1992) described it, because a cognitive architecture aspires to give an integrated account of cognition, it will be subjected to the "attack of the killer bees" – each subfield to which the architecture is applied is "resolutely defended against intruders with improper pheromones." Vere proposed creating a "Cognitive Decathlon"

> to create a sociological environment in which work on integrated cognitive systems can prosper. Systems entering the Cognitive Decathlon are judged, perhaps figuratively, based on a cumulative score of their performance in each cognitive "event." The contestants do not have to beat all of the narrower systems in their one specialty event, but compete against other well-rounded cognitive systems. (Vere 1992, p. 460)

This target article could be viewed as a proposal for the events in the decathlon and an initial calibration of the scoring for the events by providing an evaluation of two current theories, classical connectionism and ACT-R.

While classical connectionism and ACT-R offer some interesting contrasts when graded by Newell's criteria, both of these two theories are ones that have done rather well when measured by the traditional standard in psychology of correspondence to the data of particular laboratory experiments. Thus, we are not bringing to this grading what are sometimes called *artificial intelligence* theories. It is not as if we were testing "Deep Blue" as a theory of human chess, but it is as if we were asking of a theory of human chess that it be capable of playing chess – at least in principle, if not in practice.

3. Classical connectionism

Classical connectionism is the cognitively and computationally modern heir to behaviorism. Both behaviorism and connectionism have been very explicit about what they accept and what they reject. Both focus heavily on learning

and emphasize how behavior (or cognition) arises as an adaptive response to the structure of experience (Criteria 3 and 9 in Newell's list). Both reject any abstractions (Newell's original Criterion 6, which we have revamped for evaluation) except as a matter of verbal behavior (Criterion 8). Being cognitively modern, connectionism, however, is quite comfortable in addressing issues of consciousness (Criterion 8), whereas behaviorism often explicitly rejected consciousness. The most devastating criticisms of behaviorism focused on its computational adequacy, and it is here that the distinction between connectionism and behaviorism is clearest. Modern connectionism established that it did not have the inadequacies that had been shown for the earlier Perceptrons (Minsky & Papert 1969). Connectionists developed a system that can be shown to be computationally equivalent to a Turing machine (Hartley 2000; Hartley & Szu 1987; Hornik et al. 1989; Siegelman & Sontag 1992) and endowed it with learning algorithms that could be shown to be universal function approximaters (Clark 1998; 1999).

However, as history would have it, connectionism did not replace behaviorism. Rather, there was an intervening era in which an abstract information-processing conception of mind dominated. This manifested itself perhaps most strongly in the linguistic ideas surrounding Chomsky (e.g., 1965) and the information-processing models surrounding Newell and Simon (e.g., 1972). These were two rather different paradigms, with the Chomskian approach emphasizing innate knowledge only indirectly affecting behavior, and the Newell and Simon approach emphasizing the mental steps directly underlying the performance of a cognitive task. However, both approaches deemphasized learning (Criterion 9) and emphasized cognitive abstractions (Original Criterion 6). Thus, when modern connectionism arose, the targets of its criticisms were the *symbols* and *rules* of these theories. It chose to focus largely on linguistic tasks emphasized by the Chomskian approach and was relatively silent on the problem-solving tasks emphasized by the Newell and Simon approach. Connectionism effectively challenged three of the most prized claims of the Chomskian approach – that linguistic overgeneralizations were evidence for abstract rules (Brown 1973), that initial syntactic parsing was performed by an encapsulated syntactic parser (Fodor 1983), and that it was impossible to acquire language without the help of an innate language-acquisition device (Chomsky 1965). We will briefly review each of these points, but at the outset we want to emphasize that these connectionist demonstrations were significant because they established that a theory without language-specific features had functionalities which some claimed it could not have. Thus, the issues were very much a matter of functionality in the spirit of the Newell test.

Rumelhart and McClelland's (1986b) past-tense model has become one of the most famous of the connectionist models of language processing. They showed that by learning associations between the phonological representations of stems and past tense, it was possible to produce a model that made overgeneralizations without building any rules into it. This attracted a great many critiques, and, while the fundamental demonstration of generalization without rules stands, it is acknowledged by all to be seriously flawed as a model of the process of past-tense generation by children. Many more recent and more adequate connectionist models (some reviewed in Elman et al. 1996) have been proposed, and many of these have tried to use the backpropogation learning algorithm.

While early research suggested that syntax was in some way separate from general knowledge and experience (Ferreira & Clifton 1986), further research has suggested that syntax is quite penetrable by all sorts of semantic considerations and in particular the statistics of various constructions. Models like those of MacDonald et al. (1994) are quite successful in predicting the parses of ambiguous sentences. There is also ample evidence now for syntactic priming (e.g., Bock 1986; Bock & Griffin 2000) – that people tend to use the syntactic constructions they have recently heard. There are also now sociolinguistic data (reviewed in Matessa 2001) showing that the social reinforcement contingencies shape the constructions that one will use. Statistical approaches to natural-language processing have been quite successful (Collins 1999; Magerman 1995). While these approaches are only sometimes connectionist models, they establish that the statistics of language can be valuable in untangling the meaning of language.

While one might imagine these statistical demonstrations being shrugged off as mere performance factors, the more fundamental challenges have concerned whether the syntax of natural language actually is beyond the power of connectionist networks to learn. "Proofs" of the inadequacy of behaviorism have concerned their inability to handle the computational complexity of the syntax of natural language (e.g., Bever et al. 1968). Elman (1995) used a recurrent network to predict plausible continuations for sentence fragments like *boys who chase dogs see girls* that contain multiple embeddings. This was achieved by essentially having hidden units that encoded states reflecting the past words in the sentence.

The preceding discussion has focused on connectionism's account of natural language, because that is where the issue of the capability of connectionist accounts has received the most attention. However, connectionist approaches have their most natural applications to tasks that are more directly a matter of perceptual classification or continuous tuning of motor output. Some of the most successful connectionist models have involved things like letter recognition (McClelland & Rumelhart 1981). Pattern classification and motor tuning underlie some of the more successful "performance" applications of connectionism including NETtalk (Sejnowski & Rosenberg 1987), which converts orthographic representation of words into a code suitable for use with a speech synthesizer; TD-Gammon (Tesauro 2002), a world-champion backgammon program; and ALVINN (Autonomous Land Vehicle In a Neural Network) (Pomerleau 1991), which was able to drive a vehicle on real roads.

So far we have used the term *connectionism* loosely, and it is used in the field to refer to a wide variety of often incompatible theoretical perspectives. Nonetheless, there is a consistency in the connectionist systems behind the successes just reviewed. To provide a roughly coherent framework for evaluation, we will focus on what has been called *classical connectionism.* Classical connectionism is the class of neural network models that satisfy the following requirements: feedforward or recurrent network topology, simple unit activation functions such as sigmoid or radial basis functions, and local weight-tuning rules such as backpropagation or Boltzmann learning algorithms. This defini-

tion reflects both the core and the bulk of existing neural network models while presenting a coherent computational specification. It is a restriction with consequence. For instance, the proofs of Turing equivalence include assumptions not in the spirit of classical connectionism and often involving nonstandard constructs.

4. ACT-R

4.1. ACT-R's history of development

While ACT-R is a theory of cognition rather than a framework of allied efforts like connectionism, it has a family-resemblance aspect too, in that it is just the current manifestation of a sequence of theories stretching back to Anderson (1976), when we first proposed how a subsymbolic activation-based memory could interact with a symbolic system of production rules. The early years of that project were concerned with developing a neurally plausible theory of the activation processes and an adequate theory of production rule learning, resulting in the ACT* theory (Anderson 1983). The next ten years saw numerous applications of the theory, a development of a technology for effective computer simulations, and an understanding of how the subsymbolic level served the adaptive function of tuning the system to the statistical structure of the environment (Anderson 1990). This resulted in the ACT-R version of the system (Anderson 1993), where the "R" denotes rational analysis.

Since the publication of ACT-R in 1993, a community of researchers has evolved around the theory. One major impact of this community has been to help prepare ACT-R to take the Newell Test by applying it to a broad range of issues. ACT had traditionally been a theory of "higher-level" cognition and largely ignored perception and action. However, as members of the ACT-R research community became increasingly concerned with timing and dynamic behavior (Newell's Criteria 2 and 5), it was necessary to address attentional issues about how the perceptual and motor systems interact with the cognitive system. This has led to the development of ACT-R/PM (PM for perceptual-motor) (Byrne & Anderson 1998), based in considerable part on the perceptual-motor components of EPIC (Meyer & Kieras 1997). This target article focuses on ACT-R 5.0, which is an integration of the ACT-R 4.0 described in Anderson and Lebiere (1998) and ACT-R/PM.

4.2. General description of ACT-R

Since it is a reasonable assumption that ACT-R is less well known than classical connectionism, we will give it a fuller description, although the reader should refer to Anderson and Lebiere (1998) for more formal specifications and the basic equations. Figure 1 displays the current architecture of ACT-R. The flow of cognition in the system is in response to the current goal, currently active information from declarative memory, information attended to in perceptual modules (vision and audition are implemented), and the current state of motor modules (hand and speech are implemented). These components (goal, declarative memory, perceptual, and motor modules) hold the information that the productions can access in *buffers,* and these buffers serve much the same function as the subsystems of Baddeley's (1986) working-memory theory. In response to the current state of these buffers, a production is selected and executed. The central box in Figure 1 reflects the processes that determine which production to fire. There are two distinct subprocesses – pattern matching to decide which productions are applicable, and conflict resolution to select among these applicable productions. While all productions are compared in parallel, a single production is selected to fire. The selected production can cause changes in the current goal, make a retrieval request of declarative memory, shift attention, or call for new motor actions. Unlike EPIC, ACT-R is a serial-bottleneck theory of cognition (Pashler 1998) in which parallel cognitive, perceptual, and motor modules must interact through a serial process of production execution.

The architecture in Figure 1 is an abstraction from the neural level, but nonetheless it is possible to give tentative neural correlates. The motor and perceptual modules correspond to associated cortical areas; the current goal, to frontal areas; and declarative memory, to posterior cortical and hippocampal areas. There is evidence (Wise et al. 1996) that the striatum receives activation from the full cortex and recognizes patterns of cortical activation. These recognized patterns are gated by other structures in the basal ganglia (particularly the internal segment of the globus pallidus and the substantia nigra pars reticulata) (Frank et al. 2000) and the frontal cortex to select an appropriate action. Thus, one might associate the striatum with the pattern-recognition component of the production selection and the basal ganglia structures and the frontal cortex with the conflict resolution.

ACT-R is a hybrid architecture in the sense that it has both symbolic and subsymbolic aspects. The symbolic aspects involve declarative chunks and procedural production rules. The declarative chunks are the knowledge-representation units that reside in declarative memory, and the production rules are responsible for the control of cognition. Access to these symbolic structures is determined by a subsymbolic level of neural-like activation quantities. Part of the insight of the rational analysis is that the declarative and procedural structures, by their nature, need to be guided by two different quantities. Access to declarative chunks is controlled by an activation quantity that reflects the probability that the chunk will need to be retrieved. In the case of production rules, choice among competing rules is controlled by their utilities, which are estimates of the rule's probability of success and cost in leading to the goal. These estimates are based on the past reinforcement history of the production rule.

The activation of a chunk is critical in determining its retrieval from declarative memory. A number of factors determine the level of activation of a chunk in declarative memory:

1. The recency and frequency of usage of a chunk will determine its base-level activation. This base-level activation represents the probability (actually, the log odds) that a chunk is needed, and the estimates provided for by ACT-R's learning equations represent the probabilities in the environment (see Anderson 1993, Ch. 4, for examples).

2. Added to this base-level activation is an associative component that reflects the priming that the chunk might receive from elements currently in the focus of attention. The associations among chunks are learned on the basis of past patterns of retrieval according to a Bayesian framework.

ED:
Quality of art okay through out?
Please advise.
–Comp.

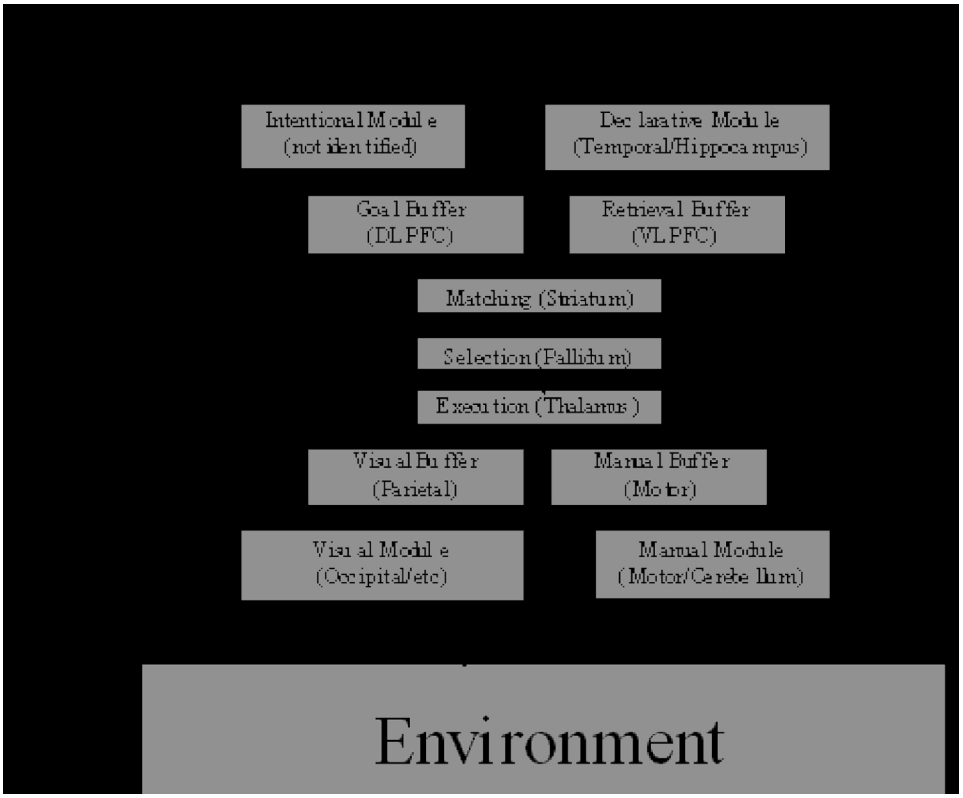


Figure 1. ACT-R Architecture

3. The activation controlled by factors 1 and 2 is modulated by the degree to which the chunk matches current retrieval specifications. Thus, for example, a chunk that encodes a similar situation to the current one will receive some activation. This partially matching component in ACT-R allows it to produce the soft, graceful behavior characteristic of human cognition. Similarities among chunks serve a similar purpose to distributed representations in connectionist networks.

4. The activation quantities are fundamentally noisy, so there is some variability in which chunk is most active, producing a stochasticity in behavior.

The activation of a chunk determines the time to retrieve it. Also, when multiple chunks can be retrieved, the most active one is selected. This principle, combined with variability in activation, produces predictions for the probability of recall according to the softmax Boltzmann distribution (Ackley et al. 1985; Hinton & Sejnowski 1986). These latency and probability functions in conjunction with the activation processes have led to a wide variety of successful models of verbal learning (e.g., Anderson et al. 1998a; Anderson & Reder 1999a).

Each production rule has a real-valued utility that is calculated from estimates of the cost and probability of reaching the goal if that production rule is chosen. ACT-R's learning mechanisms constantly update these estimates based on experience. If multiple production rules are applicable to a certain goal, the production rule with the highest utility is selected. This selection process is noisy, so the production with the highest utility has the greatest probability of being selected, but other productions get opportunities as well. This may produce errors or suboptimal behavior, but also allows the system to explore knowledge and strategies that are still evolving. The ACT-R theory of utility learning has been tested in numerous studies of strategy selection and strategy learning (e.g., Lovett 1998).

In addition to the learning mechanisms that update activation and expected outcome, ACT-R can also learn new chunks and production rules. New chunks are learned automatically: Each time a goal is completed or a new percept is encountered, it is added to declarative memory. New production rules are learned by combining existing production rules. The circumstance for learning a new production rule is that two rules fire one after another, with the first rule retrieving a chunk from memory. A new production rule is formed that combines the two into a macro-rule but eliminates the retrieval. Therefore, everything in an ACT-R model (chunks, productions, activations, and utilities) is learnable.

The symbolic level is not merely a poor approximation to the subsymbolic level as claimed by Rumelhart and McClelland (1986b) and Smolensky (1988); rather, it provides the essential structure of cognition. It might seem strange that neural computation should just so happen to satisfy the well-formedness constraints required to correspond to the symbolic level of a system like ACT-R. This would indeed be miraculous if the brain started out as an unstructured net that had to organize itself just in response to experience. However, as illustrated in the tentative brain correspondences for ACT-R components and in the following description of ACT-RN, the symbolic structure emerges out of the structure of the brain. For example, just as the two eyes converge in adjacent columns in the visual cortex to enable stereopsis, a similar convergence of information

(perhaps in the basal ganglia) would permit the condition of a production rule to be learned.

4.3. ACT-RN

ACT-R is not in opposition to classical connectionism except in connectionism's rejection of a symbolic level. Although strategically ACT-R models tend to be developed at a larger grain size than connectionist models, we do think these models could be realized by the kinds of computation proposed by connectionism. Lebiere and Anderson (1993) instantiated this belief in a system called ACT-RN that attempted to implement ACT-R using standard connectionist concepts. We will briefly review ACT-RN here because it shows how production system constructs can be compatible with neural computation.

ACT-R consists of two key memories – a declarative memory and a procedural memory. Figure 2 illustrates how ACT-RN implements declarative chunks. The system has separate memories for each different type of chunk – for example, addition facts are represented by one type memory, whereas integers are represented by a separate type memory. Each type memory is implemented as a special version of Hopfield nets (Hopfield 1982). A chunk in ACT-R consists of a unique identifier called the header, together with a number of slots, each containing a value, which can be the identifier of another chunk. Each slot, as well as the chunk identifier itself, is represented by a separate pool of units, thereby achieving a distributed representation. A chunk is represented in the pattern of connections between these pools of units. Instead of having complete connectivity among all pools, the slots are only connected to the header and vice versa. Retrieval involves activating patterns in some of the pools and trying to fill in the remaining patterns corresponding to the retrieved chunk. If some slot patterns are activated, they are mapped to the header units to retrieve the chunk identifier that most closely matches these contents (path 1 in Fig. 2). Then, the header is mapped back to the slots to fill the remaining values (path 5). If the header pattern is specified, then the step corresponding to path 1 is omitted.

To ensure optimal retrieval, it is necessary to "clean" the header. This can be achieved in a number of ways. One is to implement the header itself as an associative memory. We chose instead to connect the header to a pool of units called the chunk layer in which each unit represented a chunk, achieving a localist representation (path 2). The header units are connected to all the units in the chunk layer. The pattern of weights leading to a particular localist unit in the chunk layer corresponds to the representation of that chunk in the header. By assembling these chunk-layer units in a winner-take-all network (path 3), the chunk with the representation closest to the retrieved header ultimately wins. That chunk's representation is then reinforced in the header (path 4). A similar mechanism is described in Dolan and Smolensky (1989). The initial activation level of the winning chunk is related to the number of iterations in the chunk-layer needed to find a clear winner. This maps onto retrieval time in ACT-R, as derived in Anderson and Lebiere (1998, Ch. 3 Appendix).

ACT-RN provides a different view of the symbolic side of ACT-R. As is apparent in Figure 2, a chunk is nothing more or less than a pattern of connections between the chunk identifier and its slots.

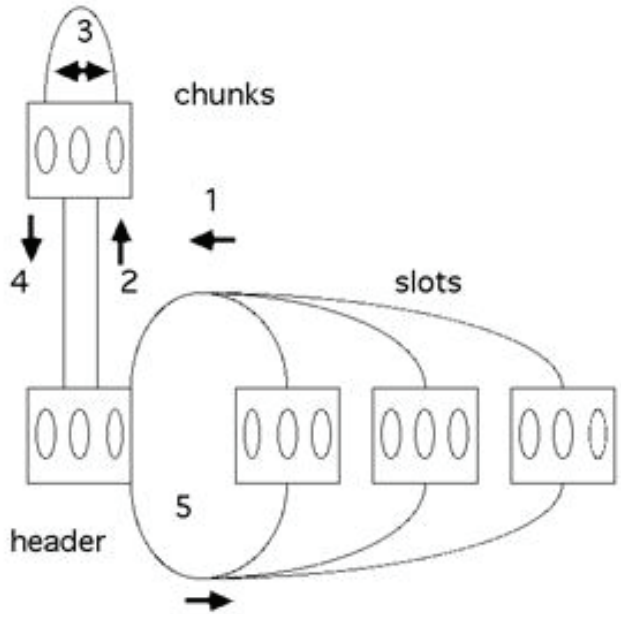


Figure 2. Declarative Memory in ACT-RN

ACT-R is a goal-oriented system. To implement this, ACT-RN has a central memory (which probably should be identified with dorsolateral prefrontal cortex), which at all times contains the current goal chunk (Fig. 3) with connections to and from each type memory. Central memory consists of pools of units, where each pool encodes a slot value of the goal. There was an optional goal stack (represented in Fig. 3), but we do not use a goal stack in ACT-R anymore. Productions in ACT-RN retrieve information from a type memory and deposit it in central memory. Such a production might retrieve from an addition memory the sum of two digits held in central memory. For example, given the goal of adding 2 and 3, a production would copy to the addition-fact memory the chunks 2 and 3 in the proper slots by enabling (gating) the proper connections between central memory and that type memory, let the memory retrieve the sum 5, and then transfer that chunk to the appropriate goal slot.

To provide control over production firing, ACT-RN needs a way to decide not only what is to be transferred where but also under what conditions. In ACT-RN, that task is achieved by gating units (which might be identified with gating functions associated with basal ganglia). Each gating unit implements a particular production and has incoming connections from central memory that reflect the goal constraints on the left-hand side of that production. For example, suppose goal slot S is required to have as value chunk C in production P. To implement this, the connections between S and the gating unit for P will be the representation for C, with an appropriate threshold. At each production cycle, all the gating units are activated by the current state of central memory, and a winner-take-all competition selects the production to fire.

Note that production rules in ACT-RN are basically rules for enabling pathways back and forth between a central goal memory and the various declarative memory modules. Thus, production rules are not really structures that are stored in particular locations but are rather specifications of information transfer. ACT-RN also offers an interesting

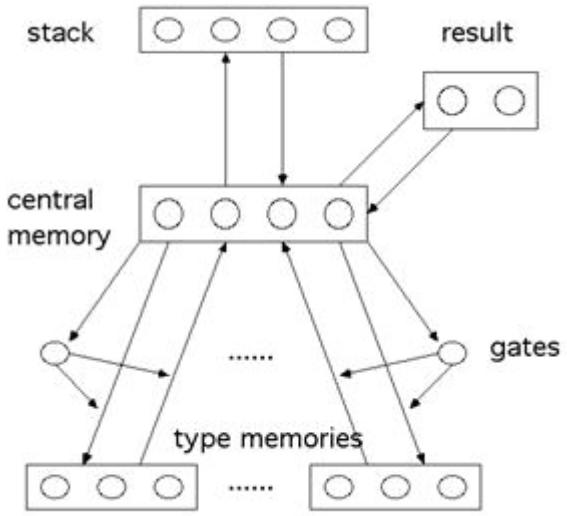


Figure 3. Procedural Memory in ACT-RN

perspective on the variables (see Marcus [2001] for a discussion of variables in connectionist models) that appear in production rules and their bindings. The effect of such bindings is basically to copy values from the goal to declarative memory and back again. This is achieved in ACT-RN without having any explicit variables or an explicit process of variable binding. Thus, while the computational power that is represented by variables is critical, one can have this without the commitment to explicit variables or a process of variable binding.

4.4. Learning past tense in ACT

Recently, Taatgen (2001; Taatgen & Anderson 2002) has developed a successful ACT-R model for learning the past-tense in English, which provides an interesting comparison point with the connectionist models. Unlike many past-tense models, it learns based on the actual frequency of words in natural language, learns without feedback, and makes the appropriate set of generalizations. While the reader should go to the original papers for details, we will briefly describe this model because the past tense has been critical in the connectionist-symbolic debate. It also serves to illustrate all of the ACT-R learning mechanisms working at once.

The model posits that children initially approach the task of past-tense generation with two strategies. Given a particular word like "give," they can either try to retrieve the past tense for that word or try to retrieve some other example of a past tense (e.g., "live"–"lived") and try to apply this by analogy to the current case. Eventually, through the production-rule learning mechanisms in ACT-R, the analogy process will be converted into a production rule that generatively applies the past-tense rule. Once the past-tense rule is learned, the generation of past tenses will largely be determined by a competition between the general rule and retrieval of specific cases. Thus, ACT-R is basically a dual-route model of past-tense generation, where both routes are implemented by production rules. The rule-based approach depends on general production rules, whereas the exemplar approach depends on the retrieval of declarative chunks by production rules that implement an instance-based strategy. This choice between retrieval and rule-based computation is a general theme in ACT-R models and is closely related to Logan's model of skill acquisition (Logan 1988). It has been used in a model of cognitive arithmetic (Lebiere 1998) and in models for a number of laboratory tasks (Anderson & Betz 2001; Lerch et al. 1999; Wallach & Lebiere, in press).

The general past-tense rule, once discovered by analogy, gradually enters the competition as the system learns that this new rule is widely applicable. This gradual entry, which depends on ACT-R's subsymbolic utility-learning mechanisms, is responsible for the onset of overgeneralization. Although this onset is not all-or-none in either the model or the data, it is a relatively rapid transition in both model and data and corresponds to the first turn in the U-shaped function. However, as this is happening, the ACT-R model is encountering and strengthening the declarative representations of exceptions to the general rule. Retrieval of the exceptions comes to counteract the overgeneralizations. Retrieval of exceptions is preferred because they tend to be shorter and phonetically more regular (Burzio 1999) than regular past tenses. Growth in this retrieval process corresponds to the second turn in the U-shaped function and is much more gradual – again, both in model and data. Note that the Taatgen model, unlike many other past-tense models, does not make artificial assumptions about frequency of exposure but learns, given a presentation schedule of words (both from the environment and its own generations) like that actually encountered by children. Its ability to reproduce the relatively rapid onset of overgeneralization and slow extinction depends critically on both its symbolic and subsymbolic learning mechanisms. Symbolically, it is learning general production rules and declarative representations of exceptions. Subsymbolically, it is learning the utility of these production rules and the activation strengths of the declarative chunks.

Beyond just reproducing the U-shaped function, the ACT-R model explains why exceptions should be high-frequency words. There are two aspects to this explanation. First, only high-frequency words develop enough base-level activation to be retrieved. Indeed, the theory predicts how frequent a word has to be in order to maintain an exception. Less obviously, the model explains why so many high-frequency words actually end up as exceptions. This is because the greater efficiency of the irregular form promotes its adoption according to the utility calculations of ACT-R. In another model that basically invents its own past-tense grammar without input from the environment, Taatgen showed that it develops one or more past-tense rules for low-frequency words but tends to adopt more efficient irregular forms for high-frequency words. In the ACT-R economy the greater phonological efficiency of the irregular form justifies its maintenance in declarative memory if it is of sufficiently high frequency.

Note that the model receives no feedback on the past tenses it generates, unlike most other models but in apparent correspondence with the facts about child language learning. However, it receives input from the environment in the form of the past tenses it hears, and this input influences the base-level activation of the past-tense forms in declarative memory. The model also uses its own past-tense generations as input to declarative memory and can learn its own errors (a phenomenon also noted in cognitive arithmetic – Siegler 1988). The amount of overgeneralization

displayed by the model is sensitive to the ratio of input it receives from the environment to its own past-tense generations.

While the model fully depends on the existence of rules and symbols, it also critically depends on the subsymbolic properties of ACT-R to produce the graded effects. This eclectic position enables the model to achieve a number of other features not achieved by many other models:

1. It does not have to rely on artificial assumptions about presentation frequency.
2. It does not need corrective feedback on its own generations.
3. It explains why irregular forms tend to be of high frequency and why high-frequency words tend to be irregular.
4. It correctly predicts that novel words will receive regular past tenses.
5. It predicts the gradual onset of overgeneralization and its much more gradual extinction.

4.5. What ACT-R doesn't do

Sometimes the suspicion is stated that ACT-R is a general computational system that can be programmed to do anything. To address this issue, we would like to specify four senses in which the system falls short of that.

First of all, ACT-R is also a system with strong limitations. Because of prior constraints on its timing, there are limits on how fast it can process material. The perceptual and motor components of the system take fixed time – for instance, it would be impossible for the system to press a button in response to a visual stimulus in less than 100 msec. At a cognitive level, it has limits on the rate of production selection and retrieval of declarative memory. This has been a major challenge in our theories of natural-language processing (Anderson et al. 2001; Budiu & Anderson, submitted), and it remains an open issue whether the general architecture can process language at the speed with which humans process it. The serial bottleneck in production selection causes all sorts of limitations – for example, the theory cannot perform mental addition and multiplication together as fast as it can perform either singly (Byrne & Anderson 2001). Limitations in memory mean that the system cannot remember a long list of digits presented at a 1-second rate (at least without having acquired a large repertoire of mnemonic skills (Chase & Ericsson 1982)). The limitations actually are successes of ACT-R as a theory of human cognition, since humans appear to display these limitations (with the issue about language open). However, their existence means that we cannot just "program" arbitrary models in ACT-R.

Second, there are also numerous mechanisms of cognition not yet incorporated into ACT-R, although there may be no in-principle reason why they cannot be incorporated. For example, ACT-R lacks any theory of the processes of speech perception or speech production. This is not without consequence for the claims of the theory. For instance, the just reviewed past-tense model made critical claims about the phonological costs of various past-tense inflections but these were just assertions not derived from the model. The absence of a phonological component makes it difficult to extend the model to making predictions about other inflectional constructions. Among other domains for which ACT-R seems to be lacking adequate mechanisms are perceptual recognition, mental imagery, emotion, and motivation. We do not think these absences reflect anything fundamentally incompatible between what the theory claims and what people can do, but that possibility always exists until it is shown how such mechanisms could be added in a consistent way to ACT-R.

Third, there are also numerous domains of great interest to cognitive science that have yet to be addressed by ACT-R. Many of these are concerned with perceptual recognition where the mechanisms of the theory are weak or lacking (the perceptual modules in ACT-R are really theories of perceptual attention) but others just reflect the failure of ACT-R researchers to take up the topic. For example, there are no ACT-R models of deductive reasoning tasks. Also, within domains that ACT-R has addressed, there are important phenomena left unaddressed. For example, although there is an ACT-R model of recognition memory (Anderson et al. 1998a), it has not addressed the *remember-know* distinction (Reder et al. 2000) or data on latency distributions (Ratcliff et al. 1999). It is not clear whether these open issues reflect simply things that ACT-R researchers have not addressed, or whether they are fundamental failings of the theory. For example, Reder (personal communication) has argued that the failure to address the *remember-know* distinction reflects the fact that ACT-R cannot deal with a whole class of metacognitive judgments because it does not have conscious access to its own subsymbolic quantities.

Finally, there is a set of implementation issues rife among researchers in the ACT-R community. We do not want to belabor them, as they have an esoteric flavor, but just to acknowledge that such things exist, we name a few (and ACT-R researchers will recognize them): avoiding repeatedly retrieving a chunk because retrievals strengthen the chunk, creating new chunk types, producing a latency function that adequately reflects competition among similar memories, and setting the temporal bounds for utility learning.

5. Grading classical connectionism and ACT-R according to the Newell Test

Having described Newell's criteria and the two theories, it is now time to apply these criteria to grading the theories. Regrettably, we were not able to state the Newell criteria in such a way that their satisfaction would be entirely a matter of objective fact. The problems are perhaps most grievous in the cases of the developmental and evolutionary criteria, where it is hard to name anything that would be a satisfactory measure, and one is largely left with subjective judgment. Even with hard criteria like computational universality, there is uncertainty about what approaches are really in keeping with the spirit of an architecture and how complete an answer particular solutions yield.

We had originally proposed a letter-grading scheme for the criteria that we applied to ACT-R. However, we were persuaded in the review process to apply the criteria to classical connectionism by the argument that the criteria became more meaningful when one sees how they apply to two rather different theories. It did not make sense to be competitively grading one's own theory alongside another one, and therefore we decided to change the grading into a rough within-theory rank ordering of how well that theory did on those criteria. That is, we will be rating how well that theory has done on a particular criterion, relative to how well it has done on other criteria (not relative to the other theory). Therefore, we will be using the following grading:

Best: The criteria on which that theory has done the best
Better: Four criteria on which that theory has done better
Mixed: Two criteria on which that theory has the most mixed record
Worse: Four criteria on which that theory has done worse
Worst: The criteria on which that theory has done the worst

This is actually more in keeping with our intentions for the Newell Test than the original letter grading because it focuses on directions for improving a given theory rather than declaring a winner. Of course, the reader is free to apply an absolute grading scheme to these two theories or any other.

5.1. Flexible behavior

Grading: *Connectionism: Mixed*
ACT-R: Better

To do well on this criterion requires that the theory achieve an interesting balance: It must be capable of computing any function, but have breakdowns in doing so, and find some functions easier to compute than others. It has been shown possible to implement a Turing machine in connectionism, but not in the spirit of classical connectionism. Breakdowns in the execution of a sequence of actions would be quite common (Botvinick & Plaut, submitted). There is a balance between capability and limitation in classical connectionism, but we and some others (e.g., Marcus 2001) believe that this is an uneven balance in favor of limitations. It is not clear that complex, sequentially organized, hierarchical behavior can be adequately produced in classical connectionistic systems, and there seems to a paucity of demonstrations. Indeed, a number of the high-performance connectionist systems have been explicitly augmented with handcrafted representations (Tesauro 2002) and symbolic capabilities (Pomerleau et al. 1991). Moreover, the connectionist models that do exist tend to be single-task models. However, the essence of computational universality is that one system can give rise to an unbounded set of very different behaviors.

ACT-R does well on this criterion in no small part because it was exactly this criterion that has most driven the design of this model. ACT-R, except for its subsymbolic limitations, is Turing equivalent, as are most production systems (proof for an early version of ACT appears in Anderson 1976). However, because of variability and memory errors, ACT-R frequently deviates from the prescribed course of its symbolic processing. This shows up, for example, in ACT-R models for the Tower of Hanoi (Anderson & Douglass 2001; Altmann & Trafton 2002), where it is shown that memory failures produce deviations from well-learned algorithms at just those points where a number of goals have to be recalled. (These are also the points where humans produce such deviations.) Nonetheless, ACT-R has also been shown to be capable of producing complex sequential behavior such as operation of an air-traffic control system (Taatgen 2002). The functions that it finds easy to compute are those with enough support from the environment to enable behavior to be corrected when it deviates from the main course.

5.2. Real-time performance

Grading: *Connectionism: Worse*
ACT-R: Best

Connectionist processing often has a poorly defined (or just poor) relationship to the demands of real-time processing. The mapping of processing to reaction time is inconsistent and often quite arbitrary; for example, some relatively arbitrary function of the unit activation is often proposed (e.g., Rumelhart & McClelland 1982). For feedforward models that depend on synchronous updates across the various levels of units, it is fundamentally inconsistent to assume that the time for a unit to reach full activation is a function of that activation. The natural factor would seem to be the number of cycles, but even when this is adopted, it is often arbitrarily scaled (e.g., a linear function of number of cycles with a negative intercept; see Plaut & Booth 2000). Another problem is that connectionist systems typically only model a single step of the full task (the main mapping) and do not account for the timing effects produced by other aspects of the task such as perceptual or motor. Finally, with respect to learning time, the number of epochs that it takes, to acquire an ability, maps poorly to the learning of humans (Schneider & Oliver 1991). This last fact is one of the major motivations for the development of hybrid models.

One of the great strengths of ACT-R is that every processing step comes with a commitment to the time it will take. It is not possible to produce an ACT-R model without timing predictions. Of course, it is no small matter that ACT-R not only makes predictions about processing time, but that these happen to be correct over a wide range of phenomena. As knowledge accumulates in the ACT-R community, these timing predictions are becoming a priori predictions. As one sign of this, in recent classes that we have taught, undergraduates at CMU were producing models that predicted absolute, as well as relative times, with no parameter estimation. In addition to performance time, ACT-R makes predictions about learning time. In a number of simulations, ACT-R was able to learn competences in human time (i.e., given as many training experiences as humans). This includes cognitive arithmetic (Lebiere 1998), past-tense formations (Taatgen & Anderson 2002), and backgammon (Sanner et al. 2000). ACT-R's treatment of time provides one answer to Roberts and Pashler's (2000) critique of model-fitting efforts. These researchers view it as so easy to fit a model to data that it is at best an uninformative activity. Their claim that it is easy or uninformative can be challenged on many grounds, but the ACT-R effort highlights the fact that one need not be fitting one experiment or paradigm in isolation.

5.3. Adaptive behavior

Grading: *Connectionism: Better*
ACT-R: Better

The positions of connectionism and ACT-R on this criterion are quite similar. Both have made efforts, often Bayesian in character (McClelland & Chappell1998), to have their underlying learning rules tune the system to the statistical structure of the environment. This is quite central to ACT-R because its subsymbolic level derives from the earlier rational analysis of cognition (Anderson 1990). However, adaptivity is not a direct function of these subsymbolic equations but rather is a function of the overall behavior of the system. ACT-R lacks an overall analysis of adaptivity, including an analysis of how the goals selected by ACT-R are biologically significant. An overall analysis is similarly lacking in classical connectionism.

The reader will recall that Newell raised the issue of the adaptivity of limitations like short-term memory. In ACT-

R, short-term memory effects are produced by decay of base-level activations. ACT-R's use of base-level activations delivers a computational embodiment of the rational analysis of Anderson (1990), which claimed that such loss of information with time reflected an adaptive response to the statistics of the environment where information loses its relevance with time. Thus, ACT-R has implemented this rational analysis in its activation computations and has shown that the resulting system satisfies Newell's requirement that it be functional.

5.4. Vast knowledge base

Grading: *Connectionism: Worse*
ACT-R: Mixed

Just because a system works well on small problems, one has no guarantee that it will do so on large problems. There have been numerous analyses of the scaling properties of neural networks. In models like NETtalk, it has shown how a great deal of knowledge can be captured in the connections among units, but that this depends on a similarity in the input-output mappings. One of the notorious problems with connectionism is the phenomenon of catastrophic interference whereby new knowledge overwrites old knowledge (McCloskey & Cohen 1989; Ratcliff 1990). Connectionists are much aware of this problem and numerous research efforts (e.g., McClelland et al. 1995) address it.

In ACT-R, the function of the subsymbolic computations is to identify the right chunks and productions out of a large data base, and the rational analysis provides a "proof" of the performance of these computations. The success of these computations has been demonstrated in "life-time" learning of cognitive arithmetic (Lebiere 1998) and past-tense learning (Taatgen 2001). However, they have been models of limited domains, and the knowledge base has been relatively small. There have been no ACT-R models of performance with large knowledge bases approaching human size. The subsymbolic mechanisms are motivated to work well with large knowledge bases, but that is no guarantee that they will. The one case of dealing with a large knowledge base in ACT-R is the effort (Emond, in preparation) to implement WordNet (Fellbaum 1998) in ACT-R, which involves more than 400,000 chunks, but this implementation awaits more analysis.

5.5. Dynamic behavior

Grading: *Connectionism: Mixed*
ACT-R: Better

Connectionism has some notable models of interaction with the environment such as ALVINN and its successors, which were able to drive a vehicle, although it was primarily used to drive in fairly safe predictable conditions (e.g., straight highway driving) and was disabled in challenging conditions (interchanges, perhaps even lane changes). However, as exemplified in this model, connectionism's conception of the connections among perception, cognition, and action is pretty ad hoc, and most connectionist models of perception, cognition, and action are isolated, without the architectural structure to close the loop, especially in timing specifications. McClelland's (1979) Cascade model offers an interesting conception of how behavior might progress from perception to action, but this conception has not actually been carried through in models that operate in dynamic environments.

Many ACT-R models have closed the loop, particularly in dealing with dynamic environments like driving, air traffic control, simulation of warfare activities, collaborative problem solving with humans, control of dynamic systems like power plants, and game playing. These are all domains where the behavior of the external system is unpredictable. These simulations take advantage of both ACT-R's ability to learn and the perceptual-motor modules that provide a model of human attention. However, ACT-R is only beginning to deal with tasks that stress its ability to respond to task interruption. Most ACT-R models have been largely focused on single goals.

5.6. Knowledge integration

Grading: *Connectionism: Worse*
ACT-R: Mixed

We operationalized Newell's symbolic criterion as achieving the intellectual combination that he thought physical symbols were needed for. Although ACT-R does use physical symbols more or less in Newell's sense, this does not guarantee that it has the necessary capacity for intellectual combination. There are demonstrations of it making inference (Anderson et al. 2001), performing induction (Haverty et al. 2000), metaphor (Budiu 2001), and analogy (Salvucci & Anderson 2001), and these all do depend on its symbol manipulation. However, these are all small-scale, circumscribed demonstrations, and we would not be surprised if Fodor found them less than convincing.

Such models have not been as forthcoming from classical connectionism (Browne & Sun 2001). A relatively well-known connectionist model of analogy (Hummel & Holyoak 1998) goes beyond classical connectionist methods to achieve variable binding by means of temporal synchrony. The Marcus demonstration of infants' learning rules has become something of a challenge for connectionist networks. It is a relatively modest example of intellectual combination – recognizing that elements occurring in different positions need to be identical to fit a rule and representing that as a constraint on novel input. The intellectual elements being combined are simply sounds in the same string. Still, it remains a challenge to classical connectionism, and some classical connectionists (e.g., McClelland & Plaut 1999) have chosen instead to question whether the phenomenon is real.

5.7. Natural language

Grading: *Connectionism: Better*
ACT-R: Worse

Connectionism has a well-articulated conception of how natural language is achieved, and many notable models that instantiate this conception. However, despite efforts like Elman's, it is a long way from providing an adequate account of human command of the complex syntactic structure of natural language. Connectionist models are hardly ready to take the SAT. ACT-R's treatment of natural language is fragmentary. It has provided models for a number of natural-language phenomena including parsing (Lewis 1999), use of syntactic cues (Matessa & Anderson 2000), learning of inflections (Taatgen 2001), and metaphor (Budiu 2001).

ACT-R and connectionism take opposite sides on the chicken-and-egg question about the relationship between symbols and natural language that Newell and others wondered about: Natural-language processing depends in part on ACT-R's symbolic capabilities, and it is not the case that natural-language processing forms the basis of the symbolic capabilities, nor is it equivalent to symbolic processing. However, classical connectionists are quite explicit that whatever might appear to be symbolic reasoning really depends on linguistic symbols like words or other formal symbols like equations.

5.8. Consciousness

Grading: Connectionism: Worse
ACT-R: Worse

The stances of connectionism and ACT-R on consciousness are rather similar. They both have models (e.g., Cleeremans 1993; Wallach & Lebiere 2000; in press) that treat one of the core phenomena – implicit memory – in the discussion of consciousness. However, neither have offered an analysis of subliminal perception or metacognition. With respect to functionality of the implicit/explicit distinction, ACT-R holds that implicit memory represents the subsymbolic information that controls the access to explicit declarative knowledge. To require that this also be explicit, would be inefficient and invite infinite regress.

ACT-R does imply an interpretation of consciousness. Essentially, what people are potentially conscious of is contained in ACT-R's set of buffers in Figure 1 – the current goal, the current information retrieved from long-term memory, the current information attended in the various sensory modalities, and the state of various motor modules. There are probably other buffers not yet represented in ACT-R to encode internal states like pain, hunger, and various pleasures. The activity of consciousness is the processing of these buffer contents by production rules. There is no Cartesian Theater (Dennett 1991; Dennett & Kinsbourne 1995) in ACT-R. ACT-R is aware of the contents of the buffers only as they are used by the production rules.

5.9. Learning

Grading: Connectionism: Better
ACT-R: Better

A great deal of effort has gone into thinking about and modeling learning in both connectionist models and ACT-R. However, learning is such a key issue and so enormous a problem that both have much more to do. They display complementary strengths and weaknesses. While connectionism has accounts to offer of phenomena in semantic memory like semantic dementia (Rogers & McClelland 2003), ACT-R has been able to provide detailed accounts of the kind of discrete learning characteristic of episodic memory such as the learning of lists or associations (Anderson et al. 1998a; Anderson & Reder 1999a). Whereas there are connectionist accounts of phenomena in perceptual and motor learning, ACT-R offers accounts of the learning of cognitive skills like mathematical problem solving. Whereas there are connectionist accounts of perceptual priming, there are ACT-R accounts of associative priming. The situation with respect to conditioning is interesting. On the one hand, the basic connectionist learning rules have a clear relationship to some of the basic learning rules proposed in the conditioning literature, such as the Rescorla-Wagner rule (see Anderson [2000] for a discussion). On the other hand, known deficits in such learning rules have been used to argue that at least in the case of humans, these inferences are better understood as more complex causal reasoning (Schoppek 2001).

5.10. Development

Grading: Connectionism: Better
ACT-R: Worse

As with language, development is an area that has seen a major coherent connectionist treatment but only spotty efforts from ACT-R. Connectionism treats development as basically a learning process, but one that is constrained by the architecture of the brain and the timing of brain development. The connectionist treatment of development is in some ways less problematic than its treatment of learning because connectionist learning naturally produces the slow changes characteristic of human development. Classical connectionism takes a clear stand on the empiricist–nativist debate, rejecting what it calls representational nativism.

In contrast, there is not a well-developed ACT-R position on how cognition develops. Some aspects of a theory of cognitive development are starting to emerge in the guise of cognitive models of a number of developmental tasks and phenomena (Emond & Ferres 2001; Jones et al. 2000; Simon 1998; submitted; Taatgen & Anderson 2002; van Rijn et al. 2000). The emerging theory is one that models child cognition in the same architecture as adult cognition and that sees development as just a matter of regular learning. Related to this is an emerging model of individual differences (Jongman & Taatgen 1999; Lovett et al. 2000) that relates them to a parameter in ACT-R that controls the ability of associative activation to modulate behavior by context. Anderson et al. (1998b) argue that development might be accompanied by an increase in this parameter.

5.11. Evolution

Grading: Connectionism: Worst
ACT-R: Worst

Both theories, by virtue of their analysis of the Bayesian basis of the mechanisms of cognition, have something to say about the adaptive function of cognition (as they were credited with under Criterion 3), but neither has much to say about how the evolution of the human mind occurred. Both theories basically instantiate the puzzlement expressed by Newell as to how to approach this topic.

We noted earlier that cognitive plasticity seems a distinguishing feature of the human species. What enables this plasticity in the architecture? More than anything else, ACT-R's goal memory enables it to abstract and retain the critical state information needed to execute complex cognitive procedures. In principle, such state maintenance could be achieved using other buffers – speaking to oneself, storing and retrieving state information from declarative memory, writing things down, and so forth. However, this would be almost as awkward as getting computational universality from a single-tape Turing machine, besides being very error-prone and time-consuming. A large expansion of the frontal cortex, which is associated with goal manipulations, occurred in humans. Of course, the frontal cortex is somewhat expanded in other primates, and it would probably be

unwise to claim that human cognitive plasticity is totally discontinuous from that of other species.

5.12. Brain

Grading: Connectionism: Best
ACT-R: Worse

Classical connectionism, as advertised, presents a strong position on how the mind is implemented in the brain. Of course, there is the frequently expressed question of whether the brain that classical connectionism assumes happens to correspond to the human brain. Assumptions of equipotentiality and the backprop algorithm are frequent targets for such criticisms, and many nonclassical connectionist approaches take these problems as starting points for their efforts.

There is a partial theory about how ACT-R is instantiated in the brain. ACT-RN has established the neural plausibility of the ACT-R computations, and we have indicated rough neural correlates for the architectural components. Recently completed neural imaging studies (Anderson et al. 2003; Fincham et al. 2002; Sohn et al. 2000) have confirmed the mapping of ACT-R processes onto specific brain regions (e.g., goal manipulations onto the dorsolateral prefrontal cortex). There is also an ACT-R model of frontal patient deficits (Kimberg & Farah 1993). However, there is not the systematic development that is characteristic of classical connectionism. While we are optimistic that further effort will improve ACT-R's performance on this criteria, it is not there yet.

6. Conclusion

Probably others will question the grading and argue that certain criteria need to be re-ranked for one or both of the theoretical positions. Many of the arguments will be legitimate complaints, and we are likely to respond by either defending the grading, or conceding an adjustment in it. However, the main point of this target article is that the theories should be evaluated on all 12 criteria, and the grades point to where the theories need more work.

Speaking for ACT-R, where will an attempt to improve lead? In the case of some areas like language and development, it appears that improving the score simply comes down to adopting the connectionist strategy of applying ACT-R in depth to more empirical targets of opportunity. We could be surprised, but so far these applications have not fundamentally impacted the architecture. The efforts to extend ACT-R to account for dynamic behavior through perception and action yielded a quite different outcome. At first, ACT-R/PM was just an importation, largely from EPIC (Meyer & Kieras 1997) to provide input and output to ACT-R's cognitive engine. However, it became clear that ACT-R's cognitive components (the retrieval and goal buffers in Fig. 1) should be redesigned to be more like the sensory and motor buffers. This led to a system that more successfully met the dynamic behavior criterion and has much future promise in this regard. Thus, incorporating the perceptual and motor modules fundamentally changed the architecture. We suspect that similar fundamental changes will occur as ACT-R is extended to deal further with the brain criterion.

Where would attention to these criteria take classical connectionism? First, we should acknowledge that it is not clear that classical connectionists will pay attention to these criteria or even acknowledge that the criteria are reasonable. However, if they were to try to achieve the criteria, we suspect that it would move connectionism to a concern with more complex tasks and symbolic processing. We would not be surprised if it took them in a direction of a theory more like ACT-R, even as ACT-R has moved in a direction that is more compatible with connectionism. Indeed, many attempts have been made recently to integrate connectionist and symbolic mechanisms into hybrid systems (Sun 1994; 2002). More generally, if researchers of all theoretical persuasions did try to pursue a broad range of criteria, we believe that distinctions among theoretical positions would dissolve and psychology will finally provide "the kind of encompassing of its subject matter – the behavior of man – that we all posit as a characteristic of a mature science" (Newell 1973, p. 288).

NOTE

1. The complete list of published ACT-R models between 1997 and 2002 is available from the ACT-R home page at: **act.psy.cmu.edu**

ACKNOWLEDGMENTS

Preparation of this manuscript was supported by ONR grant N00014–96–1-C491. We would like to thank Gary Marcus and Alex Petrov for their comments on this manuscript. We would also like to thank Jay McClelland and David Plaut for many relevant and helpful discussions, although we note they explicitly chose to absent themselves from any participation that could be taken as adopting a stance on anything in this paper.

Open Peer Commentary

Commentary submitted by the qualified professional readership of this journal will be considered for publication in a later issue as Continuing Commentary on this article. Integrative overviews and syntheses are especially encouraged.

Newell's list

Joseph Agassi

Department of Philosophy, Tel-Aviv University, Tel-Aviv 69978, Israel.
agass@post.tau.ac.il http://www.tau.ac.il/~agass/

Abstract: Newell wanted a theory of cognition to abide by some explicit criteria, here called the Newell Test. The test differs from the Turing Test because it is explicit. The Newell Test will include the Turing Test if its characterization of cognition is complete. It is not. Its use here is open-ended: A system that does not pass it well invites improvement.

Alan Newell asserted that an adequate theory of a functioning system of human cognition should abide by some explicit criteria, and he offered a list of such criteria. The list includes characteristics such as flexible, adaptive behavior; possession of a vast knowledge base; and the ability to integrate knowledge, use a natural language, and learn. The target article authors say that, although this list is not complete, it certainly is "enough to avoid theoretical myopia" (sect. 1, para. 2). Hardly: Myopia is the outcome of the claim for knowledge of natural languages and learning sufficient to per-

mit decision as to whether a given theory of cognition captures them adequately. We just do not know that much as yet.

The authors say that the criteria deserve "greater scientific prominence." They therefore try to "evaluate theories by how well they do at meeting" the criteria (sect. 1, para. 4). This may be premature. Whether it is, depends on the merit of Newell's idea more than on its applications. So, it requires examination. What the authors call the Newell Test is a test not of Newell's idea but of the theories that should agree with it – provided that it is valid. Is it? How are we to judge this?

Anderson & Lebiere (A&L) apply Newell's Test to two new ideas that are controversial, so the application cannot be such a test. Hence, their work is begging the question: Some test of it is required to show that it deserves a "greater scientific prominence."

"Newell is calling us to consider all the criteria and not pick and choose the ones to consider" (sect. 2.8). This remark renders the whole venture too questionable. The authors make it apropos discussion of the criterion of consciousness.

> Newell acknowledged the importance of consciousness to a full account of human cognition, although he felt compelled to remark that "it is not evident what functional role self-awareness plays in the total scheme of mind." We too have tended to regard consciousness as epiphenomenal . . . (sect. 2.8)

This is very shaky. Whether consciousness is or is not epiphenomenal is a red herring: It is an empirical fact that in many cases cognitive conduct differs depending on whether it is accompanied with consciousness or not, and the question may arise, should a system emulating human consciousness reflect this fact? Importantly, Turing's celebrated idea of the Turing Test is designed to avoid this question altogether.

The authors examine two sets, classical connectionism and ACT-R. Classical connectionism is a computerized version of behaviorism. ACT-R is "a theory of higher-level cognition," "a subsymbolic activation-based memory" able "to interact with a symbolic system of production rules"; the R in ACT-R "denotes rational analysis" (sect. 4.1, first paragraph). The two sets, then, are artificial intelligence or expert-systems programs. The authors report a claim that classical connectionism passes the Turing Test. Presumably they disagree. The same holds for ACT-R. "ACT-R, but for its subsymbolic limitations, is Turing equivalent, as are most production systems" and "(proof for an early version of ACT [is due to] Anderson . . .)" (sect. 5.1, para. 2). This is a bit cryptic; I will explain the difference between the Turing and the Newell Tests in the following paragraph.

The Turing Test was meant to render the mind-body problem empirically decidable. Were there a computer program that could fool an expert, Turing suggested, then it would be empirically indistinguishable from humans, and so the attribution to humans of a metaphysical soul would be redundant. Because Newell's criteria depict human characteristics, any interlocutor who can pass the Turing Test should certainly possess them, because the inability to exhibit any human characteristic the like of which Newell mentions would expose the impostor. And yet, the Turing Test is implicit and Newell's Test is explicit. This permits finding a partial success in passing the Newell Test. But, to be an explicit version of the Turing Test, the Newell Test must refer to a complete list of characteristics. We do not have this, and the Turing Test may be preferred just because it leaves this task to the experts who wish to test the humanity of their enigmatic interlocutor. Consequently, a Turing Test can never be decisive: Both expert and programmer can improve on prior situations and thus deem failure a merely temporary setback. True, the Turing Test is generally deemed possibly decisive, and, being a thought-experiment, actually decisive. Some writers, notably Daniel Dennett, claim that only exorbitant costs prevent the construction of a machine that will pass the Turing Test. That machine, then, should certainly be able to pass the Newell Test with flying colors. It is a pity that A&L do not refer to this claim and expose it as a sham. If they are any close to being right, they should be able to do so with ease.

The interesting aspect of the target article is that it is open-ended: Whenever the system A&L advocate, which is ACT-R, does not pass the examination as well as they wish, they recommend trying an improvement, leading to a retest. They should observe that such a move may be two-pronged. They refer to the improvement of the ability of a program to abide by the theory of flexibility; adaptive behavior; and the ability to integrate knowledge, use a natural language, and learn. They should not ignore the need to improve on these theories. When they refer to natural languages or to learning, they view the connectionist idea of them as more satisfactory than that of ACT-R, because it is more complete. Yet, whatever completeness is exactly, it is not enough: We seek explanations, and so to accept axiomatically what we want to understand is not good enough. We still do not know what a natural language is and how we learn; and we do not begin to understand these. Let me end with an insight of David Marr that should not be forgotten. Emulation is helpful for the understanding but is no substitute for it; sometimes, the very success of emulation, Marr (1982) observed, renders it less useful as a problematic one. We want understanding, not mere emulation.

Think globally, ask functionally

Erik M. Altmann

Department of Psychology, Michigan State University, East Lansing, MI 48824. **ema@msu.edu** **http://www.msu.edu/~ema**

Abstract: The notion of functionality is appropriately central to the Newell Test but is also critical at a lower level, in development of cognitive sub-theories. I illustrate, on one hand, how far this principle is from general acceptance among verbal theoreticians, and, on the other hand, how simulation models (here implemented within ACT-R) seem to drive the functional question automatically.

Anderson & Newell (A&L) have been carrying integrative cognitive theory, in shifts, for the past 30 years or so (if one goes back to Newell 1973). We are fortunate that Anderson is young; formulating dichotomous questions – seeing the trees but not the forest – may be the modal tenure procedure in psychology departments today, but perhaps in another generation it will be acceptable not to conduct new experiments at all but simply to integrate old data into increasingly complete computational models.

In the meantime, how can we avoid theoretical myopia in our daily research? Applying the Newell Test is well and good once a decade or so, with that many years' interim progress available to assess it. In terms of the next chunk of publishable research, however, it's useful to have more immediate guidance.

Central to the Newell Test is the idea of functionality: A theory has to explain how the cognitive system accomplishes some particular function. Among the Newell Test criteria, this function is high level, related in some relatively direct way to the fitness of the organism. However, as one develops micro-theories within a larger theory, functionality is still a relevant question; one can ask, for each process within a model, what it is for. Its outputs could, for example, be necessary inputs for another process in a chain that leads ultimately to accomplishing the task at hand. Or, one could ask whether each behavioral measure reflects a distinct process at all; perhaps it reflects a side effect of some other functionally necessary process. In both cases, it is difficult if not impossible to address the functional question without a precise representation of the processes one is talking about. In practice, this implies a computational simulation.

How does functionality play out at the level of the micro-theory that is the next chunk of publishable research? Curiously, even at this level, functionality seems to be regarded as optional, if not actually vulgar. A&L raise the example of short-term memory constructs (and Newell's frustration over them), but let's have a newer one, if only to see what might have changed. In the domain of ex-

ecutive control, there is a burgeoning literature on "switch cost" – the time cost associated with switching to a different task, as compared to performing the same task over again. One regularity to have emerged is that switch cost is difficult to erase; even with time and motivation to prepare for the other task, people are slower on the first trial under that task than on the second. The dominant theoretical account of this residual switch cost is arguably the "stimulus cued completion" hypothesis of Rogers and Monsell (1995, p. 224):

> This hypothesis proposes that an endogenous act of control deployed before onset of the stimulus can achieve only part of the process of task-set reconfiguration. Completion of the reconfiguration is triggered only by, and must wait upon, the presentation of a task-associated stimulus.

In terms of functionality, this hypothesis is vacuous. It need not be; one could ask how the system might benefit from stimulus-cued completion. For example, one could propose a benefit to the system hedging its bets and waiting to complete the reconfiguration process until there is evidence (in the form of the trial stimulus) that the new task set will be needed. One could then try to formulate scenarios in which this benefit would actually be realized and evaluate them for plausibility, or perhaps even against existing data. None of this was attempted by Rogers and Monsell, or by authors since who have invoked stimulus-cued completion as an explanatory construct. Call this a working definition of theoretical myopia: a "hypothesis" that merely relabels an empirical phenomenon.

In a subsequent ACT-R model, Sohn and Anderson (2001) explain residual switch cost in terms of stochasticity. Their model contains a "switching" production that retrieves the new task from memory and installs it in the system's focus of attention. Selection of productions is, like most other cognitive processes, subject to noise, which explains why this production is not always selected in advance of stimulus onset. Functionally, it can be selected after stimulus onset, though must be selected before response selection. This account is an improvement; it makes predictions (in terms of response-time variability), and it explains residual switch cost as a side-effect of the noise that accompanies any communication channel.

One could go further and ask, does residual switch cost reflect a process that directly contributes in some way to task performance? In another ACT-R model, Gray and I proposed that residual switch cost reflects a redundant task-encoding process that affects quality control (Altmann & Gray 2000). (Initial task encoding activates a memory trace for the current task, but noisily; redundant task encoding catches and properly strengthens memory traces that were initially weakly encoded.) The proof of functionality lay in Monte Carlo simulations showing that overall performance accuracy was higher with this redundant phase than without.

Are Sohn and Anderson right, or are Altmann and Gray? We have not found a behavioral test; perhaps neuroimaging will someday afford a diagnostic. I would predict, however, that the stimulus-cued completion hypothesis will not find its way into a precisely formulated cognitive theory, micro or otherwise, unless relevant functional questions are posed first.

ACKNOWLEDGMENT

This work was supported by an Office of Naval Research Grant N00014–03–1–0063.

The Newell Test should commit to diagnosing dysfunctions

William J. Clancey

Computational Sciences Division, MS 269–3, NASA Ames Research Center, Moffett Field, CA 94035. **william.j.clancey@nasa.gov**
http://bill.clancey.name

Abstract: "Conceptual coordination" analysis bridges connectionism and symbolic approaches by positing a "process memory" by which categories are physically coordinated (as neural networks) in time. Focusing on dysfunctions and odd behaviors, like slips, reveals the function of consciousness, especially constructive processes that are often taken for granted, which are different from conventional programming constructs. Newell strongly endorsed identifying architectural limits; the heuristic of "diagnose unusual behaviors" will provide targets of opportunity that greatly strengthens the Newell Test.

Anderson & Lebiere's (A&L's) article evaluates cognitive theories by relating them to the criteria of functionality derived from Newell. Suppose that the Newell Test (NT) has all the right categories, but still requires a significant architectural change for theoretical progress. I claim that "conceptual coordination" (CC) (Clancey 1999a) provides a better theory of memory, and that, without committing to explaining cognitive dysfunctions, NT would not provide sufficient heuristic guidance for leading in this direction.

Conceptual coordination (CC) hypothesizes that the store, retrieve, and copy memory mechanism is not how the brain works. Instead, all neural categorizations are activated, composed, and sequenced "in place," with the assumption that sufficient (latent) physical connections exist to enable necessary links to be formed (physically constructed) at run time (i.e., when a behavior or experience occurs). For example, if comprehending a natural language sentence requires that a noun phrase be incorporated in different ways, it is not moved or copied but is physically connected by activation of (perhaps heretofore unused) neural links. Effectively, Newell's "distal access" is accomplished by a capability to hold a categorization active and encapsulate it (like a pointer) so that it can be incorporated in different ways in a single construction. The no-copying constraint turns out to be extremely powerful for explaining a wide variety of odd behaviors, including speaking and typing slips, perceptual aspects of analogy formation, developmental "felt paths," multimodal discontinuity in dreams, and language comprehension limitations. CC thus specifies a cognitive architecture that bridges connectionist and symbolic concerns; and it relates well to the NT criteria for which ACT-R scores weakest – development, consciousness, language, and the brain. To illustrate, I provide a diagnostic analysis of an autistic phenomenon and then relate this back to how NT can be improved.

In CC analysis, a diagram notation is used to represent a behavior sequence, which corresponds in natural language to the conceptualization of a sentence. For example, according to Baron-Cohen (1996), an autistic child can conceptualize "I stroke the cat that drinks the milk." In one form of the CC notation, a slanting line to the right represents categorizations activated sequentially in time (e.g., "I – stroke" in Figure 1). Another sequence may qualify a categorization (e.g., "the cat – drinks" qualifies "stroke"). This pattern of sequences with qualifying details forming compositions of sequences occurs throughout CC analysis. The essential idea in CC is to understand how categories (both perceptual and higher-order categorizations of sequences and compositions of them) are related in time to constitute conscious experience (Clancey1999a).

The challenge is to understand why an autistic child finds it problematic to conceptualize "I see the cat that sees the mouse." A traditional view is that the child lacks social understanding. But CC analysis suggests a mechanistic limitation in the child's ability to physically sequence and compose categories. Relating to other agents requires being able to construct a second-order conceptualization that relates the child's activity to the other agent's activity. Figure 2 shows the CC notation for the required construction.

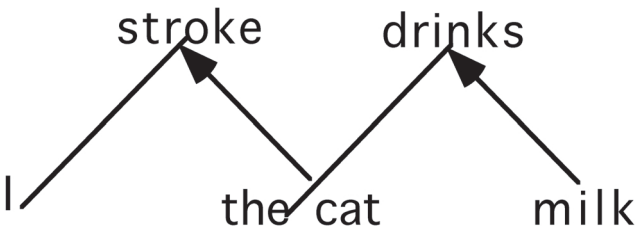


Figure 1 (Clancey). Unproblematic: "I stroke the cat that drinks the milk."

The statement (the conceptualization being constructed) involves a triadic relation: I see the cat, the cat sees the mouse, and I see the mouse. There is one mouse that we are both seeing. Two "see" constructions are unified by identifying a detail (the mouse) as common to both. In effect, the child must conceive of a problem space (Clancey 1999a): A common categorization of an operand (mouse) enables categorization of multiple actions as being one action (seeing), an operator. Because the two actions are by different agents, accomplishing this identification integrates perspectives of *self* (what I am doing now) and *other* (what that object is doing now). Indeed, the conceptualization of agent appears to be inherent in this construction.

Put another way, two sequentially occurring conceptualizations (I see the cat; the cat sees the mouse) are held active and related: "I see the cat that sees the mouse" and "I see the mouse" become "I see that the cat sees the mouse" (i.e., the mouse that I am seeing). (The second-order relation is represented in Figure 2 by the solid arrow below "I see"). Conceiving this relation is tantamount to conceiving what joint action is. Barresi and Moore (1996) characterize this as "integrating third and first person information" (p. 148), and contrast it with (Figure 1) "embedding one third person representation in a separate first person frame" (p. 148). Related to Langacker's (1986) analysis, logical relations are not extra capabilities or meta "inference" capabilities, but generalizations of concrete accomplishments that arise through the capability to physically coordinate categories through identification, sequence, and composition in time. Mental operations are physical, subconscious processes, constrained by physical limits on how inclusion in new sequences can occur. The ability to hold two sequences active and relate them constitutes a certain kind of consciousness (e.g., not present in dreaming; Clancey 2000).

To summarize, the example requires relating sequential categorizations of seeing so that they become simultaneous; it exemplifies a second-order conceptualization of intentionality (my seeing is about your seeing; Clancey 1999b); and suggests that joint action requires being able to conceive the ideas we call operator and agent.

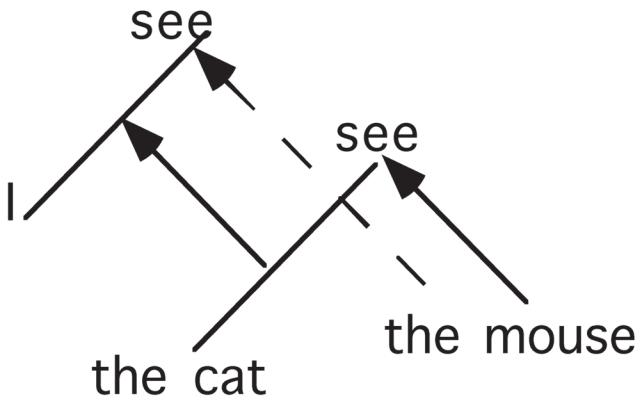


Figure 2 (Clancey). Problematic: "I see the cat that sees the mouse."

The pivotal heuristic in CC analysis is addressing *unusual* behaviors and experiences. These "targets of opportunity" appear to be de-emphasized by A&L's focus on normal behaviors "that people display on a day-to-day basis." For NT to provide heuristic guidance for discovering a theory like CC, grading for each criteria should include diagnosing unusual phenomena that everyone experiences (e.g., slips) and dysfunctions. For example, for the criteria of consciousness, we should direct theorization at explaining the phenomenology of dreaming, autism, compulsive-obsessive disorders, and the like. For natural language, include comprehension difficulties (e.g., subject relatives with center-embedded noun phrases; Clancey 1999a, Ch. 10). For development, explain how "felt paths" are constructed in children's learning (Ch. 5). For knowledge integration, explain slips (Ch. 6) and "seeing as" in analogy formation (Ch. 7). In this manner, learning in well-known architectures (e.g., MOPS, EPAM, SOAR) can be evaluated and the nature of problem spaces reformulated (Ch. 12).

The evolution criterion highlights the limitations of NT as stated. Rather than focusing on human evolution, this criterion should be about the evolution of cognition broadly construed, and hence should be inherently comparative across species (Clancey 1999b). Viewed this way, there is no "paucity of data," but rather a largely unexploited potential to make the study of animal cognition an integrated discipline with human problem solving. By including the heuristic "explain odd behavior" in the grading, we will naturally be guided to characterize and relate cognition in other primates, ravens, and the like. This is essential for relating "instinctive" mechanisms (e.g., weaving spider webs) to brain mechanisms, development, and learned higher-order categorizations (e.g., conceptualization of intentionality). A&L mention comparative considerations, but we should view this as a diagnostic problem, much as cognitive theories like ACT* have been used to explain students' different capabilities (Anderson et al. 1990). Furthermore, the research community should collect behaviors that have been heretofore ignored or poorly explained by computational theories and include them in the grading criteria.

Applying the Newell Test in this way – moving from the routine behaviors already handled more or less well, to diagnostic theories that relate aberrations to architectural variations – might bring symbolic and connectionist theories together and make the study of cognition a more mature science.

A complete theory of tests for a theory of mind must consider hierarchical complexity and stage

Michael Lamport Commons and Myra Sturgeon White
Department of Psychiatry, Harvard Medical School, Massachusetts Mental Health Center, Boston, MA 02115-6113. **Commons@tiac.net**
mswhite@fas.harvard.edu **http://www.tiac.net/~commons/**

Abstract: We distinguish traditional cognition theories from hierarchically complex stacked neural networks that meet many of Newell's criteria. The latter are flexible and can learn anything that a person can learn, by using their mistakes and successes the same way humans do. Shortcomings are due largely to limitations of current technology.

Anderson & Lebiere (A&L) raise important issues concerning criteria for evaluating the cognitive theories on which computational systems designed to simulate human intellectual abilities are based. Typically, cognitive theories are indirectly evaluated based on a theory's capacity to be translated into a computational system that produces correct answers or workable rules. The Newell 12-Criteria Test (1992; Newell & Simon 1963/1995) that A&L propose to measure theories with, makes an important move towards

measuring a theory's capacity to exhibit underlying behaviors supporting the expression of human cognitive processes.

We suggest a further dimension. Most cognitive theories are, like Athena, born fully formed, modeling the highest stages of development. However, human cognition is a product of developmental process. Humans learn to act by building one stage's actions on actions from previous stages, creating the capacity to perform ever more complex behaviors. Thus, to fully explain or model human intellectual capacity, hierarchical complexity must be factored into a theory. The *Model of Hierarchical Complexity* (MHC) (Commons et al. 1998) delineates these developmental changes (see Dawson 2002 for validity and reliability).

MHC identifies both sequences of development and reasons why development occurs from processes producing stage transition. It may be used to define complex human thought processes and computer systems simulating those processes. With this model, performed tasks are classified in terms of their order of hierarchical complexity using the following three main axioms (Commons et. al 1998). Actions at a higher order of hierarchical complexity

1. Are defined in terms of lower order actions;
2. Organize and transform lower stage actions;
3. Solve more complex problems through the nonarbitrary organization of actions.

The order of the hierarchical complexity of a task is determined by the number of its concatenation operations. An order-three task action has three concatenation operations and operates on output from order-two actions, which by definition has two concatenation operations and operates on an order-one task action. Increases in the hierarchical complexity of actions result from a dialectical process of stage transition. (Commons & Richards 2002).

To stimulate human intellectual capacities in computer systems, we design stacked neural networks that recapitulate the developmental process. This approach is necessary because currently we lack the knowledge to build into systems the myriad key behaviors formed during the developmental processes. Moreover, we lack the technology to identify the intricate web of neural connections that are created during the developmental process.

These stacked neural networks go through a series of stages analogous to those that occur during human intellectual development. Stages of development function as both theory and process in these systems. Actions (i.e., operations performed by networks resulting in a changed state of the system) are combined to perform tasks with more complex actions, permitting the performance of more complex tasks and thereby scaling up the power. The number of neural networks in a stack is the highest order of hierarchical complexity of task-required actions identified by the model. An example of a six-stage stacked neural network based on the model of hierarchical complexity (Table 1) follows.

Example. A system answers customer telephone calls, transferring them to the proper area within a large organization. Transfers are based on the customer's oral statements and responses to simple questions asked by the system. The system is capable of a three-year-old's language proficiency. A front-end recognition system translates customers' utterances (system inputs) into words that will serve as simple stimuli. It also measures time intervals between words.

Stacked neural networks based on the MHC meet many of Newell's criteria. They are flexible and can learn anything that a person can learn. They are adaptive because their responses are able to adjust when stimuli enter the stack at any level. They are dynamic in that they learn from their mistakes and successes. In the example, the system adjusts the weights throughout the stack of networks if a customer accepts or rejects the selected neural network location. Knowledge integration occurs throughout the networks in the stack. Moreover, networks based on the MHC learn in the same way as humans learn.

Some criteria are less easily met. Given current technology, neural networks cannot function in real time, are unable to trans-

Table 1 (Commons & White). *Stacked Neural Network (Example of Model of Hierarchical Complexity)*

Order of Hierarchical Complexity	What It Uses	What It Does
0. Calculatory	From Humans	Calculates and executes human written programs
1. Sensory and motor	Caller's utterances	A front-end speech recognition system translates customers' utterances into words. These "words" serve as simple stimuli to be detected.
2. Circular sensory motor	Words from speech recognition system	Forms open-ended classes consisting of groups contiguous individual words
3. Sensory-motor	Grouped contiguous speech segments	Labels and maps words to concepts. Networks are initially taught concepts that are central to the company environment: products and departments such as customer service, billing, and repair.
4. Nominal	Concept domains	Identifies and labels relationships between concept domains. Possible interconnections are trained based on the company's functions, products, and services. Interconnections are adjusted based on system success.
5. Sentential	Joint concept domains	Forms simple sentences and understands relationships between two or more named concepts. Finds possible locations to send customer's calls. Constructs statement on whether they want to be transferred to that department. Customer's acceptances or rejection feeds back to lower levels.

fer learning despite abilities to acquire a vast knowledge base, and cannot exhibit adult language skills. Whether we can build evolutions into systems – or even want to – is open to question. Finally, given our current limited understanding of the brain, we can only partially emulate brain function.

Nonclassical connectionism should enter the decathlon

Francisco Calvo Garzón

Department of Philosophy, Indiana University, Bloomington, IN, and University of Murcia, Facultad de Filosofía, Edif. Luis Vives, Campus de Espinardo Murcia 30100, Spain. **fjcalvo@um.es**

Abstract: In this commentary I explore nonclassical connectionism (NCC) as a coherent framework for evaluation in the spirit of the Newell Test. Focusing on knowledge integration, development, real-time performance, and flexible behavior, I argue that NCC's "within-theory rank ordering" would place subsymbolic modeling in a better position. Failure to adopt a symbolic level of thought cannot be interpreted as a weakness.

Granting Anderson & Lebiere's (A&L's) "cognitive decathlon" overall framework, and their proposed operationalizations and grading scheme for theory-evaluation, the aspects of their article that I address here concern the choice of contestants entering the decathlon, and, based on that choice, the exploration of nonclassical connectionism (NCC) as a coherent framework for evaluation in the spirit of the Newell Test. The range of classical connectionist architectures that A&L assess is confined to models that have a feedforward or a recurrent architecture, a locally supervised learning algorithm (e.g., backpropagation), and a simple nonlinear activation function (e.g., sigmoidal). A nonclassical framework, however, can be coherently developed. By NCC, I shall be referring to the class of models that have different combinations of pattern associator/autoassociative memory/competitive network topologies, with bidirectional connectivity and inhibitory competition, and that employ combined Hebbian and activation-phase learning algorithms (O'Reilly & Munakata 2000; Rolls & Treves 1998). Were NCC allowed to enter the competition, it would (or so I shall argue) obtain a "within-theory rank ordering" that could perhaps place it in a better position than the ACT-R theory. To demonstrate this, I will make three points with regard to 4 of the 12 functional constraints on the architecture of cognition that A&L take into consideration: knowledge integration, development, real-time performance, and flexible behavior.

On knowledge integration, classical connectionism (CC) gets a "worse" grade (see Table 1 of the target article). As an "intellectual combination" example of knowledge integration, A&L consider the literature on transfer of learning in infants. Marcus (2001) assessed the relationship between CC and rule-governed behavior by challenging the connectionist to account for experimental data that had been interpreted as showing that infants exploit (rule-governed) abstract knowledge in order to induce the implicit grammar common to different sequences of syllables (Marcus et al. 1999). Calvo and Colunga (in preparation) show how Marcus's infants-data challenge can be met with NCC (see Calvo & Colunga 2003, for a CC replica of this simulation). Our model (Fig. 1) is based on a simple recurrent network (SRN) architecture that has been supplemented with the following nonclassical features: (1) bidirectional (symmetric) propagation of activation, (2) inhibitory competition, (3) an error-driven form of learning (GenRec in McClelland 1994), and (4) the Hebbian model learning.

The fundamental component of our simulation resides in the fact that the network is pretrained with syllables that can be either duplicated or not. These first-order correlations in the environment amount to subregularities that can be exploited by the network in a semideterministic prediction task. During pretraining, the network learns to represent something general about duplication (i.e., sameness). This abstraction is crucial in encoding the patterns during the habituation phase. Like the infants in Marcus et al.'s study, the networks that were pretrained in a corpus in which some syllables were consistently duplicated learned to distinguish ABB patterns from ABA patterns after a brief period of training akin to infant's habituation.

Error-driven learning makes use of an activation-phase algorithm that, via bidirectional connectivity and symmetric weight matrices, permits the network to alter the knowledge acquired in the weights by computing the difference between an initial phase where the networks activations are interpreted as its "expectation" of what's to happen, and a later phase in which the environment provides the output response to be taken as the teaching signal. Activation-based signals in a prediction task are not to be interpreted in Marcus's terms. The ecologically grounded prediction task of the networks does not incorporate universally open-ended rules. Unsupervised Hebbian learning, on the other hand, makes its contribution by representing in hidden space the first-order correlational structure of the data pool. Our NCC architecture delivers a correct syntactic interpretation of the infants' data. The data are accounted for without the positing of rule-fitting patterns of behavior (allegedly required to constrain novel data).

On development, where CC is graded as "better," the score may be made even more robust. A&L remark upon CC's anti-nativist stance on the nature/nurture debate. Marín et al. (2003) argue, in the context of poverty-of-stimulus arguments in Creole genesis, that CC eschews any form of nativism. Creole genesis, nativists contend, can only be explained by appealing to a Chomskian Universal Grammar (UG). Substratists contend that Creole genesis is influenced, crucially, by substratum languages. We show how the process by which a Pidgin develops into a Creole can be modelled by an SRN exposed to a dynamic (substratum-based) environment. In this way, an empiricist approach is able to account for Creole grammar as a by-product of general-purpose learning mechanisms. Connectionist theory, we argue, furnishes us with a (statistical) alternative to nativism. Taking into account that combined Hebbian and activation-phase learning drives SRN networks to a better performance on generalization than the backpropagation algorithm does (O'Reilly & Munakata 2000), a NCC replica of this simulation would further strengthen connectionist's stronghold on the development criterion.

Biologically plausible NCC would cast light as well upon other Newell Test criteria: Real-time Performance, where classical con-

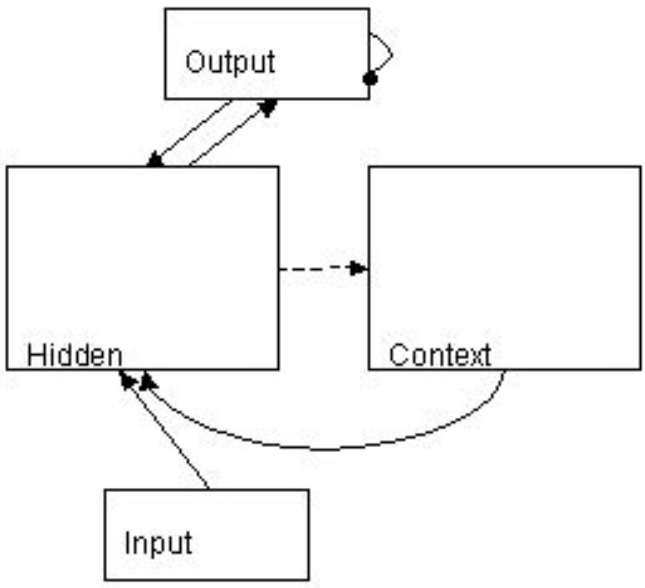


Figure 1 (Garzón). NCC network, with bidirectional connectivity and inhibitory competition, trained on a prediction task.

nectionism gets a "worse" grade, can be improved if we consider online dynamical coupling. In NCC models that do not depend on synchronous updates, it may be assumed, as A&L note, that "the time for a unit to reach full activation is a function of that activation" (sect. 5.2). Moreover, one-shot Hebbian learning (Rolls & Treves 1998), where a few event co-occurrences can contribute to fast recall, can also be seen as a motivation for not having to endorse a hybrid architecture. On the other hand, performance on the flexible behavior criterion would be enhanced as well. Notice that nonclassical, dynamical networks can compute any function to an arbitrary level of accuracy, while allowing for breakdowns in performance.

In general, I can see no good reason not to allow NCC to enter the decathlon. The best connectionist contestant should enter the competition, not a straw man (classical connectionism). It is usually argued that the endorsement of a symbolic-cum-subsymbolic stance would permit connectionism to remain at an appropriate level of realism (Palmer-Brown et al. 2002). However, "failure to acknowledge a symbolic level to thought" (target article, Abstract) cannot be interpreted as a weakness of connectionism when the score is revised as just described.

ACKNOWLEDGMENTS

The author was supported by a Ramón y Cajal research contract (Spanish Ministry of Science and Technology) during the preparation of this commentary. I thank Andy Clark, Eliana Colunga, and Javier Marín for helpful discussions.

Criteria and evaluation of cognitive theories

Petros A. M. Gelepithis

Cognitive Science Laboratory, Kingston University, Kingston upon Thames, KT1 2EE, England. **Petros@kingston.ac.uk**

Abstract: I have three types of interrelated comments. First, on the choice of the proposed criteria, I argue against any *list* and for a *system* of criteria. Second, on grading, I suggest modifications with respect to consciousness and development. Finally, on the choice of "theories" for evaluation, I argue for Edelman's theory of neuronal group selection instead of connectionism (classical or not).

Introduction. Anderson & Lebiere's (A&L's) target article is a useful contribution on the necessity and grading of criteria for a cognitive theory and their application of the Newell Test to classical connectionism and ACT-R a worthwhile exercise. The following comments are partly a criticism on their proposed list of criteria, partly a response to their invitation for modifications of their proposed grading, and partly a critique of their choice of theories for evaluation.

On the choice of criteria for a Theory of Mind (ToM).[1] A&L state that "[t]wice, Newell (1980; 1990) offered slightly different sets of 13 criteria on the human mind" and a bit further down that their table "gives the first 12 criteria from [Newell's] 1980 list, which were basically restated in the 1990 list" (target article, sect. 1: Introduction, 1st para.). Neither of these two statements is correct (as Table 1 confirms).

Furthermore, A&L's list is closer to Newell 1980 than to Newell 1990. No justification for this proximity is provided. Given that Newell's (1990) seminal book is incomparably more comprehensive than his 1980 paper, one wonders about the reasons for A&L's choice. Clearly, their claim of having *distilled* (emphasis added) Newell's two lists (cf. target article, Abstract) cannot be justified either. Although I agree that A&L's list is adequate to avoid "theoretical myopia" (Introduction, 2nd para.), it will create distortions in our quest for a ToM on account of being restricted to a fundamentally impoverished coverage of human phenomena (excluding, e.g., emotion, creativity, social cognition, and culture). It is worth noting that although Newell (1990, sect. 8.4) considered the extension of a unified theory of cognition (UTC) into the social band an important measure of its success, A&L chose to exclude from their list the one constraint with a social element that Newell had included (see item 9 in Table 2).

In contrast, evolution should not be a criterion! Humans are physical objects, but biology is fundamentally different from physics. Similarly, humans are biological systems, but psychology is fundamentally different from biology. The nature of human understanding (Gelepithis 1984; 1991; 1997) transcends the explanatory framework of modern Darwinism and, most importantly, of any future evolutionary theory. (For similar conclusions drawn upon different premises, see Mayr 1988; O'Hear 1997.)

Finally, a fourth list – very different from all previous three – has been offered by Gelepithis (1999). Of the four proposed lists, Table 2 juxtaposes the latest three. The reader can easily spot a number of obvious and significant differences among the three lists. For some of the less obvious, their corresponding serial numbers are in boldface. What all three have in common is that they do not provide necessary and sufficient conditions for a ToM. Still, the mind is a system (Bunge 1980; Hebb 1949; Sherrington 1906). We need, therefore, a *system* (not a list) of criteria characterising mind. A recent promising effort along this route is exemplified by Gelepithis (2002), which presents an *axiomatic system* delineating the class of intelligent systems as a foundation for the development of a ToM[2].

On some "objective measures." *Consciousness.* There are many volumes of readings (e.g., Hameroff et al. 1998; Revonsuo & Kampinnen 1994; Velmans 1996) at least as good as the one cited by A&L. Suggestions of measures on the basis of consciousness-related phenomena in one volume of readings should be avoided. Although universal agreement on what constitutes consciousness is nonexistent, Gelepithis (2001) has provided a list of

Table 1 (Gelepithis). *Extent of the overlap among the proposed sets of criteria by Newell and A&L*

Criteria	Comparisons with Respect to Newell's 1980 List		Comparison with Respect to Newell's 1990 List
	Newell 1990	A&L 2003	A&L 2003
New criteria	2	0	0
Significantly different criteria	3	2	5 or 6
Essentially equivalent criteria	3	3	3 or 2
Identical criteria	5	7	4

Table 2 (Gelepithis). *Three different lists of criteria on human mind.*

	Newell (1990)	Gelepithis (1999)	A&L (2003)
1	Behave flexibly as a function of the environment.		Flexible behaviour (~ Computational Universality).
2	Exhibit adaptive (rational, goal-oriented) behaviour.		Adaptive behaviour.
3	Operate in real time.		Operate in real time.
4	Operate in a rich, complex, detailed environment. Perceive an immense amount of changing detail. Use vast amounts of knowledge. Control a motor system of many degrees of freedom.	Be able to operate in environments of, at least, Earth-level complexity.	Vast knowledge base (sect. 2.4). Dynamic behaviour (sect. 2.5)
5	Use symbols and abstractions.		Knowledge integration.
6	Use language, both natural and artificial.	Acquisition and use of language to, at least, human-level complexity.	Use (natural) language.
7	Learn from the environment and from experience.		Learn from its environment.
8	Acquire capabilities through development.	Explain human neonate's capabilities for development.	Acquire capabilities through development.
9	Operate autonomously, but within a social community.	Operate autonomously, but within a social community.	
10	Be self-aware and have a sense of self.	Be conscious.	Exhibit self-awareness and a sense of self.
11	Be realisable as a neural system.		Be realisable within the brain.
12	Be constructable by an embryological growth process.		
13	Arise through evolution.		Arise through evolution.
14		Use of: (1) domain knowledge and (2) commonsense knowledge for problem solving.	
15		Able to communicate.	
16		Be able to develop skills (e.g., through earning) and judgment (e.g., through maturation).	
17		Develop *own* representational system	
18		Combine perceptual and motor information with *own* belief systems.	
19		Be creative.	
20		Be able to have and exhibit emotions.	

"topics that, *presently*, constitute the major issues in the study of consciousness." I propose that list as a measure.

Development. In view of the suggested grading for consciousness, one might be tempted to propose some or all of the phenomena covered in Johnson et al.'s (2002) reader as a measure for development. Instead, I propose as criterion what is generally agreed to be the fundamental objective in the study of development, namely, "unraveling the *interaction* between genetic specification and environmental influence" (Johnson et al. 2002, p. 3., emphasis added). This fundamental objective in the study of development is shared by most scientists in the field, and it is essentially identical with Piaget's (1967/1971) agenda for developmental psychology. Interestingly, Newell (1990, Ch. 8) has also chosen to talk about development in Piagetian terms.

Choice of "theories" for evaluation. Barring a straightforward case of a Rylean category mistake, A&L seem to believe that there is no difference between theories and a class of models. To put it less strongly, they support the school of thought that argues for theories as families of theoretical models. This is highly debatable in the philosophy of science literature (Giere 1998). Furthermore, taking *theory* in its good old-fashioned meaning, no connectionist (classical or not) model will qualify. In contrast, Edelman's (1989; 1992; Edelman & Tononi 2000) theory of neuronal group selection – based on different foundations[3] – would both have qualified and created a debate on the choice of criteria as well as the types of theories that certain criteria may or may not favour.

To conclude, A&L's concern that connectionists may question the reasonableness of their list is rather well based. Let us not forget that any theory (whether cognitive or otherwise) needs to be founded. Chapter 2 of Newell's (1990) *Unified Theories of Cognition* is an excellent starting point. Comparison between ACT-R's foundations (Anderson 1993; Anderson & Lebiere 1998) and those of SOAR would be revealing; further comparisons of a connectionist (classical or not) theoretical framework and of non-computational ToMs will greatly enhance the foundations of cognitive science and, I would argue, point to the need for a *system* – rather than a *list* – of criteria for Newell's Test.

NOTES

1. I use the terms cognitive theory, unified theories of cognition (UTCs), and ToM interchangeably with respect to their coextensive coverage of human phenomena, and UTC and ToM distinctly with respect to their characteristics.

2. For some interesting earlier results of our approach, the reader is referred to Gelepithis (1991; 1997), Gelepithis and Goodfellow (1992), Gelepithis and Parillon (2002).

3. Evolutionary and neurophysiological findings and principles and the synthetic neural modelling approach to the construction of intelligent entities. For a comparison of four ToMs, see Gelepithis (1999).

Meeting Newell's other challenge: Cognitive architectures as the basis for cognitive engineering

Wayne D. Gray, Michael J. Schoelles, and Christopher W. Myers

Cognitive Science Department, CogWorks Laboratory, Rensselaer Polytechnic Institute, Troy, NY 12180-3590.

{grayw; schoem; myersc}@rpi.edu http://www.rpi.edu/~grayw/ http://www.rpi.edu/~schoem/ http://www.rpi.edu/~myersc/

Abstract: We use the Newell Test as a basis for evaluating ACT-R as an effective architecture for cognitive engineering. Of the 12 functional criteria discussed by Anderson & Lebiere (A&L), we discuss the strengths and weaknesses of ACT-R on the six that we postulate are the most relevant to cognitive engineering.

To mix metaphors, Anderson & Lebiere (A&L) have donned Newell's mantle and picked up his gauntlet. The mantle is Newell's role as cheerleader for the cause of unified architectures of cognition (e.g., Newell 1990). The gauntlet is Newell's challenge to the modeling community to consider the broader issues that face cognitive science. Gauntlets come in pairs, so it is not surprising that Newell threw down another one (Newell & Card 1985), namely, hardening the practice of human factors to make it more like engineering and less based on soft science. (Although Newell and Card framed their arguments in terms of human-computer interaction, their arguments apply to human factors in general and cognitive engineering in particular.)

Cognitive engineering focuses on understanding and predicting how changes in the task environment influence task performance. We postulate that such changes are mediated by adaptations of the mix of cognitive, perceptual, and action operations to the demands of the task environment. These adaptations take place at the embodied cognition level of analysis (Ballard et al. 1997) that emerges at approximately ⅓ second. The evidence we have suggests that this level of analysis yields productive and predictive insights into design issues (e.g., Gray & Boehm-Davis 2000; Gray et al. 1993). However, whatever the eventual evaluation of this approach, our pursuit of it can be framed in terms of six of the Newell Test criteria.

Flexible behavior. We understand A&L to mean that the architecture should be capable of achieving computational universality by working around the limits of its bounded rationality. Hence, not every strategy is equally easy, and not every strategy works well in every task environment. ACT-R fits our cognitive engineering needs on this criterion because it provides a means of investigating, by modeling, how subtle changes in a task environment influence the interaction of perception, action, and cognition to form task strategies.

Real-time performance. When comparing models against human data, a common tack is to simulate the human's software environment to make it easier to run the model. Although such a simulation might represent the essential aspects of the human's task environment, the fidelity of the model's task environment is inevitably decreased. ACT-R enables us to run our models in the same software environment in which we run our subjects by providing time constraints at the time scale that perception, action, and cognition interact.

Adaptive behavior. Section 2.3 of the target article emphasizes Newell's complaint regarding the functionality of then extant theories of short-term memory. In our attempts to build integrated cognitive systems, we too have had similar complaints. For example, the work by Altmann and Gray (Altmann 2002; Altmann & Gray 2002) on task switching was motivated by a failed attempt to use existing theories (e.g., Rogers & Monsell 1995) to understand the role played by task switching in a fast-paced, dynamic environment. Hence, one role of a unified architecture of cognition is that it allows a test of the functionality of its component theories.

Section 5.3 emphasizes the ability to tune models to the "statistical structure of the environment." For cognitive engineering, adaptation includes changes in task performance in response to changes in the task environment, such as when a familiar interface is updated or when additional tasks with new interfaces are introduced. In our experience, ACT-R has some success on the first of these, namely, predicting performance on variations of the same interface (Schoelles 2002; Schoelles & Gray 2003). However, we believe that predicting performance in a multitask environment, perhaps by definition, will require building models of each task. Hence, it is not clear to us whether ACT-R or any other cognitive architecture can meet this critical need of cognitive engineering.

Dynamic behavior. The ability to model performance when the task environment, not the human operator, initiates change is vital for cognitive engineering. We can attest that ACT-R does well in modeling these situations (Ehret et al. 2000; Gray et al. 2000; 2002; Schoelles 2002).

Learning. For many cognitive engineering purposes, learning is less important than the ability to generate a trace of a task analysis of expert or novice performance. With all learning "turned off," ACT-R's emphasis on real-time performance and dynamic behavior makes it well suited for such purposes.

Learning is required to adapt to changes in an existing task environment or to show how a task analysis of novice behavior could, with practice, result in expert behavior. ACT-R's subsymbolic layer has long been capable of tuning a fixed set of production rules to a task environment. However, a viable mechanism for learning new rules had been lacking. With the new production compilation method of Taatgen (see Taatgen & Lee 2003) this situation may have changed.

Consciousness. A&L's discussion of consciousness includes much that cognitive engineering does not need, as well as some that it does. Our focus here is on one aspect: the distinction between implicit and explicit knowledge and the means by which implicit knowledge becomes explicit.

Siegler (Siegler & Lemaire 1997; Siegler & Stern 1998) has demonstrated that the implicit use of a strategy may precede conscious awareness and conscious, goal-directed application of that strategy. ACT-R cannot model such changes because it lacks a mechanism for generating top-down, goal-directed cognition from bottom-up, least-effort-driven adaptations.

Conclusions: Meeting Newell's other challenge. Unified architectures of cognition have an important role to play in meeting Newell's other challenge, namely, creating a rigorous and scientifically based discipline of cognitive engineering. Of the six criteria discussed here, ACT-R scores one best, four better, and one worse, whereas classical connectionism scores two better, two mixed, and two worse. We take this as evidence supporting our choice of ACT-R rather than connectionism as an architecture for cognitive engineering. But, in the same sense that A&L judge that ACT-R has a ways to go to pass the Newell Test, we judge that ACT-R has a ways to go to meet the needs of cognitive engineering. As the Newell Test criteria become better defined, we hope that they encourage ACT-R and other architectures to develop in ways that support cognitive engineering.

ACKNOWLEDGMENTS

Preparation of this commentary was supported by grants from the Office of Naval Research (ONR# N000140310046) as well as by the Air Force Office of Scientific Research (AFOSR# F49620–03–1–0143).

Bring ART into the ACT

Stephen Grossberg

Department of Cognitive and Neural Systems, Boston University, Boston, MA 02215. **steve@bu.edu** **http://www.cns.bu.edu/Profiles/Grossberg**

Abstract: ACT is compared with a particular type of connectionist model that cannot handle symbols and use nonbiological operations which do not learn in real time. This focus continues an unfortunate trend of straw man debates in cognitive science. Adaptive Resonance Theory, or ART, neural models of cognition can handle both symbols and subsymbolic representations, and meets the Newell criteria at least as well as these models.

The authors' use of the nomenclature, "classical connectionist models," falsely suggests that such models satisfy the Newell criteria better than other neural models of cognition. The authors then dichotomize ACT with "classical" connectionism based on its "failure to acknowledge a symbolic level to thought. In contrast, ACT-R includes both symbolic and subsymbolic components" (target article, Abstract). Actually, neural models of cognition such as ART include both types of representation and clarify how they are learned. Moreover, ART was introduced before the "classical" models (Grossberg 1976; 1978a; 1980) and naturally satisfies key Newell criteria. In fact, Figures 2 and 3 of ACT are reminiscent of ART circuits (e.g., Carpenter & Grossberg 1991; Grossberg 1999b). But ART goes further by proposing how laminar neocortical circuits integrate bottom-up, horizontal, and top-down interactions for intelligent computation (Grossberg 1999a; Raizada & Grossberg 2003).

Critiques of classical connectionist models, here called CM (Carnegie Mellon) connectionism, show that many such models cannot exist in the brain (e.g., Grossberg 1988; Grossberg et al. 1997; Grossberg & Merrill 1996). We claim that ART satisfies many Newell criteria better, with the obvious caveat that no model is as yet a complete neural theory of cognition.

Flexible behavior. ART models are self-organizing neural production systems capable of fast, stable, real-time learning about arbitrarily large, unexpectedly changing environments (Carpenter & Grossberg 1991). These properties suit ART for large-scale technological applications, ranging from control of mobile robots, face recognition, remote sensing, medical diagnosis, and electrocardiogram analysis to tool failure monitoring, chemical analysis, circuit design, protein/DNA analysis, musical analysis, and seismic, sonar, and radar recognition, in both software and VLSI microchips (e.g., Carpenter & Milenova 2000; Carpenter et al. 1999; Granger et al. 2001). The criticism of CM connectionism "that complex, sequentially organized, hierarchical behavior" cannot be modeled also does not apply to ART (e.g., Bradski et al. 1994; Cohen & Grossberg 1986; Grossberg 1978a; Grossberg & Kuperstein 1989; Grossberg & Myers 2000; also see the section on dynamic behavior later in this commentary).

Real-time performance. ART models are manifestly real-time in design, unlike CM connectionist models.

Adaptive behavior. ART provides a rigorous solution of the *stability-plasticity dilemma,* which was my term for *catastrophic forgetting* before that phrase was coined. "Limitations like short-term memory" (target article, sect. 5.3) can be derived from the LTM Invariance Principle, which proposes how working memories are designed to enable their stored event sequences to be stably chunked and remembered (Bradski et al. 1994; Grossberg 1978a; 1978b).

Vast knowledge base. ART can directly access the globally best-matching information in its memory, no matter how much it has learned. It includes additional criteria of value and temporal relevance through its embedding in START models that include cognitive-emotional and adaptive timing circuits in addition to cognitive ART circuits (Grossberg & Merrill 1992; 1996).

Dynamic behavior. "Dealing with dynamic behavior requires a theory of perception and action as well as a theory of cognition" (sect. 2.5). LAMINART models propose how ART principles are incorporated into perceptual neocortical circuits and how high-level cognitive constraints can modulate lower perceptual representations through top-down matching and attention (Grossberg 1999a; Raizada & Grossberg 2003). ART deals with novelty through *complementary* interactions between attentional and orienting systems (Grossberg 1999b; 2000b), the former including corticocortical, and the latter, hippocampal, circuits. Action circuits also obey laws that are *complementary* to those used in perception and cognition (Grossberg 2000b), notably VAM (Vector Associative Map) laws. VAM-based models have simulated identified brain cells and circuits and the actions that they control (e.g., Brown et al. 1999; Bullock et al. 1998; Contreras-Vidal et al. 1997; Fiala et al. 1996; Gancarz & Grossberg 1999; Grossberg et al. 1997), including models of motor skill learning and performance (Bullock et al. 1993a; 1993b; Grossberg & Paine 2000).

Knowledge integration. ART reconciles distributed and symbolic representations using its concept of resonance. Individual features are meaningless, just as pixels in a picture are meaningless. A learned category, or symbol, is sensitive to the global patterning of features but cannot represent the *contents* of the experience, including their conscious qualia, because of the very fact that a category is a compressed, or symbolic, representation. Resonance between these two types of information converts the *pattern* of attended features into a coherent context-sensitive state that is linked to its symbol through feedback. This coherent state, which binds distributed features and symbolic categories, can enter consciousness. ART predicts that *all conscious states are resonant states.* In particular, resonance binds spatially distributed features into a synchronous equilibrium or oscillation. Such synchronous states attracted interest after being reported in neurophysiological experiments. They were predicted in the 1970s when ART was introduced (see Grossberg 1999b). Recent neurophysiological experiments have supported other ART predictions (Engel et al. 2001; Pollen 1999; Raizada & Grossberg 2003). Fuzzy ART learns explicitly decodable Fuzzy IF-THEN rules (Carpenter et al. 1992). Thus ART accommodates symbols and rules, as well as subsymbolic distributed computations.

Natural language. ART has not yet modeled language. Rather, it is filling a gap that ACT-R has left open: "ACT-R lacks any theory of the processes of speech perception or speech production" (sect. 4.5, para. 3). ART is clarifying the *perceptual units* of speech perception, word recognition, working memory, and sequential planning chunks on which the brain builds language (e.g., Boardman et al. 1999; Bradski et al. 1994; Grossberg 1978a; 1978b; 1999b; Grossberg et al. 1997a; 1997b; Grossberg & Myers 2000; Grossberg & Stone 1986a; 1986b). Such studies suggest that a radical rethinking of psychological space and time is needed to understand language and to accommodate such radical claims as, "Conscious speech is a resonant wave." ACT-R also does not have "mechanisms . . . [of] perceptual recognition, mental imagery, emotion, and motivation" (sect. 4.5). These are all areas where ART has detailed models (e.g., Grossberg 2000a; 2000c). Speech production uses complementary VAM-like mechanisms (Callan et al. 2000; Guenther 1995). After perceptual units in vision became sufficiently clear, rapid progress ensued at all levels of vision (http://www.cns.bu.edu/Profiles/Grossberg). This should also happen for language.

Development. ART has claimed since 1976 that processes of cortical development in the infant are on a continuum with processes of learning in the adult, a prediction increasing supported recently (e.g., Kandel & O'Dell 1992).

Evolution. "Cognitive plasticity . . . What enables this plasticity in the architecture?" (sect. 5.11). ART clarifies how the ability to

learn quickly and stably throughout life implies cognitive properties like intention, attention, hypothesis testing, and resonance. Although Bayesian properties emerge from ART circuits, ART deals with novel experiences where no priors are defined.

Brain. CM connectionism is said to be "best," although its main algorithms are biologically unrealizable. ART and VAM are realized in verified brain circuits.

It might be prudent to include more ART in ACT. I also recommend eliminating straw man "debates" that do not reflect the true state of knowledge in cognitive science.

ACKNOWLEDGMENTS

Preparation of the this commentary was supported in part by the Air Force Office of Scientific Research (AFOSR F49620–01–1–0397) and the Office of Naval Research (ONR N00014–01–1–0624).

Developing a domain-general framework for cognition: What is the best approach?

James L. McClelland[a], David C. Plaut[a], Stephen J. Gotts[b], and Tiago V. Maia[c]

[a]*Department of Psychology and Center for the Neural Basis of Cognition, Carnegie Mellon University, Pittsburgh, PA 15213;* [b]*Department of Psychology and Center for the Neural Basis of Cognition, Carnegie Mellon University, and Laboratory of Neuropsychology, NIMH/NIH, Bethesda, MD 20892;* [c]*Department of Psychology, Carnegie Mellon University, Pittsburgh, PA 15213.* **jlm@cnbc.cmu.edu http://www.cnbc.cmu.edu/~jlm plaut@cmu.edu http://www.cnbc.cmu.edu/~plaut gotts@nih.gov http://www.cnbc.cmu.edu/~gotts tmaia@cmu.edu http://www.cnbc.cmu.edu/~tmaia**

Abstract: We share with Anderson & Lebiere (A&L) (and with Newell before them) the goal of developing a domain-general framework for modeling cognition, and we take seriously the issue of evaluation criteria. We advocate a more focused approach than the one reflected in Newell's criteria, based on analysis of failures as well as successes of models brought into close contact with experimental data. A&L attribute the shortcomings of our parallel-distributed processing framework to a failure to acknowledge a symbolic level of thought. Our framework does acknowledge a symbolic level, contrary to their claim. What we deny is that the symbolic level is the level at which the principles of cognitive processing should be formulated. Models cast at a symbolic level are sometimes useful as high-level approximations of the underlying mechanisms of thought. The adequacy of this approximation will continue to increase as symbolic modelers continue to incorporate principles of parallel distributed processing.

In their target article, Anderson & Lebiere (A&L) present a set of criteria for evaluating models of cognition, and rate both their own ACT-R framework and what they call "classical connectionism" on the criteria. The Parallel Distributed Processing (PDP) approach, first articulated in the two PDP volumes (Rumelhart et al. 1986) appears to be close to the prototype of what they take to be "classical connectionism." While we cannot claim to speak for others, we hope that our position will be at least largely consistent with that of many others who have adopted connectionist/PDP models in their research.

There are three main points that we would like to make.

1. We share with A&L (and with Newell before them) the effort to develop an overall framework for modeling human cognition, based on a set of domain-general principles of broad applicability across a wide range of specific content areas.

2. We take a slightly different approach from the one that Newell advocated, to pursuing the development of our framework. We think it worthwhile to articulate this approach briefly and to comment on how it contrasts with the approach advocated by Newell and apparently endorsed by A&L.

3. We disagree with A&L's statement that classical connectionism denies a symbolic level of thought. What we deny is only the idea that the symbolic level is the level at which the principles of processing and learning should be formulated. We treat symbolic cognition as an emergent phenomenon that can sometimes be approximated by symbolic models, especially those that incorporate the principles of connectionist models.

In what follows, we elaborate these three points, addressing the first one only briefly since this is a point of agreement between A&L and us.

The search for domain-general principles. There is a long-standing tradition within psychological research to search for general principles that can be used to address all aspects of behavior and cognition. With the emergence of computational approaches in the 1950s and 1960s, and with the triumph of the von Neumann architecture as the basis for artificial computing devices, this search could be formulated as an effort to propose what Newell called "a unified architecture for cognition." An architecture consists of a specification of (1) the nature of the building blocks out of which representations and processes are constructed, (2) the fundamental rules by which the processes operate, and (3) an overall organizational plan that allows the system as a whole to operate. Newell's SOAR architecture and A&L's ACT-R architecture are both good examples of architectures of this type. For our part, we have sought primarily to understand (1) the building blocks and (2) the fundamental rules of processing. Less effort has been devoted to the specifics of the overall organizational plan as such, although we do take a position on some of the principles that the organizational plan instantiates. Because the organization is not fully specified as such, we find it more congenial to describe what we are developing as a framework rather than an architecture. But this is a minor matter; the important point is the shared search for general principles of cognition.

We are of course well aware that this search for general principles runs counter to a strong alternative thread that treats distinct domains of cognition as distinct cognitive modules that operate according to domain-specific principles. Such a view has been articulated for language by Chomsky; for vision, by Marr. Fodor and Keil have argued the more general case, and a great deal of work has been done to try to elucidate the specific principles relevant to a wide range of alternative domains. Although we cannot prove that this approach is misguided, we have the perspective that the underlying machinery and the principles by which it operates are fundamentally the same across all different domains of cognition. While this machinery can be tuned and parameterized for domain-specific uses, understanding the broad principles by which it operates will necessarily be of very broad relevance.

How the search for domain-general principles is carried out. If one's goal is to discover the set of domain-general principles that govern all aspects of human cognition, how best is the search for such principles carried out? Our approach begins with the fundamental assumption that it is not possible to know in advance what the right set of principles are. Instead, something like the following discovery procedure is required.

1. Begin by formulating a putative set of principles.
2. Develop models based on these principles and apply them to particular target domains (i.e., bodies of related empirical phenomena).
3. Assess the adequacy of the models so developed and attempt to understand what really underlies both successes and failures of the models.
4. Use the analysis to refine and elaborate the set of principles, and return to step 2.

In practice this appears to be the approach both of Newell and of A&L. Newell and his associates developed a succession of cognitive architectures, as has Anderson; indeed, Newell suggested that his was only really one attempt, and that others should put forward their own efforts. However, Newell argued for broad application of the framework across all domains of cognition, suggesting that an approximate account within each would be satisfactory. In contrast, we advocate a more focused exploration of a few informative target domains, using failures of proposed models to guide further explorations of how the putative set of principles should be elaborated. To illustrate the power of this approach,

we briefly review two cases. Note that we do not mean to suggest that A&L explicitly advocate the development of approximate accounts. Rather, our point is to bring out the importance of focus in bringing out important principles of cognition.

1. The interactive activation model (McClelland & Rumelhart 1981) explored the idea that context effects in perception of letters – specifically, the advantage for letters in words relative to single letters in isolation – could be attributed to the bidirectional propagation of excitatory and inhibitory signals among simple processing units whose activation corresponds to the combined support for the item the unit represents. When a letter occurs in a word, it and the other letters will jointly activate the unit for the word, and that unit will in turn send additional activation back to each of the letters, thereby increasing the probability of recognition. Similar ideas were later used in the TRACE model of speech perception (McClelland & Elman 1986) to account for lexical influences on phoneme identification. Massaro (1989; Massaro & Cohen 1991) pointed out that the interactive activation model failed to account for the particular quantitative form of the influence of context on the identification of a target item. He argued that the source of the problem lay specifically in the use of bidirectional or interactive activation between phoneme or letter units on the one hand and word units on the other. Since the interactive activation model fit the data pretty well, Newell might have advocated accepting the approximation, and moving on to other issues. However, close investigation of the issue turned out to lead to an important discovery. Subsequent analysis (McClelland 1991; Movellan & McClelland 2001) showed that the failure of the interactive activation model arose from faulty assumptions about the source of variability in performance.

Discovering this was made possible by the failure of the model. It then became possible to consider what changes have to be made in order to fit the data. McClelland (1991) showed that the model had a general deficiency in capturing the joint effects of two different sources of influence even if they were both bottom up and activation was only allowed to propagate in a feedforward direction. The problem was attributed instead to the fact that in the original McClelland and Rumelhart model, the interactive activation process was completely deterministic, and activations were transformed into response probabilities only at the moment of response selection. This led to the discovery of what we take to be an important principle: that the activation process is not only graded and interactive but also intrinsically variable. Reformulated versions of the model incorporating intrinsic variability, in addition to graded representation and interactive processing, were shown through simulations (McClelland 1991) and mathematical analysis (Movellan & McClelland 2001) to produce the right quantitative form of contextual influence on phoneme and letter identification. This principle of intrinsic variability has been incorporated in several subsequent models, including a model that addresses in detail the shapes of reaction time distributions and the effects of a variety of factors on these distributions (Usher & McClelland 2001).

2. Seidenberg and McClelland (1989) introduced a model that accounted for frequency, regularity, and consistency effects in single word reading. The model relied on a single network that mapped distributed input representations of the spellings of words, via one layer of hidden units, onto a set of output units representing the phonemes in the word's pronunciation. However, as two independent critiques pointed out (Besner et al. 1990; Coltheart et al. 1993), the model performed far worse than normal human subjects at reading pronounceable nonwords. Both critiques attributed this shortcoming of the model to the fact that it did not rely on separate lexical and rule-based mechanisms. However, subsequent connectionist research (Plaut et al. 1995; 1996) demonstrated that the particular choice of input and output representations used by Seidenberg and McClelland (1989) was instead the source of the difficulty. These representations tended to disperse the regularity in the mapping from spelling to sound over a number of different processing units. This was because the input units activated by a given letter depended on the surrounding context, and the output units representing a given phoneme were likewise context dependent. Because the learning in the model is in the connections among the units, this led to a dispersion of the information about the regularities across many different connections and created a situation in which letters in nonwords might occur in contexts that had not previously been encountered by the network. This led to the discovery of the principle that to succeed in capturing human levels of generalization performance, the representations used in connectionist networks must condense the regularities. Subsequent models of word reading, inflectional morphology, and other cognitive tasks have used representations that condense the regularities, leading them to achieve human levels of performance with novel items while yet being able to learn to process both regular and exception words.[1]

These two case studies bring out the importance of taking seriously mismatches between a model's behavior and human performance data, even when the model provides an approximate account of most of the relevant phenomena. We believe that such mismatches are important forces in driving the further development of a framework. Of course, such mismatches might also reflect a fundamental inadequacy of the framework as a whole or of its most fundamental grounding assumptions. Analysis is required to determine which; but whatever the outcome, the examination of failures of fit is an important source of constraint on the further development of the framework.

With these comments in mind, we can now turn to the framing of the goals of cognitive modeling as articulated in the sorts of criteria that Newell proposed and A&L have adopted with their own modifications. We agree that it is useful to focus attention on some of these general issues, and that there is more to a good cognitive model than simply a close fit to experimental data. We would note, however, that making the effort at this stage to achieve the sort of breadth that Newell's criteria imply may distract attention from addressing critical discrepancies that can only be revealed through close comparison of models and data. We have chosen to adopt a more focused approach, but we do not deny that a broader approach may reveal other limitations, and that it may be worthwhile for some researchers to follow Newell's strategy.

The importance and nature of the symbolic level. A&L suggest that the shortcomings of the connectionist approach are fundamental, deriving from its failure to acknowledge a symbolic level of thought, whereas the shortcomings of the ACT-R theory are temporary, and derive from its failure as yet to address certain of Newell's criteria. We have a very different reading of the situation.

First of all, our PDP approach does not deny a symbolic level of thought. What we deny is only that the symbolic level is the appropriate level at which the principles of processing and learning should be formulated. We treat symbolic thought as an emergent phenomenon which can sometimes be approximated to a degree by a model formulated at the symbolic level, but which, on close scrutiny, does not conform exactly to the properties that it should have according to symbolic models.

As is well known, the issue here is one that has been extensively explored in the context of research on the formation of past tenses and other inflections of nouns and verbs. A recent exchange of articles contrasts the PDP perspective (McClelland & Patterson 2002a; 2002b) and Pinker's symbolic, dual-mechanism account (Pinker & Ullman, 2002a; 2002b). Here we will present the PDP perspective.

In several places, Pinker and his colleagues have argued that the past tense of English is characterized by two mechanisms, one involving symbolic rules, and the other involving a lexical mechanism that operates according to connectionist principles. A symbolic rule, according to Pinker's approach, is one that applies uniformly to all items that satisfy its conditions. Furthermore, such conditions are abstract and very general. For example, the past-tense rule applies uniformly to any string of phonemes, provided only that it is the stem of a verb. In many places Pinker also states that symbolic rules are acquired suddenly; this conforms to

the idea that a rule is something that one either has or does not have. Finally, the symbolic rule is thought to require a completely different kind of mechanism than the one underlying the inflection of exceptions, leading to the prediction that brain lesions could selectively impair the ability to use the rule while leaving the inflection of irregular forms intact.

Although Pinker and his colleagues have pointed to evidence they believe supports their characterization of the mechanism that produces regular past-tense inflections, in their review of that evidence McClelland and Patterson (2002a) found instead that in every case the evidence supports an alternative characterization, first proposed by Rumelhart and McClelland (1986a), in which the formation of an inflected form arises from the interactions of simple processing units via weighted connections learned gradually from exposure to example forms in the language.[2] First, the evidence indicates that the onset of use of regular forms is gradual (extending over a full year; see Brown 1973; Hoeffner 1996). It is initially restricted to verbs characterized by a set of shared semantic properties, and then gradually spreads to other verbs starting with those sharing some of the semantic properties of the members of the initial set (Shirai & Anderson 1995). Second, usage of the regular past tense by adults is not insensitive to phonology but instead reflects phonological and semantic similarity to known regular verbs (Albright & Hayes 2001; Ramscar 2002). Third, purported dissociations arising from genetic defects (Gopnik & Crago 1991) or strokes (Ullman et al. 1997) disappear when materials are used that control for frequency and phonological complexity (Bird et al. 2003; Vargha-Khadem et al. 1995); individuals with deficits in inflection of regular forms show corresponding deficits with appropriately matched exceptions. In short, the acquisition and adult use of the regular past tense exhibits exactly those characteristics expected from the connectionist formulation. Ultimate adult performance on regular items conforms approximately to the predictions of the rule; for example, reaction time and accuracy inflecting regular forms is relatively insensitive to the word's own frequency. But exactly the same effect also arises in the connectionist models; as they learn from many examples that embody the regular pattern, the connection weights come to reflect it in a way that supports generalization to novel items and makes the number of exposures to the item itself relatively unimportant.

In summary, the characteristics expected on a connectionist approach, but not the symbolic rule approach of Pinker, are exhibited by human performance in forming inflections. Such characteristics include fairly close approximation to what would be expected from use of a symbolic rule under specifiable conditions, but allow for larger discrepancies from what would be predicted from the rule under other conditions (i.e., early in development, after brain damage of particular kinds, and when the language environment is less systematic).[3]

What implications do the characteristics of human performance in forming inflections have for the ACT-R approach of A&L? They have already described an ACT-R model (Taatgen & Anderson 2002) of past-tense formation in which the acquisition of the regular past tense occurs fairly gradually, and we have no doubt that with adjustment of parameters even more gradual acquisition would occur. Furthermore, we see relatively little in A&L's formulation that ties them to the assumption that the conditions for application of symbolic rules must be abstract as Pinker (1991; Pinker & Ullman 2002a) and Marcus (2001) have claimed. Nor is there anything that requires them to posit dissociations, since production rules are used in their model for both regular and exceptional forms. Thus, although the past tense rule actually acquired in the Taatgen and Anderson model is as abstract and general as the one proposed by Pinker, a modified version of their model could surely be constructed, bringing it closer to the connectionist account. To capture the graded and stochastic aspects of human performance, they have introduced graded strengths that are tacked onto symbolic constructs (propositions and productions), thereby allowing them to capture graded familiarity and regularity effects. To capture similarity effects, there is no reason why the condition-matching operation performed by rule-like productions could not be formulated as graded constraints, so that the degree of activation of a production would depend on the degree to which its conditions match current inputs. Indeed, A&L note that by allowing graded condition matching in ACT-R, they can capture the graded, similarity-based aspects of human performance that are naturally captured within the connectionist framework.

Even these adjustments, however, would leave one aspect of connectionist models unimplemented in the Taatgen and Anderson model. This is the ability of connectionist models to exploit multiple influences simultaneously, rather than to depend on the output generated by just one production at a time. Specifically, in the Taatgen and Anderson account of past-tense formation, a past-tense form is generated either by the application of the general *-ed* rule or by the application of an item-specific production; the form that is generated depends on only one of these productions, not on their simultaneous activation. We argue that this is a serious weakness, in that it prevents the Taatgen and Anderson model from exploiting the high degree of conformity with the regular pattern that exists among the exceptions. In our view this is an important and general limitation of many symbolic models, even ones like ACT-R that have moved a long way toward incorporating many of the principles of processing espoused by connectionists.

As McClelland and Patterson (2002b) have noted, fully 59% of the exceptional past-tense verbs in English end in /d/ or /t/. In the connectionist models, the same connection-based knowledge that imposes the regular inflection on fully regular verbs also operates in the inflection of these exceptional cases. That is, the same connections that add /t/ to the regular verb *like* to make *liked* also add /t/ to the irregular verb *keep* to make *kept*. In the case of *kept,* additional influences (from experience with *kept* itself and other similar cases) also operate to allow the model to capture the alteration of the vowel that makes this item an exception. In contrast, in the Taatgen and Anderson model and many other dual-mechanism models, only one production at a time can fire, so that a past-tense form is either generated by the rule (in which case it will be treated as regular) or by a production specific to it as an exception. Given this, no benefit accrues to an exception for sharing properties of the regular past tense, and all exceptions might as well be completely arbitrary. This is problematic because it leaves unexplained important aspects of the distributions of word forms. Across languages, there are many forms that are partially regular and very few that are completely arbitrary, and those that are completely arbitrary are of very high frequency (Plunkett & Marchman 1991); the same is true for irregular spelling-to-sound correspondences. This suggests that human language users are highly sensitive to the degree to which exceptions share properties with regular items, contrary to the properties of the Taatgen and Anderson model.

In response to this, we anticipate that A&L might be tempted to modify the ACT-R framework even further in the direction of connectionist models by allowing application of multiple productions to work together to produce an individual inflected word form. We certainly think this would lead to models that would be more likely than current ACT-R–based accounts to address the influence of regularities in exceptions, and would bring ACT-R more fully into line with the fundamental idea of parallel distributed processing. After all, the essence of PDP is the idea that every act of cognition depends on and is distributed over a large number of contributing units, quite different from what happens presently in ACT-R, where any given output is the product of the application of a single production.

While such a change to ACT-R would, we believe, improve it considerably, we want to simply note two points in this context. First, this would continue the evolution of symbolic models of human cognition even further in a connectionist-like direction. This evolution, which has been in process for some time, is not, in our view, accidental, because with each step in this direction, symbolic

models have achieved a higher degree of fidelity to the actual properties of human cognition. What this indicates to us is that, although the shortcomings of symbolic models may be temporary (as A&L suppose), they are most likely to be overcome by incorporation of the very principles that govern processing as defined at the connectionist level.

Second, as symbolic modelers take each new step in the direction of connectionist models, they do so accepting the fact that the phenomena to be explained have the characteristics that served to motivate the exploration of connectionist models in the first place. This, in turn, undermines the stance that the fundamental principles of human cognition should be formulated at the symbolic level, and instead further motivates the exploration of principles at the connectionist level. While we acknowledge that connectionist models still have many limitations, we nevertheless feel that this does not arise from any failure to acknowledge a symbolic level of thought. Instead we suggest that it arises from the fact the connectionists (like symbolic modelers) have not yet had the chance to address all aspects of cognition or all factors that may affect it.

In spite of our feeling that the facts of human cognition are completely consistent with the principles of parallel distributed processing, we do not wish to give the impression that we see no merit in modeling that is directed at the symbolic level. Given that symbolic formulations often do provide fairly good approximations, it may be useful to employ them in cases where it would be helpful to exploit their greater degree of abstraction and succinctness. We believe that work at a symbolic level will proceed most effectively if it is understood to be approximating an underlying system that is much more parallel and distributed, because at that point insights from work at the connectionist level will flow even more freely into efforts to capture aspects of cognition at the symbolic level.

ACKNOWLEDGMENTS

Preparation of this commentary was supported by Interdisciplinary Behavioral Science Center Grant MH 64445 from the National Institute of Mental Health (USA). Tiago V. Maia was supported by the Foundation for Science and Technology (Portugal). We thank the other members of the PDP Research Group at Carnegie Mellon for useful discussions.

NOTES

1. It is necessary to note that none of the models we have discussed fully embody all the principles of the PDP framework. For example, the interactive activation and TRACE models use localist, not distributed, representations, while the models of spelling-to-sound mapping (Seidenberg & McClelland 1989; Plaut et al. 1996) do not incorporate intrinsic variability. This fact can lead to confusion about whether there is indeed a theoretical commitment to a common set of principles.

In fact, we do have such a commitment. The fact that individual models do not conform to all of the principles is a matter of simplification. This leads to computational tractability and can foster understanding, and we adopt the practices only for these reasons. Everyone should be aware that models that are simplified embodiments of the theory do not demonstrate that models incorporating all of its complexity will be successful. In such cases further research is necessary, especially when the possibility of success is controversial. For example, Joanisse and Seidenberg (1999) used localist word units in their model of past-tense inflection, and Pinker and Ullman (2002a; 2002b) have argued that this is essential. In this context, we fully accept that further work is necessary to demonstrate that a model using distributed semantic representations can actually account for the data.

2. It should be noted here that none of these models assume that learning occurs through correction of overtly generated errors. Instead, it is assumed that exposure provides examples of appropriate usage in context. The learner uses the context as input to generate an internal representation corresponding to the expected phonological form. Learning is driven by the discrepancy between this internal representation and the actual perceived form provided by the example.

3. Marcus et al. (1995) claimed that German has a regular plural (the so-called s plural) that conforms to the expectation of the symbolic approach, in spite of the fact that it is relatively infrequent. However, subsequent investigations indicate that the s plural does not exhibit the properties one would expect if it were based on a symbolic rule (Bybee 1995; Hahn & Nakisa 2000).

Evaluating connectionism: A developmental perspective

Claire F. O'Loughlin[a] and Annette Karmiloff-Smith[b]

[a]*Department of Psychology, University of Canterbury, Christchurch, New Zealand;* [b]*Neurocognitive Development Unit, Institute of Child Health, London WC1N 1EH, UK.* **aallardy@xtra.co.nz**
a.karmiloff-smith@ich.ucl.ac.uk

Abstract: This commentary questions the applicability of the Newell Test for evaluating the utility of connectionism. Rather than being a specific theory of cognition (because connectionism can be used to model nativist, behaviorist, or constructivist theories), connectionism, we argue, offers researchers a collection of computational and conceptual tools that are particularly useful for investigating and rendering specific fundamental issues of human development. These benefits of connectionism are not well captured by evaluating it against Newell's criteria for a unified theory of cognition.

In this commentary, we question Anderson & Lebiere's (A&L's) project of grading connectionism according to the Newell Test as an appropriate means of assessing its utility for cognitive science. In our view, connectionism, unlike ACT-R, is not a specific theory of cognition. It can be used to model nativist, behaviourist, or constructivist theories by modifying parameters with respect to built-in representational and architectural or computational structures. Rather, connectionism is a set of computational and conceptual tools that offer researchers new and precise ways of thinking about and investigating complex emergent behaviour. From this standpoint, if we take the view that theory evaluation in science is best conceived as a comparative affair in which mature theories are evaluated along a number of dimensions to determine which provides the best explanation of the phenomena in question (e.g., Lakatos 1970; Thagard 1992), then connectionism does not offer an appropriate theoretical alternative against which to evaluate ACT-R. Moreover, the current appraisal of connectionism against Newell's criteria actually misses many of the positive applications of connectionist tools in cognitive science research. In developmental psychology, for example, this methodological and conceptual toolbox has been put to use in the service of tackling long-standing issues about the mechanisms responsible for developmental change and, more generally, has supported renewed efforts to construct a genuinely interactional account as a theoretical framework for cognitive development (Elman et al. 1996; Karmiloff-Smith 1992; Newcombe 1998). It has also been successfully used to clarify the fundamental differences between adult neuropsychological patients and children with developmental disorders (Karmiloff-Smith 1997; 1998; Karmiloff-Smith et al. 2002; 2003; Thomas & Karmiloff-Smith 2002) and to model how language acquisition can follow atypical developmental trajectories (Thomas & Karmiloff-Smith 2003).

Connectionist models have been shown to be highly relevant to the concerns of developmental researchers, first, because they offer a valuable means of investigating the necessary *conditions* for development. That is, connectionist models provide concrete demonstrations of how the application of simple, low-level learning algorithms operating on local information can, over developmental time, give rise to high-level emergent cognitive outcomes (Elman et al. 1996; Karmiloff-Smith 1992; Karmiloff-Smith et al. 1998; Plunkett et al. 1997). These demonstrations in turn have forced researchers to revisit assumptions about what can actually be learned as opposed to what has to be prespecified, and to recognize that far more structure is latent in the environmental input and capable of being abstracted by basic learning algorithms than previously imagined.

Concerning assumptions about the nature of the starting state in the developing individual, explorations with connectionist models have been pivotal in clarifying the issue of innateness and identifying a range of potential ways in which innate constraints can be realised (Karmiloff-Smith et al. 1998). As Elman et al. (1996) make clear, despite the current dominance of nativist approaches

to the development of language and cognition, scant attention has been given to the issue of biological plausibility in discussions of innate properties, and there has been little investigation of the potential variety of ways in which something could be innate. In contrast, and as a direct result of their experience with connectionist models, Elman et al. (1996) not only present a case against the plausibility of "representational nativism," but also offer a framework for developing alternative conceptions of innate constraints on development that draws on architectural and timing constraints in connectionist models as a guide.

In addition to clarifying the necessary conditions for development, connectionist models also provide a vehicle for exploring the *dynamics* of development. One of the key insights provided by connectionist models is that the mapping between overt behaviour and underlying mechanism is often nonlinear. As Elman et al. (1996) emphasize, contrary to assumptions underpinning much developmental research, qualitative changes in behaviour do not necessarily signal qualitative changes in the mechanisms responsible for that behaviour. Instead, these models demonstrate that sudden dramatic effects in terms of the output of a system can be produced by tiny, incremental changes in internal processing over time. In the case of ontogenetic development, this suggests that apparent discontinuities in conceptual or linguistic understanding or output may not be the result of new mechanisms coming online at certain points in development as has often been assumed, but instead reflect the continuous operation of the same mechanism over time.

Added to demonstrations of how the same mechanism can be responsible for multiple behaviours, connectionist models can also illuminate the reverse case in which a single outcome or behaviour arises through the action of multiple interacting mechanisms. Further, Elman et al. (1996) point to instances where the same behavioural outcome can be produced in a number of different ways, as in the case of degraded performance in artificial neural networks. (See Karmiloff-Smith 1998 for how crucial this is in understanding so-called behaviour in the normal range in some developmental disorders). Precisely because connectionist models allow researchers to probe the potential range of relations that can exist between behavioural outcomes and their underlying causes, they overturn assumptions of straightforward one-to-one mapping between mechanisms and behaviour and are therefore useful in revealing the "multiplicity underlying unity" in development (Elman et al. 1996, p. 363).

The preceding are but a few examples that identify specific issues in developmental psychology where connectionist tools have demonstrated natural applications. More generally, the resources of connectionism have also been a critical factor in recent attempts to develop a viable interactionist framework for cognitive developmental research. Commenting on the connectionist inspired framework advocated by Elman et al. (1996), Newcombe (1998) points to a recent trend in cognitive developmental theorising that eschews the extremes of nativist and empiricist approaches to learning and cognition, in favour of an account that offers some substantive ideas about the reciprocal actions of organism and environment in producing developmental change. From this standpoint, the resources of connectionism can be seen to contribute to this project by offering researchers a specified, formal account of the developmental process that goes well beyond the verbal accounts typical of developmental theory. Moreover, as Elman et al. (1996) point out, the striking resemblance between the process of error reduction in artificial neural networks and earlier attempts to depict epigenesis in natural systems (e.g., Waddington 1975) offers further evidence of the utility of connectionism for attempts to formalize the interactional nature of development.

The preceding sketch serves to highlight some of the variety of ways in which the computational and conceptual resources of connectionism have been usefully applied in developmental psychology. Yet these pragmatic benefits of connectionist models are not readily apparent in A&L's present evaluation of connectionism against the Newell Test designed to reveal an adequate theory of cognition. As it stands, their evaluation falls short of a comprehensive comparative appraisal of ACT-R as a candidate theory of cognition, and it fails to bring forth the utility of the connectionist toolbox for cognitive science research.

On the encompassing of the behaviour of man

Morten Overgaard[a] and Soeren Willert[b]

[a]*Department of Psychology, University of Aarhus, Asylvej 4, Risskov 8240, Denmark;* [b]*Center for System Development, University of Aarhus, Katrinebjergvej 89GAarhus N 8200, Denmark.* **Overgard@pet.au.dk swi@psy.au.dk www.psy.au.dk/phd/morten www.psy.au.dk/ompi/soeren**

Abstract: One supposition underlying the Anderson & Lebiere (A&L) target article is that the maximally broad "encompassing of its subject matter – the behavior of man" (cf. sect. 6, last para.) is regarded as an unquestioned quality criterion for guiding cognitive research. One might argue for an explicit specification of the limitations of a given paradigm, rather than extending it to apply to as many domains as possible.

Anderson & Lebiere (A&L) set out on an important and admirable mission: to evaluate theories within the more or less well-defined area of cognitive science from one set of criteria in order to avoid a dissolving of theories into disconnected paradigms. We shall not criticise their general idea of measuring comparable theories with a common yardstick, nor the actual grading of ACT-R and connectionism presented by A&L. However, the very approach implies that there is a set of theories that can legitimately be labelled "cognitive theories." To decide whether a given theory falls under the category "cognitive science" and thus decide which theories it would be meaningful to grade with the Newell Test, certain basic requirements must be fulfilled. One could ask whether such basic requirements would be identical to the criteria in the A&L version of the Newell Test. If that were indeed the case, we could have no theory that could truly be called *cognitive* to this day. For instance, we have no theory to explain why consciousness is "a functional aspect of cognition" (let alone one that also explains dynamic behaviour, knowledge integration, etc.) (Chalmers 1996; Velmans 1991). Furthermore, it would be a circular enterprise indeed to measure a theory according to criteria identical to the ones it must already fulfil.

Most likely, however, one would not equate the basic requirements for cognitive science with the criteria of the Newell Test. For such a purpose, the criteria seem to be set much too high. Rather, one would look at the many *different* usages of the term *cognitive* within the research field in general and establish relevant criteria on this basis. This, however, leads us into the situation where we presently stand, that is, a situation where "cognitive science" is loosely defined. We have a number of core theories that definitely are cognitive – such as Treisman's attenuation model (Treisman & Gelade 1980) or the SAS model of visual attention (Norman & Shallice 1986) – and several borderline cases – such as Gibson's ecological perception theory (Gibson 1979) – where it is unclear whether the theory is truly a cognitive psychological theory.

Although our conceptualisation of cognitive science does not seem very exact, it seems safe to say that it has developed historically as an attempt to explain the transition from stimulus to response by "internal variables" (see Tolman 1948). Thus, all cognitive theories – the core cases as well as the less clear-cut ones – intend to give explanations in terms of functions. No matter how the specific theories are construed, all cognitive theories explain the function of some mental phenomenon, whether they collect empirical data from behavioural measures, computer simulations, mathematical models, or brain scannings. This common point of departure has certain consequences for the kind of theory that can be developed. First and foremost, any cognitive theory must be

able to model or causally explain observable behaviour. Response times, button presses, verbal reports, and the like, must be the basis of any such theory; without such third-person information, a cognitive science theory would have nothing to explain.

Returning to the problem of consciousness (or the mind-body problem): Why do certain cognitive and emotional processes have specific experiential or so-called qualitative features? Block (1995) has argued for a difference between so-called access-consciousness (A) and phenomenal consciousness (P). A mental state is A-conscious if it can be poised as premise in reasoning, rational control of action and speech. A mental state is P-conscious if there is something it is *like* to be in that state (Nagel 1974). The mind-body problem is, then, normally interpreted as a problem of explaining how P is related to (other) physical matter.

Any cognitive theory should be able to explain or model what happens when subjects report about consciousness, or about anything else, for that matter. In themselves, however, such explanations or modelling exercises do not necessarily point at anything more than correlations between two sets of psychological third-person data, for example, verbal reports and brain activity. At best, this will give us an understanding of A-consciousness, but not necessarily of P. When describing a cognitive process in terms of its functions or causal processes, P does not fit in unproblematically. Even when turning to some of the more optimistic accounts, one finds arguments that cognitive science can *inform* a solving of the mind-body problem but not actually solve it (Overgaard, in press). Epistemologically speaking, one can easily describe one's experiences exactly without ever referring to the kinds of descriptions and models used by cognitive scientists. Vice versa, one can make a full description of a cognitive process in terms of mathematical models or the often-seen "boxes with arrows between them" without ever referring to experiential qualities. On this basis, one might reasonably question whether an explanation of consciousness is a realistic goal for cognitive science.

For this reason, we are sceptical of one basic supposition underlying the A&L target article: that the maximally broad "encompassing of its subject matter – the behavior of man" (Newell 1973, p. 288, cited in sect. 6, Conclusion, last para.) shall be regarded as an unquestioned quality criterion for theoretical models guiding cognitive research. On the contrary, one might argue that it would be a more theoretically sound approach to explicitly specify the limitations of a given paradigm and its possible openness and connectedness with other paradigms, rather than trying to extend it to apply to as many domains as possible.

The one existing type of language in which *everything* can be spoken about is natural, everyday language. The all-encompassing semantic capacity of natural, everyday language is bought at the price of a low degree of specificity as far as the identification of statements' truth conditions is concerned. The potential utility value of theoretical languages lies in their capacity to isolate and specify knowledge domains characterised by high degrees of epistemic consistency (for scientific purposes) and action predictability (for technological purposes). Definitely, at this stage of cognitive science, we fear this utility value may become jeopardised if success in theory building gets simplistically equated with breadth of coverage.

Connectionism, ACT-R, and the principle of self-organization

Pavel N. Prudkov

Ecomon Ltd., Selskohosyastvennaya str 12-A, Moscow, Russia.
Pnprudkov@mtu-net.ru

Abstract: The target article is based upon the principle that complex mental phenomena result from the interactions among some elementary entities. Connectionist nodes and ACT-R's production rules can be considered as such entities. However, before testing against Newell's macro-criteria, self-organizing models must be tested against criteria relating to the properties of their elementary entities. When such micro-criteria are considered, they separate connectionism from ACT-R and the comparison of these theories against Newell's Tests is hardly correct.

The target article by Anderson & Lebiere (A&L) is devoted to the demonstration of the possibilities of the ACT-R theory. To this end, the authors compare their theory against connectionism on the basis of Newell's criteria for a theory of cognition. However, it is difficult to understand from the article why A&L have decided to select connectionism as a competitor of ACT-R. Indeed, if ACT-R is an unified framework, but the term "connectionism" is "used in the field to refer to a wide variety of often incompatible theoretical perspectives" (target article, sect. 3, para. 7), then A&L could test ACT-R against, for example, a bunch of symbolic models sharing certain common characteristics.

It seems that the main reason for A&L's choice (acknowledged by A&L only partially) is the principle of self-organization, that is, the assumption that complex mental phenomena can be described as a result of the interactions among some elementary entities. This principle has been suggested by me elsewhere (cf. Prudkov 1994), and it was based on the following two facts. First, we know that mental processes are heavily connected to various aspects of brain functioning, though the mechanism of this connection is still unclear. Second, neuroscience data demonstrate that the complex forms of brain activity result from the interactions among some elementary brain entities. Brain areas, single neurons, parts of a neuron, distributions of electrical fields, and the like, can be treated as such entities in accordance with the level of brain functioning considered. It seems impossible to reduce all neural levels to a basic one.

The principle of self-organization requires no correspondence between cognitive elementary entities and any of their neural counterparts, though such correspondence is possible. But all characteristics of a cognitive self-organizing process must result from the properties of its elementary entities and interactions among them, without involving any factors external to the system. The architecture of a self-organizing system is defined by three sorts of characteristics (Prudkov 1994). First, it is necessary to define the elementary entities of the system. Second, the results of the interactions between the entities must be determined. Because the idea of interaction supposes changes in components of the entities, one can say self-organizing models by definition are hybrid. And, third, all conditions or probabilities of the interactions to occur must be described. Learning, then, corresponds to long-term changes in a self-organizing system.

With connectionist nodes as elementary entities, it is intuitively clear that connectionism complies with the principle (a more detailed representation is in Prudkov 1994). With the biological implausibility of many connectionist methods, the principle is likely to be the main reason to use connectionism for understanding cognition (Green 1998). To convert the ACT-R theory into self-organization terms, suppose that production rules are elementary entities, matching the conditions of production rules, and the state of declarative memory determines which entities can interact at the given time. Finally, the rule selected for firing, the result of the firing along with the corresponding changes in declarative memory, is the consequence of an interaction.

Of course, this principle must be considered as a heuristic

rather than an established theory. It allows one to construct a wide variety of models and theories, but their efficiency should be tested against various criteria in order to construct adequate models. To some extent, this principle corresponds to the idea that various physical phenomena stem from the interactions among atoms or molecules. Before 1905, when Einstein proved the existence of these particles, this idea was also a heuristic, but its usefulness for physics is obvious.

However, the idea itself is not sufficient to construct physical models, so these interactions must correspond to various physical laws, such as the laws of thermodynamics. In a similar vein, the self-organizing models of cognition initially must be tested against some criteria relating to the properties of its architecture. Such micro-criteria seem absent (or not stated explicitly) in the target article; however, without using them, the comparison against macro-criteria such as Newell's is hardly correct because of the considerable arbitrariness in the models constructed. For instance, different models can merely describe various levels of the phenomenon under study.

Of course, the theory of cognition still does not have such strict laws as in physics, but several micro-criteria appear useful to judge self-organizing models. The first micro-criterion is the similarity in relevant brain functioning. Since self-organizing models of cognition implicitly refer to self-organizing brain activity which can involve various levels of brain functioning, various models can be compared if their architecture meets the same levels of brain functioning. The architecture of connectionism meets the level of single neurons, but the ACT-R architecture corresponds to cortical regions.

The second micro-criterion is the similarity in the determination of initial settings. Various models can be compared when similar efforts are necessary to establish their initial settings and these settings are equally robust to their changes. The robustness of connectionist settings is well known; ACT-R seems to require more precise but vulnerable settings. For example, the ACT-R model of learning the past tense in English (Taatgen & Anderson 2002) performs well, but the model seems to be vulnerable to the choice of the production rules and learning mechanisms used. It is not obvious that the model with slightly different characteristics could show similar results.

The last micro-criterion assumes that the complexity of entities, interactions, and conditions must be approximately the same in the models judged, or the architecture of one model must naturally result from emergent processes in the other. The architecture of connectionist models is simpler than ACT-R's and, realizing this, A&L describe another model, ACT-RN, which implements ACT-R by standard connectionist methods. However, this implementation seems artificial, for A&L simply predetermine the existence of ACT-R's slots and production rules instead of deriving them from more primitive features of a connectionist model. In principle, A&L simply demonstrate that ACT-RN (and, accordingly, ACT-R) meets the principle of self-organization.

One can conclude that three micro-criteria separate connectionism from ACT-R; these theories describe different levels of cognition; therefore, their direct comparison is hardly correct.

Dual-process theories and hybrid systems

Ilkka Pyysiäinen

Helsinki Collegium for Advanced Studies, University of Helsinki, FIN-00014, Finland. **ilkka.pyysiainen@helsinki.fi**
http://www.helsinki.fi/collegium/eng/staff.htm

Abstract: The distinction between such differing approaches to cognition as connectionism and rule-based models is paralleled by a distinction between two basic modes of cognition postulated in the so-called dual-process theories. Integrating these theories with insights from hybrid systems might help solve the dilemma of combining the demands of evolutionary plausibility and computational universality. No single approach alone can achieve this.

Not only are cognitive scientific "paradigms" disconnected; it also seems to be difficult for a theory of cognition to meet both Newell's criteria 1 and 11. An evolved cognitive architecture apparently cannot be computationally universal (e.g., Bringsjord 2001). Anderson & Lebiere (A&L) thus emphasize that humans can learn to perform almost arbitrary cognitive tasks, but they do not explain why some tasks are easier to learn than others. They suggest that applying a broad enough range of criteria might help us construct an exhaustive theory of cognition, referring to Sun's (1994; 2002) hybrid systems integrating connectionism and a rule-based approach as an example (see also Sun & Bookman 1995). I argue that the distinction between connectionist and functionalist models is paralleled by a distinction between two types of actual cognitive processing, as postulated within the so-called dual-process theories. These theories, developed in social psychology, personality psychology, and neuropsychology, for example, strongly suggest that there are two different ways of processing information, variously labeled

Intuition and implicit learning versus deliberative, analytic strategy (Lieberman 2000)

A reflexive and a reflective system (Lieberman et al. 2002)

Associative versus rule-based systems (Sloman 1996; 1999)

An experiential or intuitive versus a rational mode of thinking (Denes-Raj & Epstein 1994; Epstein & Pacini 1999; Epstein et al. 1992; Simon et al. 1997)

An effortless processing mode that works through associative retrieval or pattern completion in the slow-learning system elicited by a salient cue versus a more laborious processing mode that involves the intentional retrieval of explicit, symbolically represented rules from either of the two memory systems to guide processing (Smith & DeCoster 2000)

Implicit versus explicit cognition (Holyoak & Spellman 1993)

Intuitive versus reflective beliefs (Cosmides & Tooby 2000a; Sperber 1997)

Although the terminologies vary, there is considerable overlap in the substance of these distinctions. The two systems serve different functions and are applied to differing problem domains. They also have different rules of operation, correlate with different kinds of experiences, and are carried out by different brain systems. Some consider these two mechanisms as endpoints on a continuum, whereas Lieberman et al. (2002) argue that they are autonomous systems (see, e.g., Chaiken & Trope 1999; Holyoak & Spellman 1993).

By synthesizing the extant theories, with a special focus on Sloman (1996) and Lieberman et al. (2002), we may characterize the spontaneous system as follows. It operates reflexively, draws inferences, and makes predictions on the basis of temporal relations and similarity; and employs knowledge derived from personal experience, concrete and generic concepts, images, stereotypes, feature sets, associative relations, similarity-based generalization, and automatic processing. It serves such cognitive functions as intuition, fantasy, creativity, imagination, visual recognition, and associative memory (see especially, Sloman 1996). It involves such brain areas as the lateral temporal cortex, amygdala, and basal ganglia. The lateral temporal cortex is, for example, most directly in-

volved in the construction of attributions, whereas the amygdala and basal ganglia are responsible for trying to predict possible punishments and rewards related to one's actions (Lieberman et al. 2002; cf. Rolls 2000).

This system consists of a set of neural mechanisms tuned by a person's past experience and current goals; it is a subsymbolic, pattern-matching system that employs parallel distributed processing. It produces that continuous stream of consciousness we experience as "the world out there," whereas the rational system reacts to the spontaneous system, producing conscious thoughts experienced as reflections on the stream of consciousness (Lieberman et al. 2002). As a pattern-recognition system, the spontaneous system tries to combine all perceived features into a coherent representation; this is because the relevant neurons have been so paired by past experience that the activation of some will also activate others. The spontaneous system cannot consider the causal or conditional relationships between percepts because it does not operate by symbolic logic and because its links are bidirectional. Thus, simply asking a dispositional question (e.g., "Is this man prone to violent behavior?") may easily lead to an affirmative answer (Lieberman et al. 2002).

The rational system involves such brain areas as the anterior cingulate, prefrontal cortex, and hippocampus (Lieberman et al. 2002). It is a rule-based system able to encode any information that has a well-specified formal structure. Such a structure also allows the generation of new propositions on the basis of systematic inferences carried out in a language of thought which has a combinatorial syntax and semantics. It explicitly follows rules. This system thus seeks for logical, hierarchical, and causal-mechanical structure in its environment; operates on symbol manipulation; and derives knowledge from language, culture, and formal systems. It employs concrete, generic, and abstract concepts; abstracted features; compositional symbols; as well as causal, logical, and hierarchical relations. It is productive and systematic; abstracts relevant features; is strategic, not automatic; and serves such cognitive functions as deliberation, explanation, formal analysis, verification, ascription of purpose, and strategic memory (Sloman 1996).

The rational system either generates solutions to problems encountered by the spontaneous system, or it biases its processing in a variety of ways. A pre-existing doubt concerning the veracity of one's own inferences seems to be necessary for the activation of the rational system. The rational system thus identifies problems arising in the spontaneous system, takes control away from it, and remembers situations in which such control was previously required. These operations consist of generating and maintaining symbols in working memory, combining these symbols with rule-based logical schemes, and biasing the spontaneous system and motor systems to behave accordingly (Lieberman et al. 2002).

It could thus be argued that the spontaneous system is a collection of evolved mechanisms with an adaptive background, whereas computational universality is based on the ability of the rational system to exploit the evolved mechanisms to create algorithms for the performance of any cognitive task (see Pinker 1997, pp. 358–359; Atran 2002). This explains the fact that in many areas of everyday life people rely both on evolutionary intuitions and explicit theories. This distinction has recently been studied with regard to peoples' religious intuitions and their theological theories (e.g., Barrett 1998; 1999; Barrett & Keil 1996; Boyer 2001; Pyysiäinen 2003; Whitehouse 2002). Interaction between work on these types of real-life problem fields and on construction of hybrid systems might help us develop more integrated theories of human cognition.

ACKNOWLEDGMENTS

I thank Matthew Lieberman, Marjaana Lindeman, and Markku Niemivirta for help in writing this commentary.

The hardest test for a theory of cognition: The input test

Asim Roy

School of Information Systems, Arizona State University, Tempe, AZ 85287-3606. **asim.roy@asu.edu**

Abstract: This commentary defines an additional characteristic of human learning. The nature of this test is different from the ones by Newell: This is a hard, pass/fail type of test. Thus a theory of cognition cannot partially satisfy this test; it either conforms to the requirement fully, or it doesn't. If a theory of cognition cannot satisfy this property of human learning, then the theory is not valid at all.

The target article by Anderson & Lebiere (A&L) is very refreshing in the sense that it turns the focus back on accountability and tests for any theory of cognition. In examining theories of cognition, a look at system identification in science and engineering may be in order. In system identification, the basic idea is to construct an equivalent system (model) that can produce "behavior" that is similar to the actual system. So the key idea is to produce "matching external behavior." The equivalent system may not necessarily match the internal details of the system to be identified, but that is fine as long as it matches the system's external properties. And the external properties to match may be many. This is not to say that one should not take advantage of any information about the internals of the system.

Therefore, the crucial task for this science is to define the external behavioral characteristics that any system of cognition is supposed to exhibit. Understanding and characterizing the phenomenon to be modeled and explained is clearly the first and main step towards developing a theory for it. If that is not done, it is very likely that wrong theories will be proposed, because it is not known exactly what the theory should account for. This commentary defines an additional characteristic of human learning other than the ones in the Newell Test (Roy et al. 1997). In the spirit of the Newell Test, this is a characteristic of the brain that is "independent of" (1) any conjectures about the "internal" mechanisms of the brain (theories of cognition) and (2) the specific learning task. That is, this property of human learning is independent of a specific learning task like learning a language, mathematics, or a motor skill. The nature of this test is quite different from the ones provided by Newell: This is a hard, pass/fail type of test. In that sense, a theory of cognition cannot partially satisfy this test; it either conforms to its requirement fully, or it doesn't. This pass/fail test would allow one to quickly check the validity of alternative theories of cognition. If a theory of cognition cannot satisfy this property of human learning, then the theory is not valid at all. So this particular test is good enough for initial screening of theories. As explained in the following paragraphs, classical connectionism fails this test. One has to take a closer look at ACT-R and ACT-RN to pass judgment on them.

So what is this real hard test for theories of cognition? It can be summarized as follows: A brain-like system, constructed on the basis of some theory of cognition, is not permitted to use any inputs in its construction phase that are not normally supplied to a human brain. So the real hard test for any theory is in the inputs required to construct the relevant system of cognition. Let this test be called the "Input Test." The human brain has two sources of inputs during its development, both inside the womb and outside. Biological parents are the first source, and certain structures and systems can be inherited through that source. The other source of inputs for its development is the environment after birth. So any theory of cognition has to clearly delineate what pieces of its functioning system are inherited from biological parents and what pieces are developed subsequently through interactions with the environment. For humans, it is known for a fact that certain functionality of the brain is definitely not inherited, like the ability to speak a certain language, do mathematics, and so on. The modules for these functionalities/tasks do not come pre-built in the hu-

man brain; rather, they are developed and constructed gradually over time. So, to reiterate this point, the first task of a theory of cognition is to clearly delineate what pieces of its functioning system are inherited and what pieces are developed subsequently through interactions with the environment. And with regard to what can come pre-built (inherited), it has to provide sensible arguments.

Once a proposed theory of cognition maps out what is pre-built in the system in the sense of being inherited from biological parents, then the problem for the theory is to show how it develops and constructs the modules that are not pre-built. And whatever the means are for developing and constructing these modules, the hardest test for the theory is this: It has to demonstrate that it is not using any inputs for developing and constructing these modules that are not provided to humans from the environment. This input test can be explained nicely by examining classical connectionism. In classical connectionism, for example, network designs and other algorithmic information have to be externally supplied to the learning system, whereas no such information is ever an external input to the human brain. The well-known back-propagation algorithm of Rumelhart et al. (1986) is a case in point. In fact, many different network designs and other parameter values often have to be supplied to these learning systems on a trial-and-error basis in order for them to learn. However, as far as is known, no one has ever been able to externally supply any network designs or learning parameters to a human brain. So classical connectionism clearly violates this input test and is not a valid theory of cognition.

In general, for previously unknown tasks, the networks could not feasibly come predesigned in human brains; thus network designs cannot be inherited for every possible unknown learning problem faced by the brain on a regular basis. And the networks required for different tasks are different; it is not a one-size-fits-all situation. Since no information about the design of a network is ever supplied to the brain externally, it therefore implies that the brain performs network designs internally. Thus, it is expected that any theory of cognition must also demonstrate the same ability to design networks and adjust its own learning parameters without any outside intervention. But the connectionist learning systems can't demonstrate this capability, and that again implies that classical connectionism is not a valid theory of cognition.

In summary, in this input test, a theory of cognition should be restricted to accepting information that is normally supplied to a human from the environment, nothing more.

Rethinking learning and development in the Newell Test

Sylvain Sirois

Department of Psychology, The University of Manchester, Manchester M13 9PL, United Kingdom. **sylvain.sirois@man.ac.uk**
http://www.psy.man.ac.uk/staff/sirois.htm

Abstract: The Newell Test is an ambitious and promising project, but not without pitfalls. Some of the current criteria are not theoretically neutral, whereas others are unhelpful. To improve the test, the learning and development criteria are reviewed and revised, which suggests adding a maturation criterion as well. Such changes should make the Newell Test more general and useful.

Anderson & Lebiere (A&L) have certainly embarked on an ambitious project: to transform Newell's (1980; 1990) functional criteria for human cognitive architectures into the ultimate test of cognitive theories. I certainly sympathise with such ambitions, especially given their emphasis on the functional aspects of the criteria that should be used. For example, we recently conducted a similar (albeit substantially more humble) exercise for models of infant habituation (Sirois & Mareschal 2002). We identified a set of seven behavioural and neural criteria that functional models of the phenomena need to satisfy. This proved extremely useful to highlight the limitations of current models, but also (and perhaps more importantly) to suggest what the next generation of models needed to address. Given the relatively small scale of the problem addressed in our work, one could conceivably expect huge and varied rewards from A&L's far more valiant endeavour.

Whereas the rewards may prove an exponential function of those we observe in analogous but restricted projects, so may the problems. The authors quite rightly acknowledge that their criteria (which are a slightly modified version of Newell's) are not the only criteria by which a theory can be assessed. But far more crucial than how many criteria (which makes the test more or less liberal) is the question of *which* criteria (which makes the test more or less useful). If the stated goal of such a test is to avoid theoretical myopia, then a few of the criteria are certainly problematic because they either imply that a model adheres to a specific school of thought or to tests of models against highly disputable standards. For example, *knowledge integration* may have been retitled from Newell (1990) but owes no less to symbolic tradition than when it was proposed by Newell. As such, the grading of this criterion is unduly biased towards models and theories originating from this tradition. The *consciousness* criterion is even more problematic: Whether the criterion has any functional value depends on an eventual theory that would make such a suggestion!

Other commentators will likely address the relevance or appropriateness of the various criteria, if not of the test itself. Despite inherent difficulties in such projects, I believe that a revised formulation of the Newell Test could be quite useful. I would thus like to focus on two criteria that, in my view, should be kept in the Newell Test: *learning* and *development*. Surprisingly, the authors evacuated the functional role of learning in their discussion. Moreover, they discuss development as a (perhaps functional) constraint rather than as a functional mechanism. In fact, what they present as development sounds remarkably like maturation.

The authors should not be blamed too harshly for reproducing a common problem in developmental psychology: confounding learning and development by discussing them in terms of *outcomes* rather than *mechanisms* (Liben 1987). This is most explicit when they present the slow learning of *classical connectionism* as satisfying the development criterion. If anything, and contrary to what the authors suggested, the sort of learning in classical connectionism can be characterised as a nativist learning theory (Quartz 1993; Sirois & Shultz 1999).

Fortunately, the notions of learning and development can be expressed formally as non-overlapping functions (Sirois & Shultz 1999). *Learning* can be defined as parametric changes that enable a given processing structure to adapt to its environment. *Development*, however, can be defined as structural changes that foster more complex adaptations, given learning failure. These definitions not only constrain the contribution of each mechanism to cognitive change, but also specify the relationship between learning and development. Learning causes the current structure to adapt, but when that fails, development alters the structure to promote further learning. It must be noted that either form of change is a function of experience. Within this framework, then, *maturation* becomes an experience-independent structural change that delays learning, in line with what A&L discussed as development.

Like others (including A&L), I believe that an adequate theoretical formulation of cognition must be consistent with learning and developmental issues. Moreover, given the significant changes that can be introduced by maturation (i.e., the cognitive structure increases in complexity), I would suggest that the Newell Test also incorporates maturation as one of its criteria. The grading is relatively straightforward for the learning, development, and maturation criteria. If a theory allows for parametric changes as a function of experience, it can learn. If it allows for experience-dependent structural changes that support further learning, it satisfies development. Finally, if it allows for experience-independent, programmed structural changes that modify the learning space, it satisfies maturation.

These learning, development, and maturation criteria are general by design, and so are the grading proposals, in line with Newell's wish to avoid theoretical myopia. A cognitive theory should be granted with the ability to satisfy any of these criteria if it satisfies the relevant functional properties, irrespective of how the mechanisms are actually realised. This general nature does not imply that the criteria are vague, however. We initially proposed these definitions to discuss various classes of neural networks as they are applied to developmental problems. We found that the classical connectionist framework only satisfied the learning criteria (Sirois & Shultz 1999). But we applied the same framework to discuss the various mechanisms of Piagetian theory, clarifying them in the process, and allowing for a formal distinction between learning and developmental notions in Piaget's work (Sirois & Shultz 2003). If we apply these definitions to ACT-R as discussed by A&L, we could grant ACT-R with the ability to satisfy learning and developmental criteria (the latter through the construction of new rules).

To summarise, the idea of a Newell Test is quite attractive but not without design pitfalls. Whereas there may be some inadvertent myopia in the choice of criteria, most of these may well be retained (but perhaps reformulated). The peer commentaries in this journal will hopefully provide the next few steps towards the design of a generally satisfying test of cognitive theories.

ACKNOWLEDGMENT

I thank Isabelle Blanchette for useful comments on an earlier draft.

What about embodiment?

David Spurrett

Philosophy Department, University of Natal, Durban, 4041, South Africa.
spurrett@nu.ac.za **http://www.nu.ac.za/undphil/spurrett/**

Abstract: I present reasons for adding an *embodiment* criterion to the list defended by Anderson & Lebiere (A&L). I also entertain a likely objection contending that embodiment is merely a type of *dynamic behavior* and is therefore covered by the target article. In either case, it turns out that neither connectionism nor ACT-R do particularly well when it comes to embodiment.

The principle that cognitive theories should be evaluated according to multiple criteria is worth adopting, and Anderson & Lebiere's (A&L's) development of Newell's proposals in this regard is useful. One important criterion seems to be missing, though, and that is *embodiment.*

By embodiment, I understand, loosely, physical implementation in an environment. Humans, clearly a key consideration of the target article, are, of course, embodied. They exhibit striking virtuosity at moving around the world and exploiting the resources available in it. Perhaps more important for present purposes, we are talented at exploiting the structure of environments (and of our bodies in them) for cognitive ends, or as some would have it, engaging in "distributed cognition" (e.g. Hutchins 1995). One example is locomotion, where recent research (Thelen & Smith 1994) indicates that the architecture of the body, and the properties of the body in interaction with the environment, play significant roles in control of behavior. Another example, rather closer to the concerns of traditional cognitive science, is the game of Tetris, where it has been shown (Kirsh & Maglio 1994) that human players use external actions to improve the efficiency (speed, accuracy, error rate) of the spatial manipulations and judgements demanded by the game. External rotation of a Tetris piece, along with inspection to establish whether the rotated piece is in a preferable orientation (compared to before), is often faster and less error-prone than mental rotation for the same purpose. This suggests that at least some cognitive problems are tackled using a coalition of internal and external resources, and that an important feature of our cognitive makeup is that we can detect opportunities for this. (Further examples in humans, other animals, and (some) robots abound. Clark [1997] is a useful survey.) This in turn indicates that a theory of cognition that fails to take embodiment seriously is unlikely to capture such features of our own cognitive performance.

A likely objection here notes that A&L's criterion 5 is "dynamic behavior." Since this criterion concerns the relationship between a cognitive system and an environment, perhaps, properly understood, it includes embodiment and distributed cognition. Distributed cognition just *is,* the objection goes – a kind of dynamical coupling between an information-processing system and a structured body and environment. This objection may be taking charitable interpretation too far. A&L's discussion of their "dynamic behavior" criterion (sect. 2.5 of the target article) places considerable emphasis on dealing with the unexpected, and relatively less on exploiting external structure. When evaluating the relative performance of classical connectionism and ACT-R with respect to the dynamic behavior criterion (sect. 5.5 of the target article), their emphasis is on real-time control, not embodiment. Rather than try to settle the question whether embodiment is or is not a version of dynamic behavior, I propose to consider how connectionism and ACT-R fare in the case where embodiment is added as a separate criterion, and where dynamic behavior is interpreted to include it.

Were embodiment added as a criterion, I suggest that connectionism would achieve mixed results. In some cases it does extraordinarily well. Consider Quinn and Espenschied's (1993) neural network for controlling a hexapod robot. The success of this system depends to a significant extent on allowing features of the physical construction of the robot, in interaction with the environment, to play a role in control – so that the motion of individual feet will be inhibited if other specific feet do not yet have secure positions. One way of understanding this is to regard the changing physical links between some neurons, parts of the robot anatomy, the physical environment, other parts of the anatomy and (eventually, and sometimes) other neurons, as functioning like additional neurons, or interneuron connections, transforming or transmitting information about footing, load on joints, and so on. In other cases, though, it is not (yet) clear how to go about building a network, embodied or otherwise, to handle tasks (such as air traffic control) involving fairly specific and detailed functional decomposition, tasks for which systems such as ACT-R seem well suited.

ACT-R, I argue, scores worse for embodiment. Its successes at, for example, modelling driving are in constrained simulation environments, where embodied interaction with the "feel" of the vehicle and its relation to the road surface, are absent, and where attendant opportunities for exploiting environmental structure (engine tone, vibration) to help cue such actions as gear changes are absent for both the human subjects who provide the target data, and the ACT-R models of driving behavior which do well at approximating the behavior of such humans.

However, we might reinterpret A&L's "dynamical behavior" criterion in a way that includes embodiment as a subtype of dynamic behavior. In this case, and in the light of what is said in the target article and so far in this commentary, connectionism should retain its mixed score. In this case ACT-R should also, I argue, receive a mixed score: It doesn't do well at plain embodiment, but does better at non-embodied forms of dynamic behavior. In either case, the moral to draw is that if embodiment is a genuinely important criterion, then *neither* connectionism nor ACT-R seem, as they stand, in a good position to perform consistently well on it.

Conceptions and misconceptions of connectionism

Ron Sun

CECS Department, University of Missouri-Columbia, Columbia, MO 65211.
rsun@cecs.missouri.edu **http://www.cecs.missouri.edu/~rsun**

Abstract: This commentary examines one aspect of the target article – the comparison of ACT-R with connectionist models. It argues that conceptions of connectionist models should be broadened to cover the whole spectrum of work in this area, especially the so-called hybrid models. Doing so may change drastically ratings of connectionist models, and consequently shed better light on the developing field of cognitive architectures.

John Anderson has been one of the pioneers of cognitive architectures. His and Christian Lebiere's work on ACT-R has been highly influential. In many ways, their work defines this field today.

However, instead of going on praising ACT-R, I shall here focus on shortcomings of the target article. One shortcoming, as I see it, is in Anderson & Lebiere's (A&L's) treatment of connectionist models or, more precisely, in their very conception of connectionist models. In the target article, as a comparison to ACT-R, A&L focus exclusively on what they term "classical connectionism" (which I would call "strong connectionism") – the most narrowly conceived view of connectionist models, from the mid-1980s, as articulated by the classic PDP book (Rumelhart & McClelland 1986). In this view, connectionist models are the ones with regular network topology, simple activation functions, and local weight-tuning rules. A&L claim that this view "reflects both the core and the bulk of existing neural network models while presenting a coherent computational specification" (target article, sect. 3, last para.).

However, it appears that connectionist models conforming to this view have some fundamental shortcomings. For example, the limitations due to the regularity of network topology led to difficulty in representing and interpreting symbolic structures (despite some limited successes so far). Other limitations are due to learning algorithms used by such models, which led to lengthy training (with many repeated trials), requiring a priori input/output mappings, and so on. They are also limited in terms of biological relevance. These models may bear only remote resemblance to biological processes.

In coping with these difficulties, two forms of connectionism became rather separate: Strong connectionism adheres closely to the above strict precepts of connectionism (even though they may be unnecessarily restrictive), whereas weak connectionism (or hybrid connectionism) seeks to incorporate both symbolic and subsymbolic processes – reaping the benefit of connectionism while avoiding its shortcomings. There have been many theoretical and practical arguments for hybrid connectionism (see, e.g., Sun 1994). Considering our lack of sufficient neurobiological understanding at present, a dogmatic view on the "neural plausibility" of hybrid connectionist models is not warranted. It appears to me (and to many other people) that the death knell of strong connectionism has already been sounded, and it is time now for a more open-minded framework without the strait-jacket of strong connectionism.

Hybrid connectionist models have, in fact, been under development since the late 1980s. Initially, they were not tied into work on cognitive architectures. The interaction came about through some focused research funding programs by funding agencies. Several significant hybrid cognitive architectures have been developed (see, e.g., Shastri et al. 2002; Sun 2002; Sun et al. 2001).

What does this argument about the conception (definition) of connectionism have to do with ratings on the Newell Test? In my own estimate, it should affect ratings on the following items: "a vast amount of knowledge," "operating in real time," "computational universality," "integrating diverse knowledge," and possibly other items as well. Let's look into "a vast amount of knowledge," as an example. What may prevent neural networks from scaling up and using a vast amount of knowledge is mainly the well-known problem of catastrophic interference in these networks. However, the problem of scaling and "catastrophic interference" in neural networks may in fact be resolved by modular neural networks, especially when symbolic methods are introduced to help partition tasks (Sun 2002). With different subtasks assigned to different networks that are organized in a modular fashion, catastrophic interference can be avoidable. Thus, if we extend the definition of connectionist models, we can find some (partial) solutions to this problem, which are (at least) as good as what is being offered by ACT-R to the same problem. Similar things may be said about "integrating diverse knowledge" or "operating in real time," and so on. Overall, when our conceptions of connectionist models are properly expanded, our ratings of connectionist models will have to be changed accordingly too; hence the significance of this issue to the target article.

A related shortcoming of the target article is the lack of adequate discussion and rating of hybrid connectionist models besides ACT-R. Ratings of these models and comparisons with ACT-R can shed further light on the strengths and weaknesses of different approaches. There have been some detailed analyses and categorizations of hybrid connectionist models, which include "classical" connectionist models as a subset, that one might want to look into if one is interested in this area (see, e.g., Sun & Bookman 1994; Wermter & Sun 2000).

Finally, I would like to echo the authors' closing remarks in the conclusion (sect. 6) of the article: If researchers of all theoretical persuasions try to pursue a broad range of criteria, the disputes among theoretical positions might simply dissolve. I am confident that the target article (and more importantly, this entire treatment) may in fact contribute toward this end.

ACKNOWLEDGMENT

This work was supported in part by ARI contract DASW01–00–K–0012.

Poppering the Newell Test

Niels A. Taatgen

Department of Artificial Intelligence, University of Groningen, 9712 TS Groningen, The Netherlands. **niels@ai.rug.nl**
http://www.ai.rug.nl/~niels

Abstract: The Newell Test as it is proposed by Anderson & Lebiere (A&L) has the disadvantage of being too positivistic, stressing areas a theory should cover, instead of attempting to exclude false predictions. Nevertheless, Newell's list can be used as the basis for a more stringent test with a stress on the falsifiability of the theory.

The idea of the Newell Test is obviously inspired by its illustrious predecessor, the Turing Test (Turing 1950) and can be considered as an elaboration of the topics that have to be addressed by a theory to make it a plausible basis for an intelligent machine. There is a subtle difference between the two tests: Although the Turing Test stresses the fact that the computer should be able to make meaningful conversation, the main point is that the judge in the Turing Test is supposed to do everything possible to expose the computer as a fraud. This aspect of the test is very important, because noncritical discussion partners of the computer can easily be fooled by programs like ELIZA (Weizenbaum 1966; also see Lodge 1984) and its successors. Analogous to the Turing Test, the Newell Test has two aspects: a positivistic aspect (i.e., the theory should allow models of all areas of cognition) and a falsifiability aspect (i.e., the theory should restrict and eventually disallow all "false" models) (Popper 1963). The latter aspect, however, has much less prominence in the Newell Test than the former. I would like to criticize this and argue that the aspect of excluding false models is at least as important, and maybe much more important, than permitting true models.

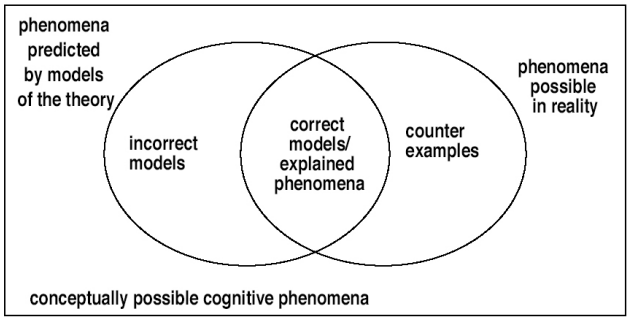


Figure 1 (Taatgen). Diagram to illustrate successes and problems of a theory of cognition.

Figure 1 illustrates the issue. Consider the set of all possibly conceivable cognitive phenomena, of which only a subset contains phenomena that can actually occur in reality. Then the goal of a theory is to predict which of the conceivable phenomena are actually possible, and the success of a theory depends on the overlap between prediction and reality. The problems of a theory can be found in two categories: counterexamples, phenomena that are possible in reality but are not predicted by the theory, and incorrect models, predictions of the theory that are not possible in reality. The issue of incorrect models is especially important, because an unrestricted Turing Machine is potentially capable of predicting any conceivable cognitive phenomenon. One way to make the Newell Test more precise would be to stress the falsifiability aspects for each of the items on the test. For some items this is already more or less true in the way they are formulated by Anderson & Lebiere (A&L), but others can be strengthened, for example:

Flexible behavior. Humans are capable of performing some complex tasks after limited instructions, but other tasks first require a period of training. The theory should be able to make this distinction as well and predict whether humans can perform the task right away or not.

Real-time performance. The theory should be able to predict human real-time performance, but should not be able to predict anything else. Many theories have parameters that allow scaling the time predictions. The more these parameters are present, the weaker is the theory. Also the knowledge (or network layout) that produces the behavior can be manipulated to adjust time predictions. Restricting the options for manipulation strengthens the theory.

Knowledge integration. One property of what A&L call "intellectual combination" is that there are huge individual differences. This gives rise to the question how the theory should cope with individual differences: Are there certain parameters that can be set that correspond to certain individual differences (e.g., Lovett et al. 1997; Taatgen 2002), or is it mainly a difference in the knowledge people have? Probably both aspects play a role, but it is of chief importance that the theory should both predict the breadth and depth of human behavior (and not more).

Use natural language. The theory should be able to use natural language but should also be able to assert what things cannot be found in a natural language. For example, the ACT-R model of learning the past tense shows that ACT-R would not allow an inflectional system in which high-frequency words are regular and low-frequency words are irregular.

Learning. For any item of knowledge needed to perform some behavior, the theory should be able to specify how that item has been learned, either as part of learning within the task, or by showing why it can be considered as knowledge that everyone has. By demanding this constraint on models within a theory, models that have unlearnable knowledge can be rejected. Also, the learning system should not be able to learn knowledge that people cannot learn.

Development. For any item of knowledge that is not specific to a certain task, the theory should be able to specify how that item of knowledge has been learned, or to supply evidence that that item of knowledge is innate. This constraint is a more general version of the learning constraint. It applies to general strategies like problem solving by analogy, perceptual strategies, memorization strategies, and the like.

Another aspect that is of importance for a good theory of cognition is parsimony. This is not an item on Newell's list, because it is not directly tied to the issue of cognition, but it was an important aspect of Newell's research agenda. This criterion means that we need the right number of memory systems, representations, processing, and learning mechanisms in the theory, but not more. An advantage of parsimony is that is makes a stronger theory. For example, SOAR has only one learning mechanism, chunking. This means that all human learning that you want to explain with SOAR has to be achieved through chunking, as opposed to ACT-R, which has several learning mechanisms. Of course, SOAR's single mechanism may eventually be found lacking if it cannot account for all human learning.

To conclude, research in cognitive modeling has always had a positivistic flavor, mainly because it is already very hard to come up with working models of human intelligence in the first place. But as cognitive theories gain in power, we also have to face the other side of the coin: to make sure that our theories rule out wrong models. This is not only an issue for philosophers of science, but a major issue if we want to apply our theories in human-computer interaction and education. There, it is of vital importance that we should be able to construct models that can provide reliable predictions of behavior without having to test them first.

Cognitive architectures have limited explanatory power

Prasad Tadepalli

School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR 97331-3202. **tadepall@cs.orst.edu**
http://www.eecs.orst.edu/~tadepall

Abstract: Cognitive architectures, like programming languages, make commitments only at the implementation level and have limited explanatory power. Their universality implies that it is hard, if not impossible, to justify them in detail from finite quantities of data. It is more fruitful to focus on particular tasks such as language understanding and propose testable theories at the computational and algorithmic levels.

Anderson & Lebiere (A&L) undertake the daunting task of evaluating cognitive architectures with the goal of identifying their strengths and weaknesses. The authors are right about the risks of proposing a psychological theory based on a single evaluation criterion. What if the several micro-theories proposed to meet different criteria do not fit together in a coherent fashion? What if a theory proposed for language understanding and inference is not consistent with the theory for language learning or development? What if a theory for playing chess does not respect the known computational limits of the brain? The answer, according to Newell, andA&L, is to evaluate a cognitive theory along multiple criteria such as flexibility of behavior, learning, evolution, knowledge integration, brain realization, and so forth. By bringing in multiple sources of evidence in evaluating a single theory, one is protected from *overfitting*, a problem that occurs when the theory has too many degrees of freedom relative to the available data. Although it is noncontroversial when applied to testable hypotheses, I believe that this research strategy does not work quite as well in evaluating cognitive architectures.

Science progresses by proposing testable theories and testing them. The problem with cognitive architectures is that they are not theories themselves but high-level languages used to imple-

ment theories, with only some weak architectural constraints. Moreover, these languages are computationally universal and thus are equivalent to one another in the sense that one language can simulate the other. How does one evaluate or falsify such universal languages? Are the multiple criteria listed by the authors sufficient to rule out anything at all, or do they simply suggest areas to improve on? The authors' grading scheme is telling in this respect. It only evaluates how an architecture satisfies one criterion better than another criterion, and does not say how to choose between two architectures. One cannot, of course, duck the question merely by choosing an architecture based on the criterion one is interested in explaining. This is precisely the original problem that Newell was trying to address through his multiple criteria.

The authors suggest that timing constraints and memory limitations imply that one cannot only program arbitrary models in ACT-R. But that still leaves room for an infinite variety of models, and ACT-R cannot tell us how to choose between them. To take an analogy to programming languages: it is possible to design an infinite variety of cognitive architectures and implement an infinite variety of models in each one. Can we ever collect enough evidence to be able to choose one over another?

This suggests to me that a cognitive theory must be carefully distinguished from the concrete implementation and the underlying architecture. Just as a programming language can implement any given algorithm, a cognitive architecture can instantiate any cognitive theory (albeit with some variations in time efficiencies). This should not count as evidence for the validity of the architecture itself, any more than good performance of an algorithm should count as evidence for the validity of the programming language. Cognitive science can make better progress by carefully distinguishing the algorithm from the architecture and confining the claims to those parts of the algorithm that are in fact responsible for the results. Consider, for example, ACT-R's theory of past-tense learning by children. More specifically, consider the empirical observation that the exceptions tend to be high-frequency words. A&L attribute this to the fact that only high-frequency words develop enough base-level activation to be retrieved in ACT-R. In more general terms, only high-frequency words provide sufficient training data for the system to be able to learn an exception. How much of this explanation is a result of the particulars of ACT-R theory as opposed to being a necessary consequence of learning in a noisy domain? If any learning system that operates in a noisy environment needs more training data to learn an exception, why should this be counted as evidence for the ACT-R theory? Similar criticisms can be leveled against other cognitive architectures and mechanisms such as SOAR and chunking, connectionism and backprop.

In other words, even when multiple criteria are used to evaluate a cognitive architecture, there still remains an explanatory gap (or a leap of faith) between the evidence presented and the paradigm used to explain it. To guard against such over-interpretation of the evidence, Ohlsson and Jewett propose "abstract computational models," which are computational models that are designed to test a particular hypothesis without taking a stand on all the details of a cognitive architecture (Ohlsson & Jewett 1997). Similar concerns are expressed by Pat Langley, who argues that the source of explanatory power often lies not in the particular cognitive architecture being advanced but in some other fact such as the choice of features or the problem formulation (Langley 1999). Putting it another way, there are multiple levels of explanations for a phenomenon such as past-tense learning or categorization, including computational theory level, algorithmic level, and implementation level. Computational theory level is concerned with *what* is to be computed, whereas algorithmic level is concerned with *how* (Marr 1982). Cognitive architecture belongs to the implementation level, which is below the algorithmic level. Where the explanatory power of an implementation mostly lies is an open question.

Only by paying careful attention to the different levels of explanations and evaluating them appropriately can we discern the truth. One place to begin is to propose specific hypotheses about the algorithmic structure of the task at hand and evaluate them using a variety of sources of evidence. This may, however, mean that we have to put aside the problem of evaluating cognitive architectures, for now or forever.

ACKNOWLEDGMENTS

I thank Sridhar Mahadevan and Pat Langley for influencing my thinking on this matter and for their comments on an earlier draft.

Cognitive modelling of human temporal reasoning

Alice G. B. ter Meulen

Center for Language and Cognition, University of Groningen, 9700 AS Groningen, The Netherlands. **atm@let.rug.nl** **http://atm.nemit.net**

Abstract: Modelling human reasoning characterizes the fundamental human cognitive capacity to describe our past experience and use it to form expectations as well as plan and direct our future actions. Natural language semantics analyzes dynamic forms of reasoning in which the real-time order determines the temporal relations between the described events, when reported with telic simple past-tense clauses. It provides models of human reasoning that could supplement ACT-R models.

Real-time performance, the second criterion for a human cognitive architecture in Newell (1990), requires the system to operate as fast (or as slow) as humans (target article, sect. 2, Table 1) on any cognitive task. Real time is hence considered a constraint on learning as well as on performance (sect. 5). Although I certainly consider it an advantage of the ACT-R system that it does not rely on artificial assumptions about presentation frequency in the way classical connectionist systems do (Taatgen & Anderson 2002), the limited focus the two systems share on the acquisition of the morphological variability in the simple past-tense inflection in English ignores its obvious common semantic properties, which also must be learned. In this commentary, I propose to include in real-time performance the characteristic human ability to use time effectively when using language to encode information that systematically depends on contextual parameters, such as order of presentation or time of utterance.

Human linguistic competence includes automated processes of temporal reasoning and understanding, evidence of which is presented in our linguistic intuitions regarding the temporal relations that obtain between events described in coherent discourse. The presentation order in which simple past-tense clauses are produced in real time often contains important clues for the correct interpretation. As opposed to the past progressive (*John was leaving*) and the past perfect (*John had left*), the English simple past tense (*John left*) refers to an event that not only precedes the time of utterance but also is temporally located with respect to other events described by prior discourse. The following examples, (1) and (2), show that the order of presentation affects our understanding of what happened.

(1) *John lit a cigarette. He left.*
(2) *John left. He lit a cigarette.*

From (1) we understand that John left after he had lit a cigarette. But (2) makes us understand that the described events occurred in the opposite order. Obviously, the real-time order of presentation in this case determines the temporal relations between the events described. But this is not always so, as we see from examples (3) and (4), where reversing the order of the simple past-tense clauses does not affect the temporal relations between the events.

(3) *John slept for hours. He dreamt of Mary.*
(4) *John dreamt of Mary. He slept for hours.*

Either (3) or (4) makes us understand that John dreamt of Mary while he slept, which is reinforced by the lexical presupposition of dreaming requiring that the dreamer be asleep.

The differences observed between the interpretations of (1)–(4), coincidentally all morphologically strong past-tense inflections, are attributed to the aspectual class of the clauses, which may be telic or atelic (Partee 1984; Hinrichs 1986). Although the compositional characterization of telicity has been a core item on the linguistic research agenda for quite some time, it is generally agreed that in English, clauses that may be modified by durative adverbials, such as *for hours,* are atelic, and clauses that are unacceptable with durative modifiers are telic (ter Meulen 1995; Verkuyl 1996). Temporal precedence effects, which conceptually shift the reference time, are determined by order of presentation of telic clauses in simple past-tense clauses.

Children gradually learn to produce cohesive discourse with simple past-tense clauses, effectively using order of presentation, instead of connecting clauses in their stories with *and the . . . and then . . . É.* It depends on their understanding of logical or causal relations between lexical items; for example, dreaming entails sleeping, leaving entails moving elsewhere. It also requires mastering deductive or abductive forms of reasoning, into which neither classical connectionism nor ACT-R have many modelling insights to offer, as Anderson & Lebiere (A&L) readily admit. Reasoning in context and exploiting the dependencies between tense and other indexical features of linguistic expressions cannot be reduced to conditioned correlations between lexical items and concepts, as classical connectionists may want to argue, because it needs a representation of the agent's own information structured information state, as well as a representation of the external domain described by linguistic input and other agents it communicates with. Human understanding of information communicated in ordinary language discourse should, therefore, constitute a core task on the common agenda of cognitive science, testing not only Newell's criteria of real-time performance and natural language, but also adaptive, dynamic, and flexible behavior, as well as knowledge integration and development. Natural language semantics is studying the structured dependencies between context, information, and described domain (Asher et al. 1994; van Eijck & Kamp 1997; ter Meulen 2000). The "Dynamic Turn" in the semantics of both formal-logical, and natural languages has profoundly changed the agenda of the traditional logical systems to require that a dynamic semantics of natural language ideally provides abstract models of our human cognitive capacities of information processing, envisaged in Partee (1980; 1997;) as the program to "naturalize formal semantics." ACT-R accounts of human cognition may well find it a congenial companion, supplementing its self-proclaimed need for an account of human reasoning.

Real-world behavior as a constraint on the cognitive architecture: Comparing ACT-R and DAC in the Newell Test

Paul F. M. J. Verschure

Institute of Neuroinformatics, University Zürich–Swiss Federal Institute of Technology (ETH), Zürich, 8057, Switzerland. **pfmjv@ini.phys.ethz.ch**
http://www.ini. ethz.ch/~pfmjv

Abstract: The Newell Test is an important step in advancing our understanding of cognition. One critical constraint is missing from this test: A cognitive architecture must be self-contained. ACT-R and connectionism fail on this account. I present an alternative proposal, called Distributed Adaptive Control (DAC), and expose it to the Newell Test with the goal of achieving a clearer specification of the different constraints and their relationships, as proposed by Anderson & Lebiere (A&L).

Anderson & Lebiere (A&L) make the important step to resurrect a number of benchmarks, originally proposed by Newell, which a theory of cognition should satisfy. One benchmark that is missing from this list is that the proposed architecture must be self-contained. *Self-contained* implies that the knowledge of the cognitive system is acquired through an autonomous learning process; that is, its ontology is derived from the interaction between the system and the world. Both ACT-R and classical connectionism do not score well on this constraint. ACT-R fails because it focuses on the use of predefined knowledge in its productions and its recombination by means of chunking. The implementation of its memory structures using artificial neural networks and the inclusion of a subsymbolic/symbolic nomenclature does not address this problem. Classical connectionism fails because it relies on learning rules, for example, backpropagation, that allow the user to compile a predefined input-output mapping into the model (Verschure 1990; 1992). In both cases the models do not tell us how knowledge is acquired in the first place. One could argue that solving this problem of priors is the most fundamental challenge to any candidate theory of cognition (Verschure 1998).

In order to challenge the authors to define more precisely what it takes to satisfy the Newell Test, I present an alternative proposal for a cognitive architecture, called Distributed Adaptive Control (DAC). DAC describes an embodied cognitive architecture implemented by a neuronal system in the context of real-time, real-world behavior. DAC assumes that behavior is organized around three tightly coupled layers of control: reactive, adaptive, and contextual (Fig. 1A). The typical paradigms in which we have developed this architecture are robot equivalents of random foraging tasks (Fig. 1B). It should be emphasized that DAC develops its own domain ontology out of its continuous interaction with the world. Hence, as opposed to ACT-R, DAC is self-contained.

Flexible behavior ("better"). DAC has been shown to organize landmark-based foraging behavior in different types of robots (Verschure et al. 1992; 1996; Verschure & Voegtlin 1998), has been applied to simple games such as tic-tac-toe (Bouvet 2001), has controlled a large scale public exhibit (Eng et al. 2003), and has been shown to be equivalent to an optimal Bayesian interpretation of goal-oriented problem solving (Verschure & Althaus 2003). By satisfying this last constraint, DAC implicitly addresses a wide range of cognitive phenomena (Massaro 1998). This latter constraint argues that our models should attack abstract models describing large repertoires of performance as opposed to single instances of particular behaviors.

Real-time performance ("better"). As opposed to ACT-R, DAC takes real time literally as the time it takes to control real-world behavior. In biologically detailed models, derived from the DAC architecture, of both the sensory (i.e., the learning-dependent changes in receptive field properties of the primary auditory cortex, as reported by Kilgard & Merzenich 1998) and motor aspects (focusing on the cerebellum) of classical conditioning, we have shown that these principles can account for learning performance both in terms of number of trials and in terms of the relevant real-time interstimulus intervals (Sanchez-Montanez et al. 2002; Hofstötter et al. 2002). Hence, these models generalize the hypothesis of DAC towards the neuronal substrate and can account for properties of performance in terms of the underlying neuronal mechanisms. Important here is that temporal properties of behavior are not redescribed in functional terms, which is an under-constrained problem, but directly interpreted in terms of neuronal mechanisms. This illustrates that the benchmarks cannot be interpreted as independent constraints.

Adaptive behavior ("best"). The DAC architecture has been designed in the context of real-world embodied cognition (see also *flexible behavior*). The claim is that only such an approach can account for this constraint. ACT-R is not embodied.

Vast knowledge base (mixed). DAC shows how task-dependent knowledge can be acquired and used to organize behavior and has been applied to a range of tasks (see *flexible behavior*). However, the full neuronal implementation of its structures for short- and long-term memory is not mature enough to make strong statements on its capacity and flexibility (Voegtlin & Verschure 1999). Hence, DAC takes satisfying neuronal constraints as a fundamental benchmark in answering functional challenges. ACT-R seems to stop at a functional interpretation.

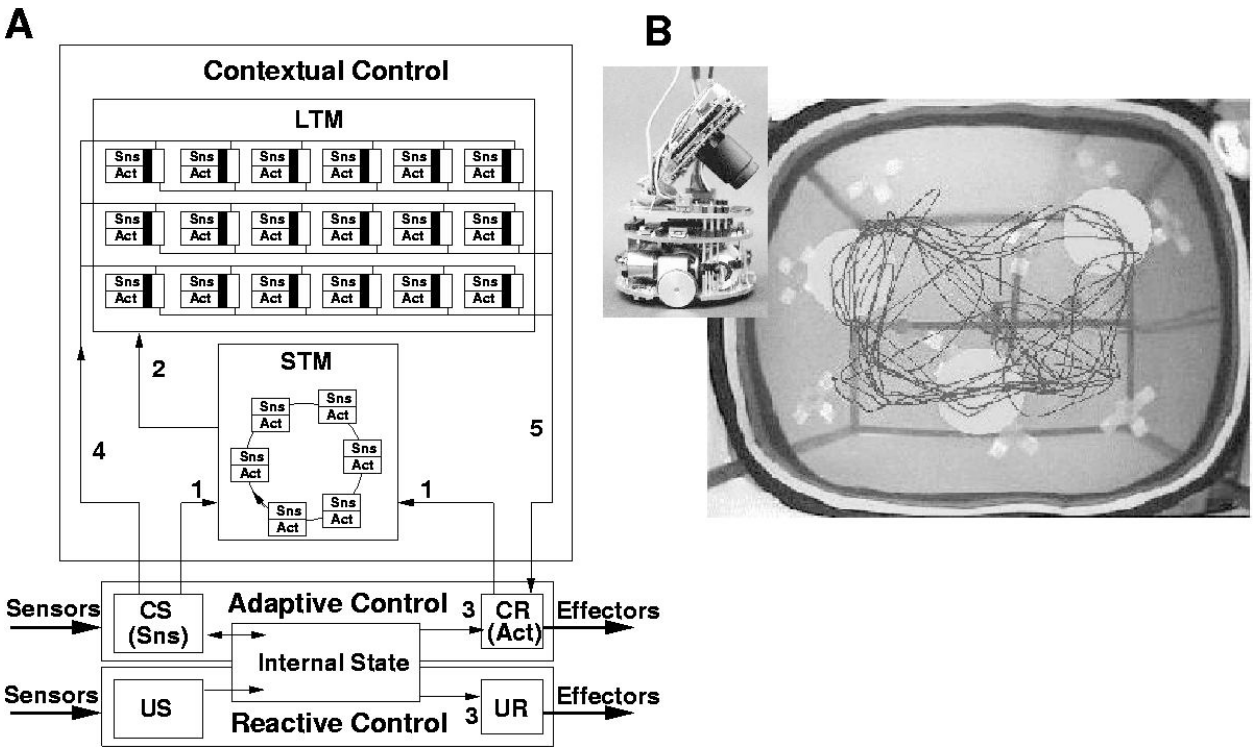


Figure 1 (Verschure). A. The DAC architecture. **B.** One example of the application of DAC to robot random foraging using a Khepera micro-robot (K-team, Lausanne).

Dynamic behavior ("best"). DAC has been applied to real-world tasks that include goal conflicts, changing motivational states, and dynamically changing environments, for example, the large-scale exhibition Ada (see *flexible behavior*). In contrast, ACT-R has only been tested on closed problem domains and has not considered the motivational components underlying the organization of dynamic behavior.

Knowledge integration ("better"). DAC has been shown to both acquire the content of its memory structures and to form goal-related recombinations of these representations. Given its Bayesian equivalence, DAC satisfies properties of inference making and induction. However, what is required is a more explicit specification of the experimental data that should be accounted for.

Natural language ("worse"). DAC has not been applied to any form of language acquisition or expression. However, DAC claims that its general learning properties will generalize to language; that is, an explanation of language should emerge from the general principles that underlie the organization of adaptive behavior and not require yet another a priori functional module. In contrast, ACT-R appears to develop in terms of a collection of functionally distinct and independent modules.

Consciousness ("worse"). For now, there is no ambition in the DAC project to attack this phenomenon.

Learning ("best"). DAC was initially conceived to address the behavioral paradigms of classical and operant conditioning. These forms of learning, as opposed to the ones the authors focus on, deal with the problem of autonomous acquisition and expression of knowledge. The biologically detailed models derived from DAC, described above, for instance, account for the phenomenon of blocking central to the Rescorla-Wagner rule of classical conditioning in terms of neuronal mechanisms and not only in functional terms (Hofstötter et al. 2002). This again emphasizes that functional and structural constraints must be satisfied simultaneously and that constraints should be defined around general models, such as the Rescorla-Wagner laws. Moreover, this approach illustrates that a theory of a cognitive architecture will probably be accompanied with a large set of specific derived models that validate a specific subset of its assumptions.

Development ("better"). The DAC architecture interprets development as the progressive involvement of its adaptive and contextual control layers. We have shown that this progression can display stage transitions characteristic for cognitive development (Verschure & Voegtlin 1998). However, the authors should be more precise in specifying what the exact datasets are that should be explained to satisfy this benchmark.

Evolution ("mixed"). Following classic examples of, for example, Pavlov (1928), DAC assumes that cognition arises out of a multilayered architecture that requires a minimum of prior specification. Because the phenomenon of classical conditioning has also been observed in insects (Menzel & Muller 1996), we are currently investigating whether the DAC principles do generalize to insects. Hence, although the results are not in, the claim is that phylogenetic continuity of principles underlying cognition should be evaluated following this comparative approach.

Brain ("better"). As mentioned earlier, the basic principles underlying the adaptive and reactive layers of DAC have been implemented and tested using biophysically and anatomically constrained models. Although the contextual layer makes predictions about the functional properties of neuronal organization, in particular, in relation to the hippocampus, basal ganglia, and prefrontal cortex, these predictions still need to be verified by developing biologically constrained models of these structures. ACT-R seems to stop at finding a correlation between neuronal responses obtained with fMRI measurements and its functional decomposition of cognition. This might not be sufficient. A&L should be congratulated for proposing a common test for theories of cognition and exposing ACT-R to it. The Newell Test in its current form, however, is not mature enough to use it as a gold standard for theories of cognition. This step should be taken in order to advance our understanding of mind, brain, and behavior.

In Figure 1, panel A, the reactive control layer provides a be-

having system with a prewired repertoire of reflexes (unconditioned stimuli and responses – US, UR) that enable it to interact with its environment and accomplish simple automatic behaviors. The activation of any reflex, however, also provides cues for learning that are used by the adaptive control layer via representations of internal states. Adaptive control provides the mechanisms for the adaptive classification of sensory events (conditioned stimulus – CS) and the reshaping of responses (conditioned responses – CR) supporting simple tasks, and can be seen as a model of classical conditioning. The sensory and motor representations formed at the level of adaptive control provide the inputs to the contextual control layer that acquires, retains, and expresses sequential representations using systems for short- and long-term memory. The contextual layer describes goal-oriented learning as observed in operant conditioning. Central-processing steps at this level in the architecture are the following: (1) The representations of sensory cues (Sns) and associated motor states (Act) acquired by the adaptive layer are stored in short-term memory (STM) as a segment. (2) If a goal state is reached, that is, a target found or a collision suffered, the contents of STM are retained in long-term memory (LTM) as a sequence. Each segment of LTM consists of a sensori-motor representation (Sns, Act) a trigger unit (black) and a collector unit (white). (3) The reactive and adaptive control layers can still trigger actions and stand in a competitive relation to the contextual control system. (4) Each Sns state of the adaptive layer is matched against those stored in LTM. (5) The collector units of LTM can trigger actions dependent on the biased competition between LTM segments. By modulating dynamic thresholds of each LTM segment, different chaining rules can be implemented.

In panel B of Figure 1, the robot learns to use the color information in the environment, the patches on the floor and the walls, in order to acquire the shortest route between goal locations, that is, light sources (grey circles). The trajectory visualized is generated during a recall task where the light sources are switched off, after learning for about 30 min. The environment measures about 1.5 by 0.8 m; and the robot, about 55 by 30 mm.

A multilevel approach to modeling human cognition

Hongbin Wang,[1] Todd R. Johnson, and Jiajie Zhang

School of Health Information Sciences, University of Texas Health Science Center at Houston, Houston, TX 77030. **hongbin.wang@uth.tmc.edu**
todd.r.johnson@uth.tmc.edu **jiajie.zhang@uth.tmc.edu**
http://www.shis.uth.tmc.edu

Abstract: Although we agree with Newell and Anderson & Lebiere (A&L) that a unified theory of cognition is needed to advance cognitive science, we disagree on how to achieve it. A hybrid system can score high in the Newell Test but may not offer a veridical and coherent theory of cognition. A multilevel approach, involving theories at both psychological and brain levels, is suggested.

Newell certainly had a very good reason for being frustrated over the progress towards a scientific understanding of the human mind. The human mind is undoubtedly one of most complex entities in the world. It is systematically shaped by genetic and evolutionary forces; fundamentally constrained by physical and biochemical laws; influenced by cultural, social, and environmental factors; and manifests itself both psychologically and neurophysiologically. Given its inherent complexity and our limited knowledge in each of these aspects, it is conceivable that we may not be able to achieve a thorough understanding of the mind's work for a long time.

While we share Newell's frustration, we doubt that the Newell Test, as proposed in the target article, would offer us relief. On the one hand, the attainability of the test is theoretically questionable. It remains controversial, for example, whether self-awareness and consciousness are computationally implementable (e.g., Penrose 1989; 1996; 1997). This controversy helps to explain why both connectionism and ACT-R were graded "worse" on criterion 8 (self-awareness and consciousness) in the target article. On the other hand, even if we ignore the possible theoretical difficulties, we may still encounter practical problems in developing theories of mind that can pass the test, as we elaborate later.

After evaluating connectionism and ACT-R based on the Newell Test and suggesting that neither was satisfactory on all criteria, the authors Anderson & Lebiere (A&L) go on to recommend some remedies. One major remedy suggested is that we should somehow dissolve the distinctions and join the two approaches close together. Specifically, ACT-R needs to be "more compatible with connectionism," and connectionism needs to be concerned "with more complex tasks and symbolic processing" (sect. 6, para. 3). The authors note that building hybrid systems that can integrate the two approaches is particularly promising (ACT-R itself is already a form of hybrid system). By combining the advantages of different sub-approaches, the authors seem to suggest that hybrid systems would bring us one step closer to a Theory of Mind (ToM) that can score high in the Newell Test.

Unfortunately, there are at least three problems with this hybrid system approach. First, it should be noted that there are two (out of 12) criteria on which both connectionism and ACT-R score worse or worst. They are criterion 8 (self-awareness and consciousness) and criterion 11 (evolution). The simultaneous failure of both approaches on both criteria suggests that simply hybridizing the two approaches might not provide a solution.

Second, what if we develop a theory of self-awareness and an evolutionary ToM, and then hybridize these two theories with the hybrid system we constructed earlier? Does this give us a better ToM? Well, maybe. If doable, it will certainly boost the Newell Test score! But it also induces a paradox. Focusing on isolated and segmented subtheories of mind is what frustrated Newell and motivated the creation of the Newell criteria in the first place. If we first need to develop subtheories to develop high-scoring hybrid systems, we then lose the very point of the Newell Test.

Third, and most important, hybrid systems are artificially assembled systems and thus bear little true psychological and neurophysiological significance. Although we all agree that the human mind is a complex, multilevel construct and involves mechanisms and operations at, among others, both psychological and neuronal networks levels, simply piecing them together is ad hoc and trivializes the problem. A ToM that explains one phenomenon using a neural-network-level mechanism and explains another phenomenon using a rule-based, symbolic-level mechanism may be a convenient hybrid ToM, but is certainly not the *unified* ToM that Newell had wished for (cf. Newell 1990).

In our view, any principled ToM must recognize that the human mind may adopt different mechanisms and follow different laws at different levels. In addition, it is highly unlikely that there exists any simple and linear one-to-one mapping across levels. Penrose, for example, went so far as to hypothesize that there is a noncomputational and nonlocal process called "objective reduction" that connects physics and consciousness (see also Woolf & Hameroff 2001). We would argue that a similar nonlinear relationship exists between the neuronal-network-level and the psychological level, and that each level tells a veracious but adequately distinct story of mind. Such a multilevel view is also consistent with both Marr's (1982) and Newell's (1990) conception of multiple-level description of human cognition. Consequently, we should not expect a single architecture, even a hybrid one, to explain all of the phenomena of mind.

We regard both ACT-R and connectionism as celebratory candidates for a ToM, but at different levels. Whereas ACT-R focuses on the symbolic mental structures and processes and offers a psychologically plausible explanation that closely links to empirical behaviors, connectionism adopts subsymbolic neural-based mechanisms and permits a biologically realistic explanation

of mind that closely links to brain functions (e.g., O'Reilly & Munakata 2000). The two approaches are distinct in that symbols simply do not exist at a subsymbolic level. A unified ToM needs to encompass both levels of description, though each may be embodied in separate cognitive architectures. We regard the attempt to vertically stretch one level of analysis to linearly map to another as problematic. For example, we doubt that there is such a simple one-to-one mapping between ACT-R components and brain structures, as suggested in Figure 1 of the target article. It is hard to imagine (and not supported by neuroscience evidence) that the damage to the basal ganglia would completely destroy the work of mind given the fundamental role that production rules play in ACT-R.

In summary, although we agree with Newell and A&L that a unified ToM is needed to advance cognitive science, we have different opinions regarding how to achieve such a unified theory. Our position is that, instead of hybridizing different approaches or linearly mapping them to boost the Newell Test score, we need to recognize the multilevel nature of the human mind and develop complementary theories at both psychological and connectionist levels, and cross-validate them.

NOTE

1. Hongbin Wang is the corresponding author for this commentary.

Newell's program, like Hilbert's, is dead; let's move on

Yingrui Yang[a] and Selmer Bringsjord[b]

[a]*Department of Cognitive Science, Rensselaer Polytechnic Institute, Troy, NY 12180;* [b]*Department of Computer Science, Rensselaer Polytechnic Institute, Troy, NY 12180.* **yangyri@rpi.edu** **selmer@rpi.edu** **http://www.rpi.edu/~brings**

Abstract: We draw an analogy between Hilbert's program (HP) for mathematics and Newell's program (NP) for cognitive modeling. The analogy reveals that NP, like HP before it, is fundamentally flawed. The only alternative is a program anchored by an admission that cognition is more than computation.

As most readers will know, Hilbert's program (HP) was devoted to building a system at or below the level of Turing machines (and their equivalents) to definitively settle all mathematical questions. Most readers will also know that in 1931, a young Viennese logician, Kurt Gödel, proved two incompleteness theorems – and Hilbert's program (HP) instantly died. Out of this death was born a much more sophisticated approach: In a word, true *meta* mathematics arose. One specific innovation was to devise and study infinitary logics not bound by Gödelian incompleteness, because, from an information-processing perspective, these logics are *beyond* Turing machines (Barwise 1980). Newell's program (NP) – the attempt to build a system at or below the level of Turing machines able to satisfy the criteria that Anderson & Lebiere (A&L) distill for us – has likewise expired. The difference is that apparently many influential cognitive scientists want to pretend the funeral never happened.

A&L, and in fact all those who see themselves as advancing NP, presuppose an exceedingly convenient sense of *universality.* According to this sense, a system is "universal" if and only if it can compute all Turing-computable functions. The construal is convenient because the vast majority of functions (in the relevant classes; e.g., functions from N to N) *aren't* Turing-computable (Boolos & Jeffrey 1989). A&L, the classical connectionists they evaluate, those they count as fans (e.g., Dennett), and so forth – all *assume* that human cognition can be nicely packaged beneath the Turing Limit. Yet, after decades of work, no system at or below the Turing Limit has the conversational power of a toddler (to pick just one criterion: 7). You would think the notion Newell (and his similarly sanguine partner, Simon) so confidently preached at the dawn of cognitive science (that thinking is computing at or below the Turing Limit, and computers operating at or below this limit with human-level intelligence will soon arrive) would, like Hilbert's dream, be a carcass at this point, but yet here is a *BBS* target article still stubbornly clinging to the religion (by cheerfully acting as if everyone is a believer). What arguments support the doctrine that cognition can be captured by standard computation? Surely no cogent argument is to be found on the basis of what has been built. LOGIC THEORIST, at the 1956 Dartmouth conference that kicked off AI, was able to prove the marvelously subtle theorem that *if p then q* implies *if not-q then not-p,* and this prompted Simon to declare that thinking machines would soon be among us. The situation is no different now: Here are A&L confidently pressing on to capture cognition in simple computation – but on the strength of what impressive artifact? Since seeing is believing, you will pardon us for not believing.

The problem isn't just criterion 7. Faced with consciousness, NP irremediably fails. Yet A&L once again cook up the convenient: They construe *consciousness* (in criterion 8) so that it simply leaves out the concept that threatens Newell's enterprise: namely, *phenomenal* consciousness. Block (1995) has recently explained the issue in this very journal. ACT-R and all forms of connectionism, and indeed every approach to sustaining NP, can't even take the first step toward expressing, in a third-person scheme, what it feels like to taste deep chocolate ice cream. ACT-R will be used to at most create what one of us (Bringsjord 2000) has called "zombanimals," that is, artificial animals with no inner lives. A robot with the behavioral repertoire of a dog, but with the inner life of a rock, might well be something NP, fueled by ACT-R, can produce.

That NP, as driven specifically by ACT-R, is dead, can be seen with help from concrete, not just philosophical, challenges. ACT-R is wholly incapable of modeling beliefs of the sort that human readers have when reading murder mysteries. For example, as detailed in Bringsjord (2000), readers have n-order beliefs about villains and detectives, and they make inferences based on these beliefs. For example, the reader of a murder mystery often believes that the villain believes that the detective believes that the villain believes the villain will never be caught. You can't represent this in ACT-R, period, because ACT-R is at best part of first-order logic devoid of doxastic operators. Of course, one could hook up a robust theory of human and machine reasoning (e.g., see Yang & Bringsjord, forthcoming) to ACT-R, but then in what sense *is* that new composite system ACT-R? If ACT-R is to be genuine science, it must be falsifiable. Yet A&L seem to describe an evolution in which serious challenges are handled by simply augmenting the system.

Just as the death of HP gave birth to infinitary logic, so should the death of NP give rise to cognitive modeling untrammeled by standard computation. Cognitive modelers need to step outside the notion that mere computation will suffice. They must face up to the fact, first, that the human mind encompasses not just the ordinary, humble computation that Newell and all his followers can't see beyond, but also *hyper*computation: information processing at a level *above* Turing machines, a level that can be formalized with help from analog chaotic neural nets, trial-and-error machines, Zeus machines, and the like (Bringsjord & Zenzen 2003; Siegelmann 1999).

In his famous "twenty questions" paper, Newell (1973) tells us that a sound science of the mind should not be steered by the willy-nilly dictates of experiment-driven empiricism. Instead, we are to do computational cognitive modeling. But such modeling, if limited by NP, fails to let cold hard reality lead the way. Instead, it lets speculative assumptions (e.g., that the mind processes information at or below the Turing Limit) prevent nature from proclaiming that we are more than ordinary machines.

Cognitive architectures need compliancy, not universality

Richard M. Young

Psychology Department, University of Hertfordshire, Hatfield, Herts AL10 9AB, United Kingdom. **r.m.young@herts.ac.uk**
http://www.psy.herts.ac.uk/pub/r.m.young/

Abstract: The criterion of computational universality for an architecture should be replaced by the notion of compliancy, where a model built within an architecture is compliant to the extent that the model allows the architecture to determine the processing. The test should be that the architecture *does easily* – that is, enables a compliant model to do – what people do easily.

Anderson & Lebiere (A&L) are to be congratulated on advancing the agenda of assessing cognitive architectures (or other cognitive theories of broad scope) as a whole. The inspiration is clearly Newell's, but the authors take a major step towards bringing Newell's criteria down to earth by operationalising them and bringing them closer to objective criteria and tests. This present commentary is offered as a minor contribution to that same goal.

In section 2.1, A&L follow Newell in identifying the criterion of *flexible behavior* with computational universality, that is, equivalence to a Turing machine. But Turing computability is inappropriate as a criterion for cognitive architectures. It is by nature an all-or-nothing test: Can, or cannot, the architecture be programmed to compute any Turing-computable function, yes or no? The authors certainly do themselves no favours by adopting Turing universality as the touchstone for flexible behaviour. Indeed, it forces them into a contradiction. Although in section 4.5 they deny that "ACT-R is a general computational system than can be programmed to do anything," that is indeed what being Turing universal means, that the architecture can be "programmed to do anything." What is needed instead is a graded measure, reflecting the reality that, as A&L acknowledge, "some things are much easier for people to learn and do than others." Ideally, the architecture should learn and do easily the things that people learn and do easily and, similarly, find the same things difficult.

Of course, what is meant by an architecture doing or learning something *easily* itself needs careful definition and explication. It is no trivial matter to replace the all-or-nothing concept of Turing computability by a more appropriate measure that both captures and makes precise these important but rather vague ideas about "doing something easily" or doing it by means "in keeping with the spirit of an architecture." However, a start has been made, with the concept of the *compliancy* of models constructed within a cognitive architecture. The idea has been worked through most thoroughly for SOAR, but is applicable in principle to any cognitive architecture.

In Howes and Young (1997), we approach the issue by considering how in practice architectures are used by cognitive modellers, and how credit and blame for the resulting models get assigned in the light of agreement with empirical data (or other evaluative criteria). We note how, in applying an architecture to a particular domain or task, the modeller inherits all the theoretical commitments of the architecture and then adds further commitments, often expressed in the form of production rules, which are specific to the domain or task being modelled. We refer to these additions as *model increments*, by analogy with the *method increments* which Laird (1986) identifies as giving rise to the "weak methods" of problem solving. We are led thereby to pose a methodological question: Given that a model (of this kind) consists of a cognitive architecture together with a specific model increment, in cases where the model does well, that is, provides a good prediction and explanation of the data, where does the credit belong: to the architecture, to the model increment, or somehow to both? And likewise if the model does badly, where lies the blame?

We note too that the extent to which cognitive architectures constrain and shape the models constructed within them, and thereby contribute to their predictions, is not widely recognised by those without first-hand experience of such architectures. Building model increments is not at all like writing programs in a theoretically neutral programming language. An architecture is not simply a programming environment for constructing cognitive models according to the modeller's fancy. Indeed, some architectures, of which SOAR (Newell 1990) is an example, are themselves capable of generating behaviour once they are given a specification of the task to be performed, even without further information about *how* it is to be performed. In such cases, the role of the model increment becomes not so much to generate behaviour, as to modulate or modify the behaviour which the architecture is already advocating.

That observation leads us to introduce the idea of compliancy. A model increment is compliant to the extent that it follows the architecture's lead, that is, takes advantage of the architecture's own tendency, allowing it mostly to do what it wants, intervening just occasionally to keep it on track. A model increment with low compliance, by contrast, disregards or overrules the architecture's own tendencies and simply forces the architecture to do what the model increment wants. (If the architecture is indeed Turing universal, then a model increment can always be written to produce any specified behaviour, but the increment may have to fight against the architecture in order to achieve that behaviour.)

The notion of compliancy allows us to answer the question about credit assignment. To the extent that the model increment is compliant with the architecture, much of the credit or blame attaches to the architecture itself. But to the extent that the model increment is noncompliant, responsibility for the resulting behaviour, whether good or ill, rests mostly with the model increment.

My suggestion is that compliancy also offers promise as a route for explicating what it means for an architecture to perform a task easily or with difficulty. An architecture can be said to find a task easy if a compliant model increment suffices to build a model to do it. Contrariwise, the architecture finds a task difficult if a noncompliant model increment is required, which therefore has to "force" the architecture to do it in a way "not in keeping with its spirit." By utilising compliancy, Newell's criterion of flexible behaviour can be interpreted as a requirement that the architecture does or learns easily (in other words, enables a compliant model to do or learn) what people find easy to do or learn, and finds difficult (in other words, requires a noncompliant model to do or learn) what people find difficult.

Authors' Response

Optimism for the future of unified theories

John R. Anderson and Christian Lebiere

Department of Psychology, Carnegie Mellon University, Pittsburgh, Pa 15213.
ja@cmu.edu **cl@andrew.cmu.edu**

Abstract: The commentaries on our article encourage us to believe that researchers are beginning to take seriously the goal of achieving the broad adequacy that Newell aspired to. The commentators offer useful elaborations to the criteria we suggested for the Newell Test. We agree with many of the commentators that classical connectionism is too restrictive to achieve this broad adequacy, and that other connectionist approaches are not so limited and can deal with the symbolic components of thought. All these approaches, including ACT-R, need to accept the idea that progress in science is a matter of better approximating these goals, and it is premature to be making judgments of true or false.

We begin by noting how pleased we were with the commentaries. Most commentators found something to disagree with, but these disagreements were by and large constructive and advanced the goals of defining criteria by which cognitive theories should be evaluated and using these criteria to evaluate many theories. In reading the commentaries and devising our responses we both increased our appreciation of alternative theories and refined our goals in pursuing ACT-R. We also increased our appreciation of the current state of theory development. The space of cognitive theories is indeed much more complex than our use of only two candidates could have suggested, with theories sharing some features and mechanisms while differing on others. As Herb Simon was advocating, one needs to go beyond evaluating theories as brands and consider them as a collection of mechanisms and evaluate them as such.

R1. Correcting the misconceptions

Before addressing specific points in the commentaries, we will correct a pair of misconceptions that were found with varying degrees of explicitness in some of the commentaries, reflecting a natural misreading of the paper. We were *not* using the criteria on the Newell Test as a basis for comparing classical connectionism and ACT-R, and we were *not* proposing them as a way to judge whether a theory should be deemed a success or a failure. We were not grading connectionism relative to ACT-R because it would not be credible for us to serve as either judge (in specifying the tests) or jury (in deciding which was best) in a contest between our theory and another one. However, it is perfectly reasonable for others to take these criteria and make judgments about the relative merits of the theories, as indeed some of the commentators have done.

Although it is fine for others to use these criteria for comparing theories, it is at least premature to be in the business of judging any theory an overall success or failure. All theories need a lot more development, and the point of such a set of criteria is to identify places where more work is needed. Therefore, we used a zero-sum grading scheme that forces one to identify where a theory needs the most work. Such a grading scheme forces a humbleness and self-criticism that the field could use.

With respect to the issue of falsification, a number of commentators (e.g., **Agassi**, **Taatgen**) speak with some fondness about the Turing Test in that it provides a criterion for rejecting theories. We too have some fondness for the Turing Test and frequently apply it to ACT-R simulations, not to provide an ultimate test of the theory but to force ourselves to see where ACT-R needs development. To try to repeat one of Herb Simon's frequent rejoinders as exactly as we can remember it: "If you just want to know whether the theory is wrong, then we can go home now. What I want to find out is how it is wrong and how it can be improved." The reason we formulated the Newell Test when the Turing Test was already available is because we wanted to provide some structure in this search for improvement.

The commentary by **Yang & Bringsjord** is surely the strongest in arguing for a yes-no judgment on theories. They argue that the whole class of computational theories, including ACT-R and classical connectionism, is dead. Their choice of the word "dead" rather than "false" gives away a lot. Unlike Gödel, whom they hold up as the ideal, Yang & Bringsjord provide nothing approaching a proof in their claims. As they should know from that important moment in the history of thought, the standards for making such sweeping negative pronouncements should be high. Gödel is such an important figure because he achieved those standards in his proofs.

We would like to particularly commend **Gray, Schoelles, and Myers** (**Gray et al.**) for bringing attention to cognitive engineering as a factor to shape these criteria. As they note, Newell thought cognitive engineering was an extremely important criterion for evaluating theories and much more than "just an application." Cognitive engineering gets at extremely profound issues about the nature of our science and the richly textured considerations that have to be brought to bear in evaluating cognitive theories and why simple yes-no, true-false judgments are typically inappropriate. This is a matter that deserves an elaborate community discussion. Such a discussion would reveal that the individual Newell tests are just the tips of a great iceberg.

R2. Developing the criteria

Agassi is correct that it is not always clear how to fully evaluate some of the criteria. In such cases the criteria should be stimuli for further thought and investigation so that they can be more fully applied. Indeed, many of the commentators have already proposed improvements and elaborations to the criteria. We particularly want to recommend the elaborations offered by **Taatgen**.

Gelepithis does a service in raising the issue of the exact relationship between the criteria we propose and those in Newell. As we think he creates too negative an impression of our scholarship, we will add some elaborations on this point. Ten of our criteria are verbatim from Newell (1980) and in the same order. We discuss at length in the target article the need to reformulate Newell's criterion 6 (symbols) as our criterion 6 (knowledge integration). Our criterion 12 ("be realizable within the brain") merges his criteria 12 ("be realizable within the brain as a physical system") and 13 ("be realizable as a physical system") because his distinction is not important to our paper nor is it a distinction that survived in his 1990 list. It is true that our list bears a less exact relationship to the 1990 list but at just three points: As can be seen from Gelepithis's Table 2, Newell in 1990 merged vast knowledge and robust behavior (criteria 4 and 5 in our table and in his 1980 table) into a single criterion (number 4 in the 1990 list), broke the developmental criterion (number 10 in our Table 1 and his list) into two criteria (8 and 12 in the 1990 list), and introduced a new criterion (social).

Criterion 4 in Newell's 1990 list covers our criteria 4 and 5 plus more. It is close to the embodiment criterion that **Spurrett** advocates, and Newell's reasons for reorganizing his list here may have been close to the arguments given by Spurrett. We think Spurrett's judgment of the relative performance of ACT-R versus connectionism on the criterion of embodiment is rather ungenerous. As **Gray et al.** note, ACT-R does well in addressing a range of HCI issues where connectionism has been almost totally silent. Nonetheless, robots are compelling demonstrations and hopefully someone in the ACT-R community will take up robots to satisfy Spurrett (and, we are sure, others).

Something may have been lost in collapsing Newell's

1990 developmental and embryological growth criteria into just the developmental criteria. **Sirois** offers a somewhat similar distinction between maturation, which he sees as a functional constraint, and development, which he sees as a functional ability to be explained.

Gelepithis offers a number of additional potential criteria from his 1999 paper. We agree that his suggestion of emotion and Newell's 1990 social behavior are two well-justified criteria. We made no claim to be exhaustive in choosing Newell's original 12. Our goal in working with those 12 was to have a manageable number for a *BBS* target article and to have criteria that came from some authoritative external source (to avoid the circularity that **Overgaard** & **Willert** mention).

As noted in the target article, the big missing criterion was accuracy of empirical predictions – having one's theory correspond to the details of empirical data. The criterion was missing only because it was not on Newell's lists, and it was not on Newell's lists only because he was talking about functionality at the points he introduced the lists, not because he did not strongly believe in its importance. Having a list that excludes predictive accuracy serves as something of a counterweight to the predominant tendency to consider only empirical adequacy and thus produce theories that are narrowly accurate in their predictions but incapable of being integrated into a complete functional theory of human cognition. However, in any final list that will serve as a "gold standard" (**Verschure**) the accuracy of empirical predictions needs to be given first place. It might make sense to give empirical adequacy half the weight in evaluating a theory and give the other half of the weight to functional criteria like those in Newell's list. As **Altmann** notes, such functional criteria can be crucial in deciding among theoretical accounts of particular phenomena that are hard to distinguish on the basis of their predictions. Functional criteria force the theories to consider difficult real-world problems rather than split hairs on tiny tasks that might not provide stringent enough tests to differentiate theories. One lesson that could be learned from the history of AI is the danger of focusing on abstract toy problems and the benefits of tackling the hard real-world problems.

One of the criteria that created some distress among several commentators (e.g., **Pyysiäinen**, **Young**), and for the reasons anticipated in the target article, is our attempt to operationalize flexible behavior as universality. Young has produced a superior version of this criterion in terms on what he calls "compliancy." It includes the test of universality as a component but connects differential difficulty of the models with the characteristics of the architecture. His development falls short of an explicit definition of what it means for one model to be more compliant than another. However, as is the case with other Newell criteria, that is a stimulus for further thought. Even in its current form it is better than the answer we could have composed to respond to **Tadepalli**'s worries about the range of models one can develop in an architecture.

Some commentators (**Wang et al.**; **Yang & Bringsjord**, **Overgaard & Willert**; **Sirois**) wonder whether it is possible to satisfy the consciousness constraint within any such framework. As both Overgaard & Willert and Yang & Bringsjord note, the answer to this question depends on what one takes to be consciousness. If we take it to be those aspects of consciousness that are amenable to scientific investigation, then we think the answer is yes. That may not include Block's (1995) phenomenal consciousness under some construals.

Wang asserts it is not possible to achieve all 12 criteria at the same level of explanation. For instance, he contends that ACT-R is too high-level to map onto brain structure. We disagree and offer the papers by Anderson et al. (2003), Qin et al. (2003), and Sohn et al. (2003) as emerging counterevidence. It is precisely because ACT-R is targeted at the architectural level of cognition that it is relevant to explaining the type of data generated by experimental neuroscience techniques such as fMRI. We think the mappings we proposed in Figure 1 of the target article have a lot of merit, but we agree with Wang that the connections displayed are not complete and that neuroscience evidence indicates that there are direct connections between some modules that do not go through the basal ganglia. Rather than be discouraged by this shortcoming, in the spirit of the Newell Test we take it as stimulus for further theoretical work.

Despite the fact that the list contains the two overlapping criteria of learning and development, a number of the commentators (**Commons & White**, **Prudkov**, **Roy**, and **Verschure**) argue that we did not give enough credit to self-organization. What they want is more emphasis on having a system that really constructed itself from experience without the guiding hand of the theorist. Though advocates of this criterion may not be giving adequate credit to what the system brings to the task as part of its genetic endowment, it is one of the holy grails of functionality. Perhaps it should be raised to a more prominent position. In addition to satisfying the subliminal "mad scientist" desire to see a being grow *de novo* in a computer program, achieving this serves an important role in constraining the degrees of freedom in proposing models within an architecture. Roy and Verschure are quite correct in noting that classical connectionism does not achieve this criterion even in its learning simulations, but we think this criterion is the dimension on which ACT-R suffers most in comparison to classical connectionism. As Prudkov notes, more has to be specified in typical ACT-R models before ACT-R learning can take over, than needs to be specified in connectionist models before connectionist learning can take over. We think this is because ACT-R models address more complex cognition, but the consequence is that it is more difficult to teach ACT-R aspirants what they need to know to become competent ACT-R modelers. One of our major goals in the future development of ACT-R is to move closer to achieving this holy grail.

Clancey challenges us to account for dysfunctional behavior as well as the functional. Of course, one cannot really have a theory of what is dysfunctional without first defining and accounting for functionality. This may not be another criterion to add to the list; rather it seems a different emphasis in evaluating the criteria that Newell has already given. However, we certainly agree with the importance of accounting for dysfunctionalities. Accounting for the full range of functional and dysfunctional behavior would also constitute a response by cognitive modeling to those who suggest that it is merely a parameters tuning game (since specific parameter values may map onto specific dysfunctionalities).

R3. Theories to which the criteria can be applied

An issue in applying the Newell criteria to classical connectionism or ACT-R is the degree to which these are re-

ally theories that can be so evaluated. **O'Loughlin & Karmiloff-Smith** argue that connectionism is a collection of tools that are useful. A similar point is often raised about ACT-R (e.g., by **Tadepalli**), and often elaborated in discussions of the distinctions between models and the architecture. We are certainly aware that connectionism in its largest sense is too broad for such an evaluation but we tried to focus on what we chose to call classical connectionism. **McClelland et al.** believe they have something to be evaluated, although they prefer to call it a framework in contrast to ACT-R, which they correctly call "an architecture." Nonetheless, they do regard themselves as having a "theoretical commitment to a common set of principles" that can serve as a basis for evaluation.

It is true that from among these theories one can define many models for performing tasks and that different models may differ in their predictions. However, it is just because of this fact that one needs to take the broader perspective of the overall functionality of the architecture. In part, this is so one can judge which models are in the spirit of the architecture, or "compliant" in **Young**'s term.

Many commentators (**Commons & White**, **Garzón**, **Gelepithis**, **Grossberg**, **Sun**, and **Verschure**) think that we unnecessarily restricted the kinds of neural networks considered by focusing on classical connectionism. Grossberg refers to classical connectionism as "Carnegie Mellon connectionism," implying that we were provincial in our outlook. Sun reminds us that we wrote that classical connectionism reflects "the core and the bulk" of existing neural network models (cf. target article, last para. of sect. 3). We clearly misspoke when we said "bulk" but we think we can still defend the claim that it is "the core" and not just a reflection of our provincialism. However, such a defense would be a digression here and we will just note our point of agreement with these commentators: They believe that classical connectionism is too restrictive and suffers weaknesses that more liberal uses of neural net ideas do not suffer. In particular, other forms of neural networks need have no problem with symbols. We agree and indeed view ACT-R as just a higher-level description of such a nonclassical connectionist theory. But there is a trade-off between assuming an overly broad definition of a framework that can account for anything (and its opposite) and an overly narrow one that leaves out many related efforts. We tried to strike the best balance possible in our definition of classical connectionism, expressing a common set of principles that are significantly constraining but broad enough to encompass a substantial part of connectionist efforts.

One of the things that encouraged us most was that some of commentators (**Clancey**, **Garzón**, **Grossberg**, **Verschure**) took many or all of the Newell criteria seriously and evaluated their theories on the basis of these criteria. Reading their short descriptions helped us appreciate those theories and caused us to read some of the sources they cited. Having done so, we do not want to take issue with their self-evaluations, and we hope the exercise helped them to see how they could improve their architectures.

R4. Past-tense issues

The target article held up the Taatgen and Anderson past-tense model as a paradigm of what could be accomplished in current ACT-R (cf. sect. 4.4; Taatgen & Anderson 2002), and the claims of that model came in for some analysis. One of the reasons for highlighting this model is that it depends so much on ACT-R learning mechanisms and so little on the initial structuring of the problem. As such it comes closest to achieving the *de novo* test that others want. Still, **Tadepalli** wonders to what degree its behavior reflects characteristics of the problem rather than ACT-R. This is an important question that needs to be asked more often. However, we do list things this model achieves that most other models facing the same problem do not achieve.

A number of commentators correctly point out shortcomings of the current model. **Ter Meulen** points out the inadequate conception of the semantics of past tense and failure to embed the model in a system that generates full utterances. **McClelland et al.** point out the impoverished conception of phonology, which limits the ability to extend the model because it relies on measures of phonological cost. One of the virtues of taking the Newell Test seriously is that one cannot just circle the wagons in response to criticisms like these and say that they are beyond the scope of the model. These are valid criticisms and point to directions for future work. Indeed, some steps have already been taken to enrich the treatment of the phonology (Misker & Anderson 2003; Taatgen & Dijkstra 2003). Taatgen and Dijkstra show how the approach can be used to produce "irregular generalizations like bring-brang." The Misker and Anderson analysis shows how complex phonological constraints like those in optimality theory (Prince & Smolensky 1993) can be represented and computed within ACT-R. Although it has not yet been done, we believe that if the Taatgen and Anderson (2002) learning approach were embedded on top of the Misker and Anderson approach, we would be able to account for such things as the distributional evidence that McClelland et al. cite with respect to the phonological characteristics of past tense exceptions.

R5. McClelland, Plaut, Gotts, and Maia (McClelland et al.)

We tried to define classical connectionism somewhat more broadly, but it is worthwhile to follow the lead of **McClelland et al.** and consider parallel distributed processing (PDP) specifically. The similarities between the broad goals of ACT-R and PDP and between some of their mechanisms can appear quite striking. From the perspective of a commentary like that of **Yang & Bringsjord**, our disagreements might seem like disputes between Baptists and Methodists. Aspects of ACT-R have been strongly influenced by connectionist ideas (frequently specifically PDP ideas) as described in the target article. Indeed, we think one of the major reasons for the success of the ACT-R effort is our willingness to incorporate good ideas – whether they come from EPIC (Meyer & Kieras 1997) or PDP.

The **McClelland et al.** commentary brings out three issues between ACT-R and PDP that need discussion. One has to do with the word "approximate," the second with the word "unified," and the third with the word "symbolic."

With respect to the word "approximate" one cannot help but read the commentary as using it a little bit as a dirty word (presumably in contrast to a good word like "exact"). In fact, to avoid any sense of not being collegial in their commentary, **McClelland et al.** hasten to note that they do not mean to suggest that we advocate approximation al-

though they wonder if Newell would. We cannot find where he said it in print, but one of the remarks we remember from our interactions with Newell is his assertion that the development of scientific theories is like an "approximating sequence." We agree with Newell on this score. Presumably no one can lay claim to having the true theory. **McClelland et al.** describe the symbolic level as "sometimes useful as high-level approximations to the underlying mechanisms of thought" (see their commentary Abstract). However, surely the units in a PDP model are only approximations to any neural processing which can at most claim to be useful as well. Their own recounting of the history of the development of their ideas is surely well described as an approximating sequence.

If one acknowledges that one's theory is an approximation that one is trying to make closer to the truth, then it becomes a strategic decision where one wants to work on improving the approximation. **McClelland et al.** advocate sticking within a well-circumscribed domain and working at getting their account closer and closer. Certainly we have done this, trying for more than 25 years (Anderson 1974; Anderson & Reder 1999b) to get an account of associative interference or the fan effect correct because we view this as central to the ACT theory. However, we do agree that we have put more attention in getting the approximations to work reasonably well across domains. This is even true in our work on the fan effect where we have tried to study it over a wide range of tasks. It is a strategic decision whether to try get some things really well, narrowly, and then go on to other topics, or whether to try to get a broad range of topics relatively well and then seek better approximations everywhere. The jury is surely still out on which is the better strategy. If the field of operations research offers any lesson in this regard, it is that the number and distribution of points that one is trying to fit is a stronger constraint than how closely they are fitted.

The second word, "unified," comes from the title of Newell's book, but thinking about it helps us understand the differences and similarities between the ACT-R and the PDP research strategies. Unified can mean two things: (1) that the theory tries to explain everything from the same few basic principles and (2) that the theory tries to explain how the broad range of intellectual functions is achieved in a single brain. We will refer to the first sense as "unitary" and the second sense as "integrated." Theoretical efforts can be cross-classified as to where they stand on these two dimensions. As **McClelland et al.** note, most theoretical accounts are neither unitary nor integrated, and PDP efforts share with Newell's SOAR and the ACT-R effort the aspiration to achieve more. However, it turns out that ACT-R, SOAR, and PDP each occupy a different cell of the remaining three in the two-by-two cross-classification. PDP shares with SOAR and differs from ACT-R in the desire to find a unitary theory – a small set of domain-general principles. ACT-R's predecessor, ACT* (Anderson 1983), did aspire to the same sort of unitary theory as SOAR and PDP. However, in response to the need to make progress on the Newell criteria we found this constraint to be an obstacle. Also, our understanding of biology "inspires" us to take the modular view described in Figure 1 of the target article. Imagine trying to account for respiration and digestion from a unitary set of principles! We see no more reason in our understanding of the brain to have that attitude about, for example, audition and manual control. (However, when possible we do try to exploit common principles in accounting for different modules – for we too like generalizations that work.)

On the other hand, we share with Newell and differ from the described PDP goals in having the aspiration to produce an integrated theory that explains how diverse and complex behaviors arise from one brain that has one set of mechanisms. This naturally leads to a focus on more complex behaviors such as mathematical problem solving or driving. We suspect we are more sympathetic to **ter Meulen**'s argument that the past tense model should be extended to deal with more complex constructions. If one believes that it is the same few principles working out the same way in domain after domain, then it makes sense to look at relatively simple tasks and model them intensely. If one believes that it is many modules interacting to produce complex adaptations, then it makes sense to look at a number of complex tasks.

Of course, there is the danger of becoming a jack of many trades and a master of none. This is why Anderson in his work on tutoring (Anderson et al. 1995; Koedinger et al. 1997) has focused almost exclusively on mathematical problem solving (and of a high school variety at that) because one has to understand that domain deeply. Newell (1973) himself saw the need to focus in depth on topics like chess to properly treat their richness. Fortunately, others in the ACT-R community have taken up other topics such as driving (Salvucci 2001) or past tense. Therefore, we certainly respect the decision of PDP researchers to focus on certain domains such as reading of words. One enviable feature of connectionism is the number of researchers who have taken up applying it to different domains. However, our bet is that the lack of concern with integration will lead to systems that cannot be put together – all the king's horses and all the king's men won't be able to put Humpty Dumpty together.

Finally, there is the word "symbolic." We changed Newell's criterion 6 to avoid the use of that word because it seemed too hopelessly loaded to ever serve as a useful criterion (and because his specification of this criterion really did not fit the functional character of the other criteria). Despite the frequency with which "symbolic" is used in Cognitive Science it seems to be more often a hindrance to communication than a help. A case in point is our claims about **McClelland et al.**'s attitude toward the symbolic level. McClelland et al. deny that the symbolic level is "the appropriate level at which principles of processing and learning should be formulated." That is what we meant when we said they did not "acknowledge a symbolic level to thought" (target article, Abstract), but apparently for them treating the symbolic level as sometimes a "fairly good approximation" amounts to acknowledging it. We did understand this about the PDP account (e.g., see Anderson 1990, pp 11–14). So we agree on what the PDP position does and does not say about the symbolic level, even if we cannot agree on the words to describe it.

For better or worse, we cannot entirely abandon using the word "symbolic" because we have long since committed to describing certain of ACT-R's principles as being at the symbolic level and others being at the subsymbolic level. Presumably, **McClelland et al.** would deny the appropriateness of the ACT-R principles that we describe as being at the symbolic level. We and others believe that it is this failure to incorporate such principles that produces the

limitations in their accounts. As we described in the target article, ACT-R's symbolic account cashes out at a connectionist level as prior constraints on the communication among the modules. Although McClelland et al. may not want to acknowledge such constraints, other connectionists have done so in terms of things like architectural constraints (Elman et al. 1996).

R6. Conclusion

Ours was a different target article than Newell (1992) and so naturally provoked a different set of commentaries. Still, we think that if he were to compare the commentaries in 2003 with those in 1992 he would see growth in the attitudes in Cognitive Science, maturation in the theories, and hope for the future.

References

Letters "a" and "r" appearing before authors' initials refer to target article and response respectively.

Ackley, D. H., Hinton, G. E. & Sejnowsky, T. J. (1985) A learning algorithm for Boltzmann machines. *Cognitive Science* 9:147–69. [aJRA]

Agassi, J. (1987) The wisdom of the eye. *Journal of Social and Biological Structures* 10:408–13. [JA]

(1988/2003) Winter 1988 Daedalus. *SIGArt Newsletter* 105:15–22; reprinted in Agassi (2003). [JA]

(1992) Heuristic computer-assisted, not computerized: Comments on Simon's project. *Journal of Epistemological and Social Studies on Science and Technology* 6:15–18. [JA]

(2003) *Science and culture. Boston Studies in the Philosophy of Science, vol. 231.* [JA]

Agassi, J. & Laor, N. (2000) How ignoring repeatability leads to magic. *Philosophy of the Social Sciences* 30:528–86. [JA]

Albright, A. & Hayes, B. (2001) *Rules vs. analogy in English past tenses: A computational/experimental study*. Department of Linguistics, UCLA. [JLM]

Altmann, E. M. (2002) Functional decay of memory for tasks. *Psychological Research* 66:287–97. [WDG]

Altmann, E. M. & Gray, W. D. (2000) An integrated model of set shifting and maintenance. In: *Proceedings of the third international conference on cognitive modeling,* pp. 17–24, ed. N. Taatgen & J. Aasman. Universal Press. [EMA]

(2002) Forgetting to remember: The functional relationship of decay and interference. *Psychological Science* 13(1):27–33. [WDG]

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

Anderson, J. R. (1974) Retrieval of propositional information from long-term memory. *Cognitive Psychology* 5:451–74. [rJRA]

(1976) *Language, memory, and thought*. Erlbaum. [aJRA]

(1983) *The architecture of cognition*. Harvard University Press. [arJRA]

(1990) *The adaptive character of thought*. Erlbaum. [arJRA]

(1991) Is human cognition adaptive? *Behavioral and Brain Sciences* 14:471–84. [aJRA]

(1993) *Rules of the mind*. Erlbaum. [aJRA, PAMG]

(2000) *Learning and memory*, 2nd edition. Wiley. [aJRA]

Anderson, J. R. & Betz, J. (2001) A hybrid model of categorization. *Psychonomic Bulletin and Review* 8:629–47. [aJRA]

Anderson, J. R., Bothell, D., Lebiere, C. & Matessa, M. (1998a) An integrated theory of list memory. *Journal of Memory and Language* 38:341–80. [aJRA]

Anderson, J. R., Boyle, C. F., Corbett, A. T. & Lewis, M. W. (1990) Cognitive modeling and intelligent tutoring. In: *Artificial intelligence and learning environments,* ed. W. J. Clancey & E. Soloway. Elsevier. [WJC]

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* 45:337–67. [aJRA]

Anderson, J. R., Corbett, A. T., Koedinger, K. & Pelletier, R. (1995) Cognitive tutors: Lessons learned. *The Journal of Learning Sciences* 4:167–207. [rJRA]

Anderson, J. R. & Douglass, S. (2001) Tower of Hanoi: Evidence for the cost of goal retrieval. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 27:1331–46. [aJRA]

Anderson, J. R. & Lebiere, C. (1998) *The atomic components of thought*. Erlbaum. [aJRA, PAMG]

Anderson, J. R., Lebiere, C., Lovett, M. C. & Reder, L. M. (1998b) ACT-R: A higher-level account of processing capacity. *Behavioral and Brain Sciences* 21:831–32. [aJRA]

Anderson, J. R., Qin, Y., Sohn, M-H., Stenger, V. A. & Carter, C. S. (2003) An information-processing model of the BOLD response in symbol manipulation tasks. *Psychonomic Bulletin and Review* 10:241–61. [arJRA]

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

Anderson, J. R. & Reder, L. M. (1999b) The size of the fan effect: Process not representation. *Journal of Experimental Psychology: General* 128:207–10. [rJRA]

Asher, N., Aurnague, M., Bras, M., Sblayrolles, P. & Vieu. L. (1994) Computing the spatiotemporal structure of discourse. *1st International Workshop on Computational Semantics,* Dept. of Computational Linguistics, University of Tilburg, NL. [AGBtM]

Atran, S. (2002) Modes of adaptationism: Muddling through cognition and language. Commentary on Andrews et al. *Behavioral and Brain Sciences* 25(4):504–06. [IP]

Baddeley, A. D. (1986) *Working memory*. Oxford University Press. [aJRA]

Ballard, D. H., Hayhoe, M. M., Pook, P. K. & Rao, R. P. N. (1997) Deictic codes for the embodiment of cognition. *Behavioral and Brain Sciences* 20(4):723–42. [WDG]

Baron-Cohen, S. (1996) Can children with autism integrate first and third person representations? *Behavioral and Brain Sciences* 19(1):123–24. [WJC]

Barresi, J. & Moore, C. (1996) Intentional relations and social understanding. *Behavioral and Brain Sciences* 19(1)107–54. [WJC]

Barrett, J. L. (1998) Cognitive constraints on Hindu concepts of the divine. *Journal for the Scientific Study of Religion* 37:608–19. [IP]

(1999) Theological correctness: Cognitive constraint and the study of religion. *Method and Theory in the Study of Religion* 11:325–39. [IP]

Barrett, J. L. & Keil, F. E. (1996) Conceptualizing a nonnatural entity: Anthropomorphism in God concepts. *Cognitive Psychology* 31:219–47. [IP]

Barwise, J. (1980) Infinitary logics. In: *Modern logic: A survey,* ed. E. Agazzi. Reidel. [YY]

Besner, D., Twilley, L., McCann, R. S. & Seergobin, K. (1990) On the connection between connectionism and data: Are a few words necessary? *Psychological Review* 97(3):432–46. [JLM]

Bever, T. G., Fodor, J. A. & Garret, M. (1968) A formal limitation of association. In: *Verbal behavior and general behavior theory,* ed. T. R. Dixon & D. L. Horton. Prentice Hall. [aJRA]

Bird, H., Lambon Ralph, M. A., Seidenberg, M. S., McClelland, J. L. & Patterson, K. (2003) Deficits in phonology and past-tense morphology: What's the connection? *Neuropsychologia* 48:502–26. [JLM]

Block, N. (1995) On a confusion about a function of consciousness. *Behavioral and Brain Sciences* 18:227–87. [rJRA, MO, YY]

Boardman, I., Grossberg, S., Myers, C. & Cohen, M. (1999) Neural dynamics of perceptual order and context effects for variable-rate speech syllables. *Perception and Psychophysics* 61:1477–1500. [SG]

Bock, K. (1986) Syntactic persistence in language production. *Cognitive Psychology* 18:355–87. [aJRA]

Bock, K. & Griffin, Z. M. (2000) The persistence of structural priming: Transient activation or implicit learning? *Journal of Experimental psychology: General* 129:177–92. [aJRA]

Boolos, G. & Jeffrey, R. (1989) *Computability and logic*. Cambridge University Press. [YY]

Botvinick, M. & Plaut, D. C. (submitted) Doing without schema hierarchies: A recurrent connectionist approach to routine sequential action and its pathologies. [aJRA]

Bouvet, S. (2001) Learning an ideal strategy in tic-tac-toe with DAC5. *Technical Report: Institute of Neuroinformatics* 2001–12. [PFMJV]

Boyer, P. (2001) *Religion explained: The evolutionary origins of religious thought*. Basic Books. [IP]

Bradski, G., Carpenter, G. A. & Grossberg, S. (1994) STORE working memory networks for storage and recall of arbitrary temporal sequences. *Biological Cybernetics* 71:469–80. [SG]

Bringsjord, S. (2000) Animals, zombanimals, and the total Turing Test: The essence of artificial intelligence. *Journal of Logic, Language, and Information* 9:397–18. [YY]

(2001) Are we evolved computers?: A critical review of Steven Pinker's *How the mind works. Philosophical Psychology* 14(2):227–43. [IP]

Bringsjord, S. & Zenzen, M. (2003) *Superminds: People harness hypercomputation, and more*. Kluwer. [YY]

Brown, J., Bullock, D. & Grossberg, S. (1999) How the basal ganglia use parallel

excitatory and inhibitory learning pathways to selectively respond to unexpected rewarding cues. *Journal of Neuroscience* 19:10502–511. [SG]
Brown, R. (1973) *A first language.* Harvard University Press. [aJRA,JLM]
Browne, A. & Sun, R. (2001) Connectionist inference models. *Neural Networks* 14:1331–55. [aJRA]
Budiu, R. (2001) *The role of background knowledge in sentence processing.* Doctoral dissertation, Carnegie Mellon University, Pittsburgh, PA. [aJRA]
Budiu, R. & Anderson, J. R. (submitted) Interpretation-based processing: A unified theory of semantic processing. *Cognitive Science.* [aJRA]
Bullock, D., Cisek, P. & Grossberg, S. (1998) Cortical networks for control of voluntary arm movements under variable force conditions. *Cerebral Cortex* 8:48–62. [SG]
Bullock, D., Grossberg, S. & Guenther, F. (1993a) A self-organizing neural model of motor equivalent reaching and tool use by a multijoint arm. *Journal of Cognitive Neuroscience* 5:408–35. [SG]
Bullock, D., Grossberg, S. & Mannes, C. (1993b) A neural network model for cursive script production. *Biological Cybernetics* 70:15–28. [SG]
Bunge, M. (1980) *The mind-body problem: A psychobiological approach.* Pergamon Press. [PAMG]
Burzio, L. (1999) Missing players: Phonology and the past-tense debate. Unpublished manuscript. Johns Hopkins University, Baltimore, MD. [aJRA]
Bybee, J. L. (1995) Regular morphology and the lexicon. *Language and Cognitive Processes* 10:425–55. [JLM]
Byrne, M. D. & Anderson, J. R. (1998) Perception and action. In: *The atomic components of thought,* ed. J. R. Anderson & C. Lebiere. Erlbaum. [aJRA]
(2001) Serial modules in parallel: The psychological refractory period and perfect time-sharing. *Psychological Review* 108:847 – 69. [aJRA]
Callan, D. E., Kent, R. D., Guenther, F. H. & Vorperian, H. K. (2000) An auditory-feedback-based neural network model of speech production that is robust to developmental changes in the size and shape of the articulatory system. *Journal of Speech, Language, and Hearing Research* 43:721–36. [SG]
Calvo, F. & Colunga, E. (2003) The statistical brain: Reply to Marcus' *The Algebraic Mind.* In: *Proceedings of the Twenty-Fifth Annual Conference of the Cognitive Science Society,* ed. R. Alterman & D. Kirsh. Erlbaum. [FCG]
(in preparation) Transfer of learning in infants: Combined Hebbian and error-driven learning. [FCG]
Carpenter, G. A., Gopal, S., Macomber, S., Martens, S., Woodcock, C. E. & Franklin, J. (1999) A neural network method for efficient vegetation mapping. *Remote Sensing of Environment* 70:326–38. [SG]
Carpenter, G. A. & Grossberg, S., eds. (1991) *Pattern recognition by self-organizing neural networks.* MIT Press. [SG]
Carpenter, G. A., Grossberg, S., Markuzon, N., Reynolds, J. H. & Rosen, D. (1992) Fuzzy ARTMAP: A neural network architecture for incremental supervised learning of analog multidimensional maps. *IEEE Transactions on Neural Networks* 3(5):698–713. [SG]
Carpenter, G. A. & Milenova, B. L. (2000) ART neural networks for medical data analysis and fast distributed learning. In: *Artificial neural networks in medicine and biology. Proceedings of the ANNIMAB-1 Conference, Göteborg, Sweden, 13–16 May 2000,* ed. H. Malmgren, M. Borga & L. Niklasson. Springer-Verlag. (Springer series, *Perspectives in Neural Computing.*) [SG]
Chaiken, S. & Trope, Y., eds. (1999) *Dual-process theories in social psychology.* Guilford Press. [IP]
Chalmers, D. J. (1996) *The conscious mind.* Oxford University Press. [MO]
Chase, W. G. & Ericsson, K. A. (1982) Skill and working memory. In: *The psychology of learning and motivation, vol. 16,* ed G. H. Bower. Academic Press. [aJRA]
Chomsky, N. A. (1965) *Aspects of a theory of syntax.* MIT Press. [aJRA]
Clancey, W. J. (1999a) *Conceptual coordination: How the mind orders experience in time.* Erlbaum. [WJC]
(1999b) Studying the varieties of consciousness: Stories about zombies or the life about us? *Journal of the Learning Sciences* 8(3–4):525–40. [WJC]
(2000) Conceptual coordination bridges information processing and neuropsychology. *Behavioral and Brain Sciences* 23(6):919–22. [WJC]
Clark, A. (1997) *Being there.* MIT Press. [DS]
(1998) The dynamic challenge. *Cognitive Science* 21(4):461–81. [aJRA]
(1999) Where brain, body, and world collide. *Journal of Cognitive Systems Research* 1:5–17. [aJRA]
Cleeremans, A. (1993) *Mechanisms of implicit learning: Connectionist models of sequence processing.* MIT Press. [aJRA]
Cohen, J. D. & Schooler, J. W., eds. (1997) *Scientific approaches to consciousness: 25th Carnegie Symposium on Cognition.* Erlbaum. [aJRA]
Cohen, M. A. & Grossberg, S. (1986) Neural dynamics of speech and language coding: Developmental programs, perceptual grouping, and competition for short-term memory. *Human Neurobiology* 5:1–22. [SG]
Collins, M. (1999) *Head-driven statistical models for nature language parsing.* Doctoral dissertation, University of Pennsylvania. [aJRA]
Coltheart, M., Curtis, B., Atkins, P. & Haller, M. (1993) Models of reading aloud: Dual-route and parallel-distributed-processing approaches. *Psychological Review* 100(4):589–608. [JLM]
Commons, M. L. & Richards, F. A. (2002) Organizing components into combinations: How stage transition works. *Journal of Adult Development* 9(3):159–77. [MLC]
Commons, M. L., Trudeau, E. J., Stein, S. A., Richards, F. A. & Krause, S. R. (1998) The existence of developmental stages as shown by the hierarchical complexity of tasks. *Developmental Review* 8(3):237–78. [MLC]
Contreras-Vidal, J. L., Grossberg, S. & Bullock, D. (1997) A neural model of cerebellar learning for arm movement control: Cortico-spino-cerebellar dynamics. *Learning and Memory* 3:475–502. [SG]
Cosmides, L. & Tooby, J. (2000a) Consider the source: The evolution of adaptations for decoupling and metarepresentation. In: *Metarepresentations: A multidisciplinary perspective,* ed. D. Sperber. Oxford University Press. [IP]
(2000b) The cognitive neuroscience of social reasoning. In: *The new cognitive neurosciences,* 2nd edition, ed. M. S. Gazzaniga. MIT Press. [aJRA]
Dawson, T. L. (2002) A comparison of three developmental stage scoring systems. *Journal of Applied Measurement* 3(2):146–89. [MLC]
Denes-Raj, V. & Epstein, S. (1994) Conflict between intuitive and rational processing: When people behave against their better judgment. *Journal of Personality and Social Psychology* 66(5):819–29. [IP]
Dennett, D. C. (1991) *Consciousness explained.* Little, Brown. [aJRA]
Dennett, D. C. & Kinsbourne, M. (1995) Time and the observer: The where and when of consciousness in the brain. *Behavioral and Brain Sciences* 15(2):183–247. [aJRA]
Dolan, C. P. & Smolensky, P. (1989) Tensor product production system: A modular architecture and representation. *Connection Science* 1:53–68. [aJRA]
Edelman, G. M. (1989) *The remembered present: A biological theory of consciousness.* BasicBooks. [PAMG]
(1992) *Bright air, brilliant fire: On the matter of the mind.* BasicBooks. [PAMG]
Edelman, G. M. & Tononi, G. (2000) *Consciousness: How matter becomes imagination.* Penguin Press. [PAMG]
Ehret, B. D., Gray, W. D. & Kirschenbaum, S. S. (2000) Contending with complexity: Developing and using a scaled world in applied cognitive research. *Human Factors* 42(1):8–23. [WDG]
Elman, J. L. (1995) Language as a dynamical system. In: *Mind as motion: Explorations in the dynamics of cognition,* ed. R. F. Port & T. V. Gelder. MIT Press. [aJRA]
Elman, J. L., Bates, E. A, Johnson, M. H., Karmiloff-Smith, A., Parisi, D. & Plunkett, K. (1996) *Rethinking innateness: A connectionist perspective on development.* MIT Press. [arJRA, CFO]
Emond, B. (in preparation) ACT-R/WN: Towards an implementation of WordNet in the ACT-R cognitive architecture. [aJRA]
Emond, B. & Ferres, L. (2001) Modeling the false-belief task: An ACT-R implementation of Wimmer & Perner (1983). Paper presented at the Second Bisontine Conference for Conceptual and Linguistic Development in the Child Aged from 1 to 6 Years, Besançon, France, March 21–23, 2001. [aJRA]
Eng, K., Klein, D., Bäbler, A., Bernardet, U., Blanchard, U., Costa, M., Delbrück, T., Douglas, R. J., Hepp, K., Manzolli, J., Mintz, M., Roth, F., Rutishauser, U., Wassermann, K., Whatley, A. M., Wittmann, A., Wyss, R., Verschure, P. F. M. J. (2003) Design for a brain revisited: The neuromorphic design and functionality of the interactive space Ada. *Reviews in the Neurosciences* 1–2:145–80. [PFMJV]
Engel, A. K., Fries, P. & Singer, W. (2001) Dynamic predictions, oscillations and synchrony in top-down processing. *Nature Reviews Neuroscience* 2:704–16. [SG]
Epstein, S., Lipson, A., Holstein, C. & Huh, E. (1992) Irrational reactions to negative outcomes: Evidence for two conceptual systems. *Journal of Personality and Social Psychology* 62(2):328–39. [IP]
Epstein, S. & Pacini, R. (1999) Some basic issues regarding dual-process theories from the perspective of cognitive-experiential self-theory. In: *Dual-process theories in social psychology,* ed. S. Chaiken & Y. Trope. Guilford Press. [IP]
Ericsson, K. A. & Kintsch, W. (1995) Long-term working memory. *Psychological Review* 102:211–45. [aJRA]
Fellbaum, C., ed. (1998) *WordNet: An electronic lexical database.* MIT Press. [aJRA]
Ferreira, F. & Clifton, C. (1986) The independence of syntactic processing. *Journal of Memory and Language* 25:348–68. [aJRA]
Fiala, J. C., Grossberg, S. & Bullock, D. (1996) Metabotropic glutamate receptor activation in cerebellar Purkinje cells as substrate for adaptive timing of the classically conditioned eye-blink response. *Journal of Neuroscience* 16:3760–74. [SG]
Fincham, J. M., VanVeen, V., Carter, C. S., Stenger, V. A. & Anderson, J. R. (2002) Integrating computational cognitive modeling and neuroimaging: An event-

related fMRI study of the Tower of Hanoi task. *Proceedings of the National Academy of Science* 99:3346–51. [aJRA]
Fodor, J. A. (1983) *The modularity of mind.* MIT Press/Bradford Books. [aJRA]
(2000) *The mind doesn't work that way.* MIT Press. [aJRA]
Frank, M. J., Loughry, B. & O'Reilly, R. C. (2000) Interactions between frontal cortex and basal ganglia in working memory: A computational model. *Technical Report* (00–01, November), Institute of Cognitive Science, University of Colorado, Boulder, CO. [aJRA]
Gancarz, G. & Grossberg, G. (1999) A neural model of the saccadic eye movement control explains task-specific adaptation. *Vision Research* 39:3123–43. [SG]
Gelepithis, P. A. M. (1984) *On the foundations of artificial intelligence and human cognition.* Ph. D. thesis, Department of Cybernetics, Brunel University, England. [PAMG]
(1991) The possibility of machine intelligence and the impossibility of human-machine communication. *Cybernetica* 34(4):255–68. [PAMG]
(1997) A Rudimentary theory of information: Consequences for information science and information systems. *World Futures* 49:263–74. (Reprinted in: *The quest for a unified theory of information,* ed. W. Hofkirchner. Gordon and Breach.) [PAMG]
(1999) Embodiments of theories of mind: A review and comparison. In: *Computational methods and neural networks,* ed. M. P. Bekakos, M. Sambandham & D. J. Evans. Dynamic. [PAMG]
(2001) A concise comparison of selected studies of consciousness, *Cognitive Systems* 5(4):373–92. [PAMG]
(2002) An axiomatic approach to the study of mind. *Res-Systemica: Proceedings of the Fifth European Systems Science Congress, October 2002, Crete.* http://www.afscet.asso.fr/resSystemica/. [PAMG]
Gelepithis, P. A. M. & Goodfellow, R. (1992) An alternative architecture for intelligent tutoring systems: Theoretical and implementational aspects. *Interactive Learning International* 8(3):171–75. [PAMG]
Gelepithis, P. A. M. & Parillon, N. (2002) Knowledge management: Analysis and some consequences. In: *Knowledge management and business process reengineering,* ed. V. Hlupic. Idea Book. [PAMG]
Gibson, J. J. (1979) *The ecological approach to visual perception.* Houghton Mifflin. [MO]
Giere, R. (1998) *Explaining science.* Chicago University Press. [PAMG]
Gigerenzer, G. (2000) *Adaptive thinking: Rationality in the real world.* Oxford University Press. [aJRA]
Gopnik, M. & Crago, M. B. (1991) Familial aggregation of a developmental language disorder. *Cognition* 39:1–50. [JLM]
Gould, S. J. & Lewontin, R. C. (1979) The Spandrels of San Marco and the Panglossian paradigm: A critique of the adaptationist program. *Proceedings of the Royal Society of London* 205:581–98. [aJRA]
Granger, E., Rubin, M., Grossberg, S. & Lavoie, P. (2001) A what-and-where fusion neural network for recognition and tracking of multiple radar emitters. *Neural Networks* 14:325–44. [SG]
Gray, W. D. & Boehm-Davis, D. A. (2000) Milliseconds matter: An introduction to microstrategies and to their use in describing and predicting interactive behavior. *Journal of Experimental Psychology: Applied* 6(4):322–35. [WDG]
Gray, W. D., John, B. E. & Atwood, M. E. (1993) Project Ernestine: Validating a GOMS analysis for predicting and explaining real-world performance. *Human-Computer Interaction* 8(3):237–309. [WDG]
Gray, W. D., Schoelles, M. J. & Fu, W.-T. (2000) Modeling a continuous dynamic task. In: *Third International Conference on Cognitive Modeling,* ed. N. Taatgen & J. Aasman. Universal Press. [WDG]
Gray, W. D., Schoelles, M. J. & Myers, C. W. (2002) Computational cognitive models ISO ecologically optimal strategies. In: *46th Annual Conference of the Human Factors and Ergonomics Society,* pp. 492–96. Human Factors & Ergonomics Society. [WDG]
Green C. D. (1998) Are connectionist models theories of cognition? *Psycoloquy* 9(04). http://www.cogsci.ecs.soton.ac.uk/cgi/psyc/newpsy?9.04. [PNP]
Greeno, J. G. (1989) Situations, mental models and generative knowledge. In: *Complex information processing: The impact of Herbert A. Simon,* ed. D. Klahr & K. Kotovsky. Erlbaum. [aJRA]
Grossberg, S. (1976) Adaptive pattern classification and universal recoding, II: Feedback, expectation, olfaction, and illusions. *Biological Cybernetics* 23:187–202. [SG]
(1978a) A theory of human memory: Self-organization and performance of sensory-motor codes, maps, and plans. In: *Progress in theoretical biology, vol. 5,* ed. R. Rosen & F. Snell. Academic Press. [SG]
(1978b) Behavioral contrast in short-term memory: Serial binary memory models or parallel continuous memory models? *Journal of Mathematical Psychology* 3:199–219. [SG]
(1980) How does a brain build a cognitive code? *Psychological Review* 87:1–51. [SG]
(1988) Nonlinear neural networks: Principles, mechanisms, and architectures. *Neural Networks* 1:17–61. [SG]
(1999a) How does the cerebral cortex work? Learning, attention, and grouping by the laminar circuits of visual cortex. *Spatial Vision* 12:163–86. [SG]
(1999b) The link between brain learning, attention, and consciousness. *Consciousness and Cognition* 8:1–44. [SG]
(2000a) How hallucinations may arise from brain mechanisms of learning, attention, and volition. *Journal of the International Neuropsychological Society* 6:583–92. [SG]
(2000b) The complementary brain: Unifying brain dynamics and modularity. *Trends in Cognitive Sciences* 4:233–46. [SG]
(2000c) The imbalanced brain: From normal behavior to schizophrenia. *Biological Psychiatry* 48:81–98. [SG]
Grossberg, S., Boardman, I. & Cohen, C. (1997a) Neural dynamics of variable-rate speech categorization. *Journal of Experimental Psychology: Human Perception and Performance* 23:418–503. [SG]
Grossberg, S. & Kuperstein, M. (1989) Neural dynamics of adaptive sensory-motor control: Expanded edition. Pergamon Press. [SG]
Grossberg, S. & Merrill, J. W. L. (l992) A neural network model of adaptively timed reinforcement learning and hippocampal dynamics. *Cognitive Brain Research* 1:3–38. [SG]
(l996) The hippocampus and cerebellum in adaptively timed learning, recognition, and movement. *Journal of Cognitive Neuroscience* 8:257–77. [SG]
Grossberg, S. & Myers, C. W. (2000) The resonant dynamics of speech perception: Interword integration and duration-dependent backward effects. *Psychological Review* 107:735–67. [SG]
Grossberg, S. & Paine, R. W. (2000) A neural model of corticocerebellar interactions during attentive imitation and predictive learning of sequential handwriting movements. *Neural Networks* 13:999–1046. [SG]
Grossberg, S., Roberts, K., Aguilar, M. & Bullock, D. (1997b) A neural model of multimodal adaptive saccadic eye movement control by superior colliculus. *Journal of Neuroscience* 17:9706–25. [SG]
Grossberg, S. & Stone, G. O. (1986a) Neural dynamics of attention switching and temporal order information in short-term memory. *Memory and Cognition* 14:451–68. [SG]
(1986b) Neural dynamics of word recognition and recall: Attentional priming, learning, and resonance. *Psychological Review* 93:46–74. [SG]
Guenther, F. H. (1995) Speech sound acquisition, coarticulation, and rate effects in a neural network model of speech production. *Psychological Review* 102:594–621. [SG]
Hahn, U. & Nakisa, R. C. (2000) German inflection: Single-route or dual-route? *Cognitive Psychology* 41:313–60. [JLM]
Hameroff, S. R., Kaszniak, A. W. & Scott, A. C., eds. (1998) *Toward a science of consciousness II: The second Tucson discussions and debates.* MIT Press. [PAMG]
Harnad, S. (1990) The symbol grounding problem. *Physica D* 42:335–46. [aJRA]
(1994) Computation is just interpretable symbol manipulation; Cognition isn't. *Minds and Machines* 4:379–90. [aJRA]
Hartley, R. F. (2000) Cognition and the computational power of connectionist networks. *Connection Science* 12(2):95–110. [aJRA]
Hartley, R. & Szu, H. (1987) A comparison of the computational power of neural network models. *Proceedings of the First International Conference on Neural Networks* 3:15–22. [aJRA]
Haverty, L. A., Koedinger, K. R., Klahr, D. & Alibali, M. W. (2000) Solving induction problems in mathematics: Not-so-trivial pursuit. *Cognitive Science* 24(2):249–98. [aJRA]
Hebb, D. O. (1949) *The organization of behavior.* Wiley. [PAMG]
Hinrichs, E. (1986) Temporal anaphora in discourses of English. *Linguistics and Philosophy* 9:63–82. [AGBtM]
Hinton, G. E. & Sejnowsky, T. J. (1986) Learning and relearning in Boltzmann machines. In: *Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1: Foundations,* ed. D. E. Rumelhart, J. L. McClelland & The PDP Group. MIT Press. [aJRA]
Hoeffner, J. H. (1996) A single mechanism account of the acquisition and processing of regular and irregular inflectional morphology. Unpublished doctoral dissertation, Department of Psychology, Carnegie Mellon University, Pittsburgh, PA. [JLM]
Hofstötter, C., Mintz, M. & Verschure, P. F. M. J. (2002) The cerebellum in action: A simulation and robotics study. *European Journal of Neuroscience* 16:1361–76. [PFMJV]
Holyoak, K. J. & Spellman, B. A. (1993) Thinking. *Annual Review of Psychology* 44:265–315. [IP]
Hopfield, J. J. (1982) Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences USA* 79:2554–58. [aJRA]
Hornik, K., Stinchcombe, M. & White, H. (1989) Multilayer feed forward networks are universal approximators. *Neural Computation* 2:210–15. [aJRA]
Howes, A. & Young, R. M. (1997) The role of cognitive architecture in modelling

the user: Soar's learning mechanism. *Human-Computer Interaction* 12:311–43. [RMY]

Hummel, J. E. & Holyoak, K. J. (1998) Distributed representations of structure. A theory of analogical access and mapping. *Psychological Review* 104:427–66. [aJRA]

Hutchins, E. (1995) *Cognition in the wild.* MIT Press. [DS]

Joanisse, M. F. & Seidenberg, M. S. (1999) Impairments in verb morphology following brain injury: A connectionist model. *Proceedings of the National Academy of Sciences* 96:7592–97. [JLM]

Johnson, M. H., Munakata, Y. & Gilmore, R. O., eds. (2002) *Brain development and cognition: A reader.* Blackwell. [PAMG]

Jones, G., Ritter, F. E. & Wood, D. J. (2000) Using a cognitive architecture to examine what develops. *Psychological Science* 11(2):1–8. [aJRA]

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* 20:27–41. [aJRA]

Jongman, L. & Taatgen, N. A. (1999) An ACT-R model of individual differences in changes in adaptivity due to mental fatigue. In: *Proceedings of the 21st Annual Conference of the Cognitive Science Society.* Erlbaum. [aJRA]

Kandel, E. R. & O'Dell, T. J. (1992) Are adult learning mechanisms also used for development? *Science* 258:243–45. [SG]

Karmiloff-Smith, A. (1992) *Beyond modularity: A developmental perspective on cognitive science.* MIT Press. [CFO]

(1997) Crucial differences between developmental cognitive neuroscience and adult neuropsychology. *Developmental Neuropsychology* 13(4):513–24. [CFO]

(1998) Development itself is the key to understanding developmental disorders. *Trends in Cognitive Sciences* 2(10):389–98. [CFO]

Karmiloff-Smith, A., Plunkett, K., Johnson, M. H., Elman, J. L. & Bates, E. A. (1998) What does it mean to claim that something is "innate"? *Mind and Language* 13(4):588–97. [CFO]

Karmiloff-Smith, A., Scerif, G. & Ansari, D. (2003) Double dissociations in developmental disorders? Theoretically misconceived, empirically dubious. *Cortex* 39:161–3. [CFO]

Karmiloff-Smith, A., Scerif, G. & Thomas, M. S. C. (2002) Different approaches to relating genotype to phenotype in developmental disorders. *Developmental Psychobiology* 40:311–22. [CFO]

Kilgard, M. P. & Merzenich, M. M. (1998) Cortical map reorganization enabled by nucleus basalis activity. *Science* 279:1714–18. [PFMJV]

Kimberg, D. Y. & Farah, M. J. (1993) A unified account of cognitive impairments following frontal lobe damage: The role of working memory in complex, organized behavior. *Journal of Experimental Psychology: General* 122:411–28. [aJRA]

Kirsh, D. & Maglio, P. P. (1994) On distinguishing epistemic from pragmatic action. *Cognitive Science* 18(4):513–49. [DS]

Koedinger, K. R., Anderson, J. R., Hadley, W. H. & Mark, M. (1997) Intelligent tutoring goes to school in the big city. *International Journal of Artificial Intelligence in Education* 8:30–43. [rJRA]

Laird, J. E. (1986) Universal subgoaling. In: *Universal subgoaling and chunking: The automatic generation and learning of goal hierarchies,* ed. J. E. Laird, P. S. Rosenbloom & A. Newell. Kluwer. [RMY]

Lakatos, I. (1970) Falsification and the methodology of scientific research programmes. In: *Criticism and the growth of knowledge,* ed. I. Lakatos & A. Musgrave. Cambridge University Press. [CFO]

Langacker, R. W. (1986) An introduction to cognitive grammar. *Cognitive Science* 10(1):1–40. [WJC]

Langley, P. (1999) Concrete and abstract models of category learning. In: *Proceedings of the 21st Annual Conference of the Cognitive Science Society.* Erlbaum. [PT]

Lave, J. (1988) *Cognition in practice: Mind, mathematics, and culture in everyday life.* Cambridge University Press. [aJRA]

Lebiere, C. (1998) *The dynamics of cognition: An ACT-R model of cognitive arithmetic.* Doctoral dissertation. CMU Computer Science Dept. Technical Report CMU-CS-98–186. Pittsburgh, PA. [aJRA]

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.* [aJRA]

Lerch, F. J., Gonzalez, C. & Lebiere, C. (1999) Learning under high cognitive workload. In: *Proceedings of the Twenty-first Conference of the Cognitive Science Society.* Erlbaum. [aJRA]

Lewis, R. L. (1999) Attachment without competition: A race-based model of ambiguity resolution in a limited working memory. Presented at the CUNY Sentence Processing Conference, New York. [aJRA]

Liben, L. S. (1987) Approaches to development and learning: Conflict and congruence. In: *Development and learning: Conflict or congruence?* ed. L. S. Liben. Erlbaum. [SS]

Lieberman, M. D. (2000) Intuition: A social cognitive neuroscience approach. *Psychological Bulletin* 126(1):109–37. [IP]

Lieberman, M. D., Gaunt, R., Gilbert, D. T. & Trope, Y. (2002) Reflexion and reflection: A social cognitive neuroscience approach to attributional inference. *Advances in Experimental Social Psychology* 34:200–250. [IP]

Lodge, D. (1984) *Small world.* Penguin. [NAT]

Logan, G. D. (1988) Toward an instance theory of automatization. *Psychological Review* 95:492–527. [aJRA]

Lovett, M. C. (1998) Choice. In: *The atomic components of thought,* ed. J. R. Anderson & C. Lebiere. Erlbaum. [aJRA]

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. [aJRA]

Lovett, M. C., Reder, L. & Lebiere, C. (1997) Modeling individual differences in a digit working memory task. In: *Proceedings of the 19th Annual Conference of the Cognitive Science Society,* ed. M. G. Shafto. & P. Langley. Erlbaum. [NAT]

MacDonald, M. C., Pearlmutter, N. J. & Seidenberg, M. S. (1994) The lexical nature of syntactic ambiguity resolution. *Psychological Review* 101:676–703. [aJRA]

Magerman, D. (1995) Statistical decision-tree models for parsing. In: *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics.* [aJRA]

Marcus, G. F. (2001) *The algebraic mind: Integrating connectionism and cognitive science.* MIT Press. [aJRA, FCG, JLM]

Marcus G. F., Brinkmann U., Clahsen H., Wiese R. & Pinker S. (1995) German inflection: The exception that proves the rule. *Cognitive Psychology* 29:189–256. [JLM]

Marcus, G. F., Vijayan, S., Rao, S. B. & Vishton, P. M. (1999) Rule learning in seven-month-old infants. *Science* 283:77–80. [FCG]

Marín, J., Calvo, F. & Valenzuela, J. (2003) The creolization of pidgin: A connectionist exploration. In: *Proceedings of the European Cognitive Science Conference,* ed. F. Schmalhofer, R. M. Young & G. Katz. Erlbaum. [FCG]

Marr, D. (1982) *Vision.* Freeman. [JA, PT, HW]

Massaro, D. W. (1989) Testing between the TRACE model and the fuzzy logical model of speech perception. *Cognitive Psychology* 21:398–421. [JLM]

(1998) *Perceiving talking faces: From speech perception to a behavioral principle.* MIT Press. [PFMJV]

Massaro, D. W. & Cohen, M. M. (1991) Integration versus interactive activation: The joint influence of stimulus and context in perception. *Cognitive Psychology* 23:558–614. [JLM]

Matessa, M. (2001) *Simulating adaptive communication.* Doctoral dissertation, Carnegie Mellon University, Pittsburgh, PA. [aJRA]

Matessa, M. & Anderson, J. R. (2000) Modeling focused learning in role assignment. *Language and Cognitive Processes* 15:263–92. [aJRA]

Mayr, E. (1988) *Toward a new philosophy of biology: Observations of an evolutionist.* Harvard University Press. [PAMG]

McClelland, J. L. (1979) On time relations of mental processes: An examination of systems of processes in cascade. *Psychological Review* 86:287–330. [aJRA]

(1991) Stochastic interactive processes and the effect of context on perception. *Cognitive Psychology* 23:1–44. [JLM]

(1994) The interaction of nature and nurture in development: A parallel distributed processing perspective. In: *International perspectives on psychological science, vol. 1: Leading themes,* ed. P. Bertelson, P. Eelen & G. D'Ydewalle. Erlbaum. [FCG]

McClelland, J. L. & Chappell, M. (1998) Familiarity breeds differentiation: A Bayesian approach to the effects of experience in recognition memory. *Psychological Review* 105:724–60. [aJRA]

McClelland, J. L. & Elman, J. L. (1986) The TRACE model of speech perception. *Cognitive Psychology* 18:1–86. [JLM]

McClelland, J. L., McNaughton, B. L. & O'Reilly, R. C. (1995) Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionist models of learning and memory. *Psychological Review* 102:419–57. [aJRA]

McClelland, J. L. & Patterson, K. (2002a) Rules or connections in past-tense inflections: What does the evidence rule out? *Trends in Cognitive Sciences* 6:465–72. [JLM]

(2002b) 'Words or Rules' cannot exploit the regularity in exceptions. *Trends in Cognitive Sciences* 6:464–65. [JLM]

McClelland, J. L. & Plaut, D. C. (1999) Does generalization in infant learning implicate abstract algebra-like rules? *Trends in Cognitive Sciences* 3:166–68. [aJRA]

McClelland, J. L. & Rumelhart, D. E. (1981) An interactive activation model of context effects in letter perception: Part 1. An account of basic findings. *Psychological Review* 88(5):375–407. [aJRA,JLM]

(1986) *Parallel distributed processing. Explorations in the microstructure of cognition, vol. 2.* MIT Press/Bradford. [aJRA]

McCloskey, M. & Cohen, N. J. (1989) Catastrophic interference in connectionist networks: The sequential learning problem. In: *The psychology of learning*

and motivation: Advances in research and theory, vol. 24, ed. G. H. Bower. Academic Press. [aJRA]
Menzel, R. & Muller, U. (1996) Learning and memory in honeybees: From behavior to neural substrate. *Annual Review Neuroscience* 19:379–404. [PFMJV]
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. [arJRA]
Minsky, M. L. & Papert, S. A. (1969) *Perceptrons.* MIT Press. [aJRA]
Misker, J. M. V. & Anderson, J. R. (2003) Combining optimality theory and a cognitive architecture. In: *Proceedings of the Fifth International Conference on Cognitive Modeling, Bamberg, Germany, April 2003.* [rJRA]
Movellan, J. R. & McClelland, J. L. (2001) The Morton-Massaro law of information integration: Implications for models of perception. *Psychological Review* 108(1):113–48. [JLM]
Nagel, T. (1974) What is it like to be a bat? *Philosophical Review* 83:435–51. [MO]
Newcombe, N. (1998) Defining the "radical middle." *Human Development* 41:210–14. [CFO]
Newell, A. (1973) You can't play 20 questions with nature and win: Projective comments on the papers of this symposium. In: *Visual information processing,* ed. W. G. Chase. Academic Press. [EMA, arJRA, YY]
(1980) Physical symbol systems. *Cognitive Science* 4:135–83. [arJRA, PAMG, SS]
(1990) *Unified theories of cognition.* Harvard University Press. [arJRA, PAMG, WDG, AGBtM, SS, HW, RMY]
(1992) Précis of *Unified theories of cognition. Behavioral and Brain Sciences* 15:425–92. [aJRA, MLC]
Newell, A. & Card, S. K. (1985) The prospects for psychological science in human-computer interaction. *Human-Computer Interaction* 1(3):209–42. [WDG]
Newell, A. & Simon, H. A. (1963/1995) GPS, a program that simulates human thought. In: *Computers and thought,* ed. E. Feigenbaum, J. Feldman & P. Armer. AAAI Press. [MLC]
(1972) *Human problem solving.* Prentice Hall. [aJRA]
Norman, D. A. & Shallice, T. (1986) Attention to action: Willed and automatic control of behaviour. In: *The design of everyday things,* ed. R. J. Davidson, G. E. Schwartz & D. Shapiro. Doubleday. [MO]
Oaksford, M. & Chater, N., eds. (1998) *Rational models of cognition.* Oxford University Press. [aJRA]
O'Hear, A. (1997) *Beyond evolution: Human nature and the limits of evolutionary explanation.* Clarendon Press. [PAMG]
Ohlsson, S. & Jewett, J. J. (1997) Simulation models and the power law of learning. *Proceedings of the 19th Annual Conference of the Cognitive Science Society.* Erlbaum. [PT]
O'Reilly, R. & Munakata, Y. (2000) *Computational explorations in cognitive neuroscience: Understanding the mind by simulating the brain.* MIT Press. [FCG, HW]
Overgaard, M. (in press) On the theoretical and methodological foundations for a science of consciousness. *Bulletin from Forum for Antropologisk Psykologi.* [MO]
Palmer-Brown, D., Tepper, J. A. & Powell, H. M. (2002) Connectionist natural language parsing. *Trends in Cognitive Sciences* 6:437–42. [FCG]
Partee, B. (1984) Nominal and temporal anaphora. *Linguistics and Philosophy* 7:243–86. [AGBtM]
(1997) Montague grammar. In: *Handbook of logic and language,* ed. J. van Benthem & A. G. B. ter Meulen. Elsevier Science/MIT Press. [AGBtM]
Pashler, H. (1998) *The psychology of attention.* MIT Press. [aJRA]
Pavlov, I. P. (1928) *Lectures on conditioned reflexes: Twenty-five years of objective study of the higher nervous ability (behavior) of animals.* International Publishers. [PFMJV]
Penrose, R. (1989) *The emperor's new mind: Concerning computers, minds, and the laws of physics.* Oxford University Press. [HW]
(1996) *Shadows of the mind: A search for the missing science of consciousness.* Oxford University Press. [HW]
(1997) *The large, the small and the human mind.* Cambridge University Press. [HW]
Piaget, J. (1967/1971) *Biology and Knowledge.* University of Chicago Press. [PAMG]
Pinker, S. (1991) Rules of language. *Science* 253:530–35. [JLM]
(1994) *The language instinct.* Morrow. [aJRA]
(1997) *How the mind works.* Norton. [IP]
Pinker, S. & Bloom, P. (1990) Natural language and natural selection. *Behavioral and Brain Sciences* 13(4):707–84. [aJRA]
Pinker, S. & Ullman, M. T. (2002a) The past and future of the past tense. *Trends in Cognitive Sciences* 6:456–63. [JLM]
(2002b) Combination and structure, not gradedness, is the issue. *Trends in Cognitive Sciences* 6:472–74. [JLM]
Plaut, D. C. & Booth, J. R. (2000) Individual and developmental differences in semantic priming: Empirical and computational support for a single-mechanism account of lexical processing. *Psychological Review* 107:786–823. [aJRA]
Plaut, D. C., McClelland, J. L. & Seidenberg, M. S. (1995) Reading exception words and pseudowords: Are two routes really necessary? In: *Connectionist models of memory and language,* ed. J. P. Levy, D. Bairaktaris, J. A. Bullinaria & P. Cairns. UCL Press. [JLM]
Plaut, D. C., McClelland, J. L., Seidenberg, M. S. & Patterson, K. (1996) Understanding normal and impaired word reading: Computational principles in quasi-regular domains. *Psychological Review* 103:56–115. [JLM]
Plunkett, K., Karmiloff-Smith, A., Bates, E., Elman, J. L. & Johnson, M. H. (1997) Connectionism and developmental psychology. *Journal of Child Psychology and Psychiatry* 38:53–80. [CFO]
Plunkett, K. & Marchman, V. A. (1991) U-shaped learning and frequency effects in a multi-layered perceptron: Implications for child language acquisition. *Cognition* 38:43–102. [JLM]
Pollen, D. A. (1999) On the neural correlates of visual perception. *Cerebral Cortex* 9:4–19. [SG]
Pomerleau, D. A. (1991) Efficient training of artificial neural networks for autonomous navigation. *Neural Computation* 3:88–97. [aJRA]
Pomerleau, D. A., Gowdy, J. & Thorpe, C. E. (1991) Combining artificial neural networks and symbolic processing for autonomous robot guidance. *Engineering Applications of Artificial Intelligence* 4:279–85. [aJRA]
Popper, K. R. (1963) *Conjectures and refutations: The growth of scientific knowledge.* Routledge. [NAT]
Prince, A. & Smolensky, P. (1993) Optimality theory: Constraint interaction in generative grammar. *Technical Report CU-CS-696–93,* Department of Computer Science, University of Colorado at Boulder, and *Technical Report TR-2,* Rutgers Center for Cognitive Science, Rutgers University, New Brunswick, NJ. April. [rJRA]
Prudkov, P. (1994) A model of self-organization of cognitive processes. *Cognitive Systems* 4(1):1–19. [PNP]
Pyysiäinen, I. (2003) True fiction: Philosophy and psychology of religious belief. *Philosophical Psychology* 16(1):109–25. [IP]
Qin, Y., Sohn, M-H, Anderson, J. R., Stenger, V. A., Fissell, K., Goode, A. & Carter, C. S. (2003) Predicting the practice effects on the blood oxygenation level-dependent (BOLD) function of fMRI in a symbolic manipulation task. *Proceedings of the National Academy of Sciences USA* 100(8):4951–56. [rJRA]
Quartz, S. R. (1993) Neural networks, nativism, and the plausibility of constructivism. *Cognition* 48:223–42. [SS]
Quinn, R. & Espenschied, K. (1993) Control of a hexapod robot using a biologically inspired neural network. In: *Biological neural networks in invertebrate neuroethology and robotics,* ed. R. Beer et al. Academic Press. [DS]
Raizada, R. & Grossberg, S. (2003) Towards a theory of the laminar architecture of cerebral cortex: Computational clues from the visual system. *Cerebral Cortex* 13:100–13. [SG]
Ramscar, M. (2002) The role of meaning in inflection: Why the past tense does not require a rule. *Cognitive Psychology* 45(1):45–94. [JLM]
Ratcliff, R. (1990) Connectionist models of recognition memory: Constraints imposed by learning and forgetting functions. *Psychological Review* 97:285–308. [aJRA]
Ratcliff, R., Van Zandt, T & McKoon, G. (1999) Connectionist and diffusion models of reaction time. *Psychological Review* 106:261–300. [aJRA]
Reder, L. M., Nhouyvansivong, A., Schunn, C. D., Ayers, M. S., Angstadt, P. & Hiraki, K. (2000) A mechanistic account of the mirror effect for word frequency: A computational model of remember/know judgments in a continuous recognition paradigm. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 26:294–320. [aJRA]
Revonsuo, A. & Kampinnen, M., eds. (1994) *Consciousness in philosophy and cognitive neuroscience.* Erlbaum. [PAMG]
Roberts, S. & Pashler, H. (2000) How persuasive is a good fit? A comment on theory testing. *Psychological Review* 107:358–67. [aJRA]
Rogers, R. D. & Monsell, S. (1995) Costs of a predictable switch between simple cognitive tasks. *Journal of Experimental Psychology: General* 124(2):207–31. [EMA, WDG]
Rogers, T. T. & McClelland, J. L. (2003) *Semantic cognition: A parallel distributed processing approach.* MIT Press. [aJRA]
Rolls, E. T. (2000) Memory systems in the brain. *Annual Reviews, Psychology* 51(1):599–630. [IP]
Rolls, E. T. & Treves, A. (1998) *Neural networks and brain function.* Oxford University Press. [FCG]
Roy, A., Govil, S. & Miranda, R. (1997) A neural network learning theory and a polynomial time RBF algorithm. *IEEE Transactions on Neural Networks* 8(6):1301–13. [AR]
Rumelhart, D. E. & McClelland, J. L. (1982) An interactive activation model of context effects in letter perception: Part 2. The contextual enhancement effect

and some tests and extensions of the model. *Psychological Review* 89:60–94. [aJRA]

(1986a) On learning the past tenses of English verbs. In: *Parallel distributed processing: Explorations in the microstructure of cognition, vol. 2,* ed. J. L. McClelland, D. E. Rumelhart & the PDP Research Group. MIT Press. [JLM]

(1986b) PDP models and general issues in cognitive science. In: *Parallel distributed processing: Explorations in the microstructure of cognition: Foundations, vol. 1,* ed. J. L. McClelland, D. E. Rumelhart & The PDP Research Group. MIT Press. [aJRA]

Rumelhart, D. E., McClelland, J. L. & the PDP Research Group (1986) *Parallel distributed processing: Explorations in the microstructure of cognition: Vol. I: Foundations; Vol. II: Psychological and biological models.* MIT Press. [JLM, AR, RS]

Salvucci, D. D. (2001) Predicting the effects of in-car interface use on driver performance: An integrated model approach. *International Journal of Human-Computer Studies* 55:85–107. [rJRA]

Salvucci, D. D. & Anderson, J. R. (2001) Integrating analogical mapping and general problem solving: The path-mapping theory. *Cognitive Science* 25:67–110. [aJRA]

Sanchez-Montanes, M. A., König, P. & Verschure, P. F. M. J. (2002) Learning sensory maps with real-world stimuli in real time using a biophysically realistic learning rule. *IEEE Transactions on Neural Networks* 13:619–32. [PFMJV]

Sanner, S., Anderson J. R., Lebiere C. & Lovett, M. (2000) Achieving efficient and cognitively plausible learning in backgammon. In: *Proceedings of the Seventeenth International Conference on Machine Learning.* Morgan Kaufmann. [aJRA]

Schneider, W. & Oliver, W. L. (1991) An instructable connectionist/control architecture: Using rule-based instructions to accomplish connectionist learning in a human time scale. In: *Architecture for intelligence: The 22nd Carnegie Mellon Symposium on Cognition,* ed. K. VanLehn. Erlbaum. [aJRA]

Schoelles, M. J. (2002) Simulating human users in dynamic environments. Unpublished doctoral dissertation, George Mason University, Fairfax, VA. [WDG]

Schoelles, M. J. & Gray, W. D. (2003) Top-down versus bottom-up control of cognition in a task switching paradigm, *Fifth International Conference on Cognitive Modeling.* Bamberg. [WDG]

Schoppek, W. (2001) The influence of causal interpretation on memory for system states. In: *Proceedings of the 23rd Annual Conference of the Cognitive Science Society,* ed. J. D. Moore & K. Stenning. Erlbaum. [aJRA]

Seidenberg, M. S. & McClelland, J. L. (1989) A distributed, developmental model of word recognition and naming. *Psychological Review* 96:523–68. [JLM]

Sejnowski, T. J. & Rosenberg, C. R. (1987) Parallel networks that learn to pronounce English text. *Complex Systems* 1:145–68. [aJRA]

Shastri, L., Grannes, D., Narayanan, S. & Feldman, J. (2002) A connectionist encoding of parameterized schemas and reactive plans. In: *Hybrid information processing in adaptive autonomous vehicles,* ed. G. K. Kraetzschmar & G. Palm. Springer-Verlag. [RS]

Sherrington, C. (1906) *The integrative action of the nervous system.* Charles Scribner's. [PAMG]

Shirai, Y. & Anderson, R. W. (1995) The acquisition of tense-aspect morphology: A prototype account. *Language* 71:743–62. [JLM]

Siegelmann, H. (1999) *Neural networks and analog computation: Beyond the Turing Limit.* Birkhauser. [YY]

Siegelman, H. T. & Sontag, E. D. (1992) On the computational power of neural nets. In: *Proceedings of the 5th ACM Workshop on Computational Learning Theory.* [aJRA]

Siegler, R. S. (1988) Strategy choice procedures and the development of multiplication skill. *Journal of Experimental Psychology: General* 117:258–75. [aJRA]

Siegler, R. S. & Lemaire, P. (1997) Older and younger adults' strategy choices in multiplication: Testing predictions of ASCM using the choice/no-choice method. *Journal of Experimental Psychology: General* 126(1):71–92. [WDG]

Siegler, R. S. & Stern, E. (1998) Conscious and unconscious strategy discoveries: A microgenetic analysis. *Journal of Experimental Psychology: General* 127(4):377–97. [WDG]

Simon, L., Greenberg, J., Harmon-Jones, E., Solomon, S., Pyszczynski, T., Arndt, J. & Abend, T. (1997) Terror management and cognitive-experiential self-theory: Evidence that terror management occurs in the experiential system. *Personality and Social Psychology* 72(5):1132–46. [IP]

Simon, T. J. (1998) Computational evidence for the foundations of numerical competence. *Developmental Science* 1:71–78. [aJRA]

(submitted) De-mystifying magical object appearance with a theory of the foundations of numerical competence. *Developmental Science.* [aJRA]

Sirois, S. & Mareschal, D. (2002) Computational approaches to infant habituation. *Trends in Cognitive Sciences* 6:293–98. [SS]

Sirois, S. & Shultz, T. R. (1999) Learning, development, and nativism: Connectionist implications. In: *Proceedings of the Twenty-first Annual Conference of the Cognitive Science Society,* ed. M. Hahn & S. C. Stoness. Erlbaum. [SS]

(2003) A connectionist perspective on Piagetian development. In: *Connectionist models of development,* ed. P. Quinlan. Psychology Press. [SS]

Sloman, S. A. (1996) The empirical case for two systems of reasoning. *Psychological Bulletin* 119(1):3–22. [IP]

(1999) Rational versus arational models of thought. In: *The nature of cognition,* ed. R. J. Sternberg. MIT Press. [IP]

Smith, E. R. & DeCoster, J. (2000) Dual-process models in social and cognitive psychology: Conceptual integration and links to underlying memory systems. *Personality and Social Psychology Review* 4(2):108–31. [IP]

Smolensky, P. (1988) On the proper treatment of connectionism. *Behavioral and Brain Sciences* 11:1–74. [aJRA]

Sohn, M.-H. & Anderson, J. R. (2001) Task preparation and task repetition: Two-component model of task switching. *Journal of Experimental Psychology: General* 130:764–78. [EMA]

Sohn, M.-H., Goode, A., Stenger, V. A, Carter, C. S. & Anderson, J. R. (2003). Competition and representation during memory retrieval: Roles of the prefrontal cortex and the posterior parietal cortex, *Proceedings of the National Academy of Sciences USA* 100:7412–17. [rJRA]

Sohn, M.-H., Ursu, S., Anderson, J. R., Stenger, V. A. & Carter, C. S. (2000) The role of prefrontal cortex and posterior parietal cortex in task-switching. *Proceedings of the National Academy of Science* 13:448–53. [aJRA]

Sperber, D. (1997) Intuitive and reflective beliefs. *Mind and Language* 12(1):67–83. [IP]

Squire, L. R. (1992) Memory and the hippocampus: A synthesis from findings with rats, monkeys, and humans. *Psychological Review* 99:195–232. [aJRA]

Suchman, L. A. (1987) *Plans and situated actions: The problem of human-machine communication.* Cambridge University Press. [aJRA]

Sun, R. (1994) *Integrating rules and connectionism for robust commonsense reasoning.* Wiley. [aJRA, IP, RS]

(2002) *Duality of the mind: A bottom-up approach toward cognition.* Erlbaum. [aJRA, IP, RS]

Sun, R. & Bookman, L., eds. (1994) *Computational architectures integrating neural and symbolic processes. A perspective on the state of the art.* Kluwer. [IP, RS]

Sun, R., Merrill, E. & Peterson, T. (2001) From implicit skills to explicit knowledge: A bottom-up model of skill learning. *Cognitive Science* 25(2):203–44. [RS]

Taatgen, N. A. (2001) Extending the past-tense debate: A model of the German plural. *Proceedings of the Twenty-third Annual Conference of the Cognitive Science Society,* pp. 1018–23. Erlbaum. [aJRA]

(2002) A model of individual differences in skill acquisition in the Kanfer-Ackerman air traffic control task. *Cognitive Systems Research* 3:103–12. [aJRA, NAT]

Taatgen, N. & Anderson, J. R. (2002) Why do children learn to say "broke"? A model of learning the past tense without feedback. *Cognition* 86:(2)123–55. [arJRA, JLM, AGBtM, PNP]

Taatgen, N. & Dijkstra, M. (2003) Constraints on generalization: Why are past-tense irregularization errors so rare? *Proceedings of the 25th Annual Conference of the Cognitive Science Society.* Erlbaum. [rJRA]

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

ter Meulen, A. (1995) *Representing time in natural language: The dynamic interpretation of tense and aspect.* Bradford Books. [AGBtM]

(2000) Chronoscopes: The dynamic representation of facts and events. In: *Speaking about events,* ed. J. Higginbotham et al. Oxford University Press. [AGBtM]

Tesauro, G. (2002) Programming backgammon using self-teaching neural nets. *Artificial Intelligence* 134:181–99. [aJRA]

Thagard, P. (1992) *Conceptual revolutions.* Princeton University Press. [CFO]

Thelen, E. & Smith, L. B. (1994) *A dynamic systems approach to the development of cognition and action.* MIT Press. [DS]

Thomas, M. S. C. & Karmiloff-Smith, A. (2002) Are developmental disorders like cases of adult brain damage? Implications from connectionist modelling. *Behavioral and Brain Sciences* 25:727–88. [CFO]

(2003) Modelling language acquisition in atypical phenotypes. *Psychological Review* 110. (in press). [CFO]

Tolman, E. C. (1948) Cognitive maps in rats and men. *Psychological Review* 55:189–208. [MO]

Treisman, A. M. & Gelade, G. (1980) A feature integration theory of attention. *Cognitive Psychology* 12:97–136. [MO]

Turing, A. (1950) Computing machinery and intelligence. *Mind* 49:433–60. [NAT]

Ullman, M. T., Corkin, S., Coppola, M., Hicock, G., Growdon, J. H., Koroshetz, W. J. & Pinker, S. (1997) A neural dissociation within language: Evidence that the

mental dictionary is part of declarative memory and that grammatical rules are processed by the procedural system. *Journal of Cognitive Neuroscience* 9:266–76. [JLM]

Usher, M. & McClelland, J. L. (2001) On the time course of perceptual choice: The leaky competing accumulator model. *Psychological Review* 108:550–92. [JLM]

van Eijck, J. & Kamp, H. (1997) Representing discourse in context. In: *Handbook of logic and language,* ed. J. van Benthem & A. G. B. ter Meulen. Elsevier Science/MIT Press. [AGBtM]

van Rijn, H., Someren, M. & van der Maas, H. (2000) Modeling developmental transitions in ACT-R. Simulating balance scale behavior by symbolic and subsymbolic learning. In: *Proceedings of the 3rd International Conference on Cognitive Modeling*, pp. 226–33. Universal Press. [aJRA]

Vargha-Khadem, F., Watkins, K., Alcock, K., Fletcher, P. & Passingham, R. (1995) Praxic and nonverbal cognitive deficits in a large family with a genetically transmitted speech and language disorder. *Proceedings of the National Academy of Science* 92:930–33. [JLM]

Velmans, M. (1991) Is human information processing conscious? *Behavioral and Brain Sciences* 14(4):651–726. [MO]

Velmans, M., ed. (1996) *The science of consciousness: Psychological, neuropsychological and clinical reviews.* Routledge. [PAMG]

Vere, S. A. (1992) A cognitive process shell. *Behavioral and Brain Sciences* 15:460–61. [aJRA]

Verkuyl, H. (1996) *A theory of aspectuality: The interaction between temporal and atemporal structure.* Cambridge University Press. [AGBtM]

Verschure, P. F. M. J. (1990) Smolensky's theory of mind. *Behavioral and Brain Sciences* 13:407. [PFMJV]

(1992) Taking connectionism seriously: The vague promise of subsymbolism and an alternative. *Proceedings of the 14th Annual Conference of the Cognitive Science Society,* pp. 653–58, Erlbaum. [PFMJV]

(1998) Synthetic epistemology: The acquisition, retention, and expression of knowledge in natural and synthetic systems. In: *Proceedings 1998 IEEE World Conference on Computational Intelligence,* pp. 147–153. IEEE. [PFMJV]

Verschure, P. F. M. J. & Althaus, P. (2003) A real-world rational agent: Unifying old and new AI. *Cognitive Science* 27:561–90. [PFMJV]

Verschure, P. F. M. J., Kröse, B. J. A. & Pfeifer, R. (1992) Distributed adaptive control: The self-organization of structured behavior. *Robotics and Autonomous Systems* 9:181–96.

Verschure, P. F. M. J. & Voegtlin, T. (1998) A bottom-up approach towards the acquisition, retention, and expression of sequential representations: Distributed adaptive control, III. *Neural Networks* 11:1531–49. [PFMJV]

Verschure, P. F. M. J., Wray, J., Sporns, O., Tononi, G. & Edelman, G. M. (1996) Multilevel analysis of classical conditioning in a behaving real world artifact. *Robotics and Autonomous Systems* 16:247–65. [PFMJV]

Voegtlin, T. & Verschure, P. F. M. J. (1999) What can robots tell us about brains? A synthetic approach towards the study of learning and problem solving. *Reviews in the Neurosciences* 10:291–310. [PFMJV]

Waddington, C. H. (1975) *The evolution of an evolutionist.* Edinburgh University Press. [CFO]

Wallach, D. & Lebiere, C. (2000) Learning of event sequences: An architectural approach. In: *Proceedings of the 3rd International Conference on Cognitive Modeling,* ed. N. Taatgen. Universal Press. [aJRA]

(in press) Conscious and unconscious knowledge: Mapping to the symbolic and subsymbolic levels of a hybrid architecture. In: *Attention and implicit learning,* ed. L. Jimenez. John Benjamins. [aJRA]

Weizenbaum, J. (1966) ELIZA – A computer program for the study of natural language communication between man and machine. *Communications of the Association for Computing Machinery* 9:36–45. [NAT]

Wermter, S. & Sun, R., eds. (2000) *Hybrid neural systems* (Lecture Notes in Artificial Intelligence, LNCS 1778). Springer-Verlag. [RS]

Whitehouse, H. (2002) Modes of religiosity: Towards a cognitive explanation of the sociopolitical dynamics of religion. *Method and Theory in the Study of Religion* 14(3–4):293–315. [IP]

Wise, S. P., Murray, E. A. & Gerfen, C. R. (1996) The frontal cortex–basal ganglia system in primates. *Critical Reviews in Neurobiology* 10:317–56. [aJRA]

Woolf, N. J. & Hameroff, S. R. (2001) A quantum approach to visual consciousness. *Trends in Cognitive Sciences 5*(11):472–78. [HW]

Yang, Y. & Bringsjord, S. (forthcoming) *Mental metalogic: A new, unifying theory of human and machine reasoning.* Erlbaum. [YY]

Cognitive Architectures: Research Issues and Challenges

Pat Langley

Computational Learning Laboratory
Center for the Study of Language and Information
Stanford University, Stanford, CA 94305 USA

John E. Laird

EECS Department, The University of Michigan
1101 Beal Avenue, Ann Arbor, MI 48109 USA

Seth Rogers

Computational Learning Laboratory
Center for the Study of Language and Information
Stanford University, Stanford, CA 94305 USA

Abstract

In this paper, we examine the motivations for research on cognitive architectures and review some candidates that have been explored in the literature. After this, we consider the capabilities that a cognitive architecture should support, some properties that it should exhibit related to representation, organization, performance, and learning, and some criteria for evaluating such architectures at the systems level. In closing, we discuss some open issues that should drive future research in this important area.

Key words: cognitive architectures, intelligent systems, cognitive processes

1 Background and Motivation

A cognitive architecture specifies the underlying infrastructure for an intelligent system. Briefly, an architecture includes those aspects of a cognitive agent that are constant over time and across different application domains. These typically include:

- the short-term and long-term memories that store content about the agent's beliefs, goals, and knowledge;

- the representation of elements that are contained in these memories and their organization into larger-scale mental structures;

- the functional processes that operate on these structures, including the performance mechanisms that utilize them and the learning mechanisms that alter them.

Because the contents of an agent's memories can change over time, one would not consider the knowledge and beliefs encoded therein to be part of that agent's architecture. Just as diþerent programs can run on the same computer architecture, so diþerent knowledge bases and beliefs can be interpreted by the same cognitive architecture. There is also a direct analogy with a building's architecture, which consists of permanent features like its foundation, roof, and rooms, rather than its furniture and appliances, which one can move or replace.

As we will see, alternative cognitive architectures can diþer in the speciþc assumptions they make about these issues, just as distinct buildings diþer in their layouts. In addition to making diþerent commitments about how to represent, use, or acquire knowledge and beliefs, alternative frameworks may claim that more or less is built into the architectural level, just as some buildings embed shelves and closets into their þxed structures, whereas others handle the same functions with replaceable furniture.

Research on cognitive architectures is important because it supports a central goal of artiþcial intelligence and cognitive science: the creation and understanding of synthetic agents that support the same capabilities as humans. Some work focuses on modeling the invariant aspects of human cognition, whereas other eþorts view architectures as an eþective path to building intelligent agents. However, these are not antithetical goals; cognitive psychology and AI have a long history of building on the other's ideas (Langley, 2006), and research on cognitive architectures has played a key role in this beneþcial interchange.

In some ways, cognitive architectures constitute the antithesis of expert systems, which provide skilled behavior in narrowly deþned contexts. In contrast, architectural research aims for breadth of coverage across a diverse set of tasks and domains. More important, it oþers accounts of intelligent behavior at the systems level, rather than at the level of component methods designed for specialized tasks. Newell (1973a) has argued persuasively for systems-level research in cognitive science and artiþcial intelligence, claiming \You can't play 20 questions with nature and win". Instead of carrying out micro-studies that

address only one issue at a time, we should attempt to unify many findings into a single theoretical framework, then proceed to test and refine that theory.

Since that call to arms, there has been a steady flow of research on cognitive architectures. The movement was associated originally with a specific class of architectures known as production systems (Newell, 1973b; Neches et al., 1987) and emphasized explanation of psychological phenomena, with many current candidates still taking this form and showing similar concerns. However, over the past three decades, a variety of other architectural classes have emerged, some less concerned with human behavior, that make quite different assumptions about the representation, organization, utilization, and acquisition of knowledge. At least three invited symposia have brought together researchers in this area (Laird, 1991; VanLehn, 1991; Shapiro & Langley, 2004), and there have been at least two edited volumes (Sun, 2005; VanLehn, 1991). The movement has gone beyond basic research into the commercial sector, with applications to believable agents for simulated training environments (e.g., Tambe et al., 1995), computer tutoring systems (Koedinger, Anderson, Hadley, & Mark, 1997), and interactive computer games (e.g., Magerko et al., 2004).

Despite this progress, there remains a need for additional research in the area of cognitive architectures. As artificial intelligence and cognitive science have matured, they have fragmented into a number of well-defined subdisciplines, each with its own goals and its own criteria for evaluation. Yet commercial and government applications increasingly require integrated systems that exhibit intelligent behavior, not just improvements to the components of such systems. This demand can be met by an increased focus on system-level architectures that support complex cognitive behavior across a broad range of relevant tasks.

In this document, we examine some key issues that arise in the design and study of integrated cognitive architectures. Because we cannot hope to survey the entire space of architectural theories, we focus on the ability to generate intelligent behavior, rather than matching the results of psychological experiments.[1] We begin with a brief review of some sample architectures, then discuss the capabilities and functions that such systems should support. After this, we consider a number of design decisions that relate to the properties of cognitive architectures, followed by some dimensions along which one should evaluate them. In closing, we note some open issues in the area and propose some directions that future research should take to address them.

[1] Sun (2007) provides another treatment of cognitive architectures that discusses the second topic in greater detail.

2 Example Cognitive Architectures

Before turning to abstract issues that arise in research on cognitive architectures, we should consider some concrete examples that have been reported in the literature. Here we review four distinct frameworks that fall at diÞerent points within the architectural space. We have selected these architectures because each has appeared with reasonable frequency in the literature, and also because they exhibit diÞerent degrees of concern with explaining human behavior. We have ordered them along this dimension, with more devoted psychological models coming earlier.

In each case, we discuss the manner in which the architecture represents, organizes, utilizes, and acquires knowledge, along with its accomplishments. Because we review only a small sample of extant architectures, we summarize a variety of other frameworks brie y in the Appendix. Nevertheless, this set should give readers some intuitions about the space of cognitive architectures, which later sections of the paper discuss more explicitly.

One common feature of the architectures we examine is that, although they have some theoretical commitment to parallelism, especially in memory retrieval, they also rely on one or a few decision modules. We have not included connectionist approaches in our treatment because, to our knowledge, they have not demonstrated the broad functionality associated with cognitive architectures in the sense we discuss here. However, they have on occasion served as important components in larger-scale architectures, as in Sun, Merrill, and Peterson's (2001) CLARION framework.

2.1 ACT

ACT-R (Anderson & Lebiere, 1998, Anderson et al., 2004) is the latest in a family of cognitive architectures, concerned primarily with modeling human behavior, that has seen continuous development since the late 1970s. ACT-R 6 is organized into a set of modules, each of which processes a diÞerent type of information. These include sensory modules for visual processing, motor modules for action, an intentional module for goals, and a declarative module for long-term declarative knowledge. Each module has an associated buÞer that holds a relational declarative structure (often called `chunks', but diÞerent from those in Soar). Taken together, these buÞers comprise ACT-R's short-term memory.

A long-term production memory coordinates the processing of the modules. The conditions of each production test chunks in the short-term buÞers, whereas its actions alter the buÞers upon application. Some changes modify

existing structures, whereas others initiate actions in the associated modules, such as executing a motor command or retrieving a chunk from long-term declarative memory. Each declarative chunk has an associated base activation that re ects its past usage and in uences its retrieval from long-term memory, whereas each production has an expected cost (in terms of time needed to achieve goals) and probability of success.

On every cycle, ACT determines which productions match against the contents of short-term memory. This retrieval process is in uenced by the base activation for each chunk it matches. ACT computes the utility for each matched production as the diþerence between its expected beneþt (the desirability of its goal times its probability of success) and its expected cost. The system selects the production with the highest utility (after adding noise to this score) and executes its actions. The new situation leads new productions to match and þre, so that the cycle continues.

Learning occurs in ACT-R at both the structural and statistical levels. For instance, the base activation for declarative chunks increases with use by productions but decays otherwise, whereas the cost and success probability for productions is updated based on their observed behavior. The architecture can learn entirely new rules from sample solutions through a process of production compilation that analyzes dependencies of multiple rule þrings, replaces constants with variables, and combines them into new conditions and actions (Taatgen, 2005).

The ACT-R community has used its architecture to model a variety of phenomena from the experimental psychology literature, including aspects of memory, attention, reasoning, problem solving, and language processing. Most publications have reported accurate þts to quantiative data about human reaction times and error rates. More recently, Anderson (2007) has related ACT-R modules to diþerent areas of the brain and developed models that match results from brain-imaging studies. One the more applied front, the framework has played a central role in tutoring systems that have seen wide use in schools (Koedinger et al., 1997), and it has also been used to control mobile robots that interact with humans (Trafton et al., 2005).

2.2 Soar

Soar (Laird, 2008; Laird, Newell, & Rosenbloom, 1987; Newell, 1990) is a cognitive architecture that has been under continuous development since the early 1980s. Procedural long-term knowledge in Soar takes the form of production rules, which are in turn organized in terms of operators associated with problem spaces. Some operators describe simple, primitive actions that mod-

ify the agent's internal state or generate primitive external actions, whereas others describe more abstract activities. For many years, Soar represented all long-term knowledge in this form, but recently separate episodic and semantic memories have been added. The episodic memory (Nuxoll & Laird, 2007) holds a history of previous states, while semantic memory contains previously known facts.

All tasks in Soar are formulated as attempts to achieve goals. Operators perform the basic deliberative acts of the system, with knowledge used to dynamically determine their selection and application. The basic processing cycle repeatedly proposes, selects, and applies operators of the current problem space to a problem state, moving ahead one decision at a time. However, when knowledge about operator selection is insuŽcient to determine the next operator to apply or when an abstract operator cannot be implemented, an impasse occurs; in response, Soar creates a new goal to determine which operator it should select or how it should implement the abstract operator.

This process can lead to the dynamic generation of a goal hierarchy, in that problems are decomposed into subproblems as necessary. The `state' of a speciþc goal includes all features of its supergoals, plus any additional cognitive structures necessary to select and apply operators in the subgoal. Processing in a subgoal involves the same basic processing cycle of selecting and applying operators. Subgoals can also deliberately access episodic or semantic memory to retrieve knowledge relevant to resolving the impasse. The top state includes all sensor data obtained from the external environment, so this information is also available to all subgoals. On any cycle, the states at various levels of the goal hierarchy can change, typically due to changes in sensor values or as the result of operator applications in subgoals. When the system resolves the impasse that generated a goal, that goal disappears, along with all the subgoals below it.

Soar has multiple learning mechanisms for diÞerent types of knowledge: chunking and reinforcement learning acquire procedural knowledge, whereas episodic and semantic learning acquire their corresponding types of declarative knowledge. Chunking occurs when one or more result is produced in a subgoal (Laird, Rosenbloom, & Newell, 1986). When this happens, Soar learns a new chunk, represented as a production rule which summarizes the processing that generated the results. A chunk's actions are based on the results, whereas its conditions are based on those aspects of the goals above the subgoal that were relevant to determining the results. Once the agent has learned a chunk, it þres in new situations that are similar along relevant dimensions, often giving the required results directly and thus avoiding the impasse that led to its formation. Reinforcement learning adjusts numeric values associated with rules that help select operators (Nason & Laird, 2004). Episodic learning records the contents of working memory in snapshots, while semantic learning stores

individual elements of working memory for later retrieval.

Researchers have used Soar to develop a variety of sophisticated agents that have demonstrated impressive functionality. The most visible has been TAC-Air-Soar (Tambe et al., 1995), which modeled þghter pilots in military training exercises that involved air combat scenarios. More recently, Soar has supported a number of intelligent agents that control synthetic characters in interactive computer games (Margerko et al., 2004). Another thrust has involved using Soar to model the details of human language processing (Lewis, 1993), categorization (Miller & Laird, 1996), and other facets of cognition, but the emphasis has been on demonstrating high-level functionality rather than on þts to quantitative measurements.

2.3 ICARUS

Icarus is a more recent architecture (Langley, Cummings, & Shapiro, 2004) that stores two distinct forms of knowledge. Concepts describe classes of environmental situations in terms of other concepts and percepts, whereas skills specify how to achieve goals by decomposing them into ordered subgoals. Both concepts and skills involve relations among objects, and both impose a hierarchical organization on long-term memory, with the former grounded in perceptions and the latter in executable actions. Moreover, skills refer to concepts in their heads, their initiation conditions, and their continuation conditions.

The basic Icarus interpreter operates on a recognize-act cycle. On each step, the architecture deposits descriptions of visible objects into a perceptual buÞfer. The system compares primitive concepts to these percepts and adds matched instances to short-memory as beliefs. These in turn trigger matches of higher-level concepts, with the process continuing until Icarus infers all deductively implied beliefs. Next, starting from a top-level goal, it þnds a path downward through the skill hierarchy in which each subskill has satisþed conditions but an unsatisþed goal. When a path terminates in a primitive skill with executable actions, the architecture applies these actions to aÞect the environment. This leads to new percepts, changes in beliefs, and reactive execution of additional skill paths to achieve the agent's goals.

However, when Icarus can þnd no applicable path through the skill hierarchy that is relevant to a top-level goal, it resorts to problem solving using a variant of means-ends analysis. This module chains backward oÞ either a skill that would achieve the current goal or oÞ the goal concept's deþnition, and it interleaves problem solving with execution in that it carries out selected skills when they become applicable. Whenever problem solving achieves a goal, Icarus creates a new skill with that goal as its head and with one or more ordered subgoals that are based on the problem solution. If the system encounters

similar problems in the future, it executes the learned skills to handle them reactively, without need for deliberative problem solving (Langley & Choi, 2006b).

Researchers have used Icarus to develop agents for a number of domains that involve a combination of inference, execution, problem solving, and learning. These have included tasks like the Tower of Hanoi, multi-column subtraction, FreeCell solitaire, and logistics planning. They have also used the architecture to control synthetic characters in simulated virtual environments, including ones that involve urban driving (Langley & Choi, 2006a) and first-person shooter scenarios (Choi et al., 2007). Ongoing work aims to link Icarus to physical robots that carry out joint activities with humans.

2.4 PRODIGY

Prodigy (Carbonell, Knoblock, & Minton, 1990) is another cognitive architecture that saw extensive development from the middle 1980s to the late 1990s. This framework incorporates two main kinds of knowledge. Domain rules encode the conditions under which actions have certain effects, where the latter are described as the addition or deletion of first-order expressions. These refer both to physical actions that affect the environment and to inference rules, which are purely cognitive. In contrast, control rules specify the conditions under which the architecture should select, reject, or prefer a given operator, set of operator bindings, problem state, or goal during the search process.

As in Icarus, Prodigy's basic problem-solving module involves search through a problem space to achieve one or more goals, which it also casts as first-order expressions. This search relies on means-ends analysis, which selects an operator that reduces some difference between the current state and the goal, which in turn can lead to subproblems with their own current states and goals. On each cycle, Prodigy uses its control rules to select an operator, binding set, state, or goal, to reject them out of hand, or to prefer some over others. In the absence of such control knowledge, the architecture makes choices at random and carries out depth-first means-ends search with backtracking.

Prodigy's explanation-based learning module constructs control rules based on its problem-solving experience (Minton, 1990). Successful achievement of a goal after search leads to creation of selection or preference rules related to that goal and to the operators whose application achieved it. Failure to achieve a goal leads to creation of rejection or preference rules for operators, goals, and states that did not produce a solution. To generate these control rules, Prodigy invokes a learning method that analyzes problem-solving traces and proves the reasons for success or failure. The architecture also collects statistics

on learned rules and retains only those whose inclusion, over time, leads to more eŽcient problem solving.

In addition, Prodigy includes separate modules for controlling search by analogy with earlier solutions (Veloso & Carbonell, 1993), learning operator descriptions from observed solutions or experimentation (Wang, 1995), and improving the quality of solutions (Perez & Carbonell, 1994). Although most research in this framework has dealt exclusively with planning and problem solving, Prodigy also formed the basis for an impressive system that interleaved planning and execution for a mobile robot that accepted asychronous requests from users (Haigh & Veloso, 1996).

3 Capabilities of Cognitive Architectures

Any intelligent system is designed to engage in certain activities that, taken together, constitute its functional capabilities. In this section, we discuss the varied capabilities that a cognitive architecture can support. Although only a few abilities, such as recognition and decision making, are strictly required to support a well-deþned architecture, the entire set seems required to cover the full range of human-level intelligent activities.

A central issue that confronts the designer of a cognitive architecture is how to let agents access diþerent sources of knowledge. Many of the capabilities we discuss below give the agent access to such knowledge. For example, knowledge about the environment comes through perception, knowledge about implications of the current situation comes through planning, reasoning, and prediction, knowledge from other agents comes via communication, and knowledge from the past comes through remembering and learning. The more such capabilities an architecture supports, the more sources of knowledge it can access to inform its behavior.

Another key question is whether the cognitive architecture supports a capability directly, using embedded processes, or whether it instead provides ways to implement that capability in terms of knowledge. Design decisions of this sort in uence what the agent can learn from experience, what the designers can optimize at the outset, and what functionalities can rely on specialized representations and mechanisms. In this section, we attempt to describe functionality without referring to the underlying mechanisms that implement them, but this is an important issue that deserves more attention in the future.

3.1 Recognition and Categorization

An intelligent agent must make some contact between its environment and its knowledge. This requires the ability to recognize situations or events as instances of known or familiar patterns. For example, a reader must recognize letters and the words they make up, a chess player must identify meaningful board conþgurations, and an image analyst must detect buildings and vehicles in aerial photographs. However, recognition need not be limited to static situations. A fencing master can identify diÞerent types of attacks and a football coach can recognize the execution of particular plays by the opposing team, both of which involve dynamic events.

Recognition is closely related to categorization, which involves the assignment of objects, situations, and events to known concepts or categories. However, research on cognitive architectures typically assumes recognition is a primitive process that occurs on a single cycle and that underlies many higher-level functions, whereas categorization is sometimes viewed as a higher-level function. Recognition and categorization are closely linked to perception, in that they often operate on output from the perceptual system, and some frameworks view them as indistinguishable. However, they can both operate on abstract mental structures, including those generated internally, so we will treat them as distinct.

To support recognition and categorization, a cognitive architecture must provide some way to represent patterns and situations in memory. Because these patterns must apply to similar but distinct situations, they must encode general relations that hold across these situations. An architecture must also include some recognition process that lets it identify when a particular situation matches a stored pattern or category and, possibly, measure the degree to which it matches. In production system architectures, this mechanism determines when the conditions of each production rule match and the particular ways they are instantiated. Finally, a complete architecture should include some means to learn new patterns or categories from instruction or experience, and to reþne existing patterns when appropriate.

3.2 Decision Making and Choice

To operate in an environment, an intelligent system also requires the ability to make decisions and select among alternatives. For instance, a student must decide which operation will simplify an integration problem, a speaker must select what word to use next in an utterance, and a baseball player must decide whether or not to swing at a pitch. Such decisions are often associated with the recognition of a situation or pattern, and most cognitive architec-

tures combine the two mechanisms in a recognize-act cycle that underlies all cognitive behavior.

Such one-step decision making has much in common with higher-level choice, but diþers in its complexity. For example, consider a consumer deciding which brand of detergent to buy, a driver choosing which route to drive, and a general selecting which target to bomb. Each of these decisions can be quite complex, depending on how much time and energy the person is willing to devote. Thus, we should distinguish between decisions that are made at the architectural level and more complex ones that the architecture enables.

To support decision making, a cognitive architecture must provide some way to represent alternative choices or actions, whether these are internal cognitive operations or external ones. It must also oþer some process for selecting among these alternatives, which most architectures separate into two steps. The þrst determines whether a given choice or action is allowable, typically by associating it with some pattern and considering it only if the pattern is matched. For instance, we can specify the conditions under which a chess move is legal, then consider that move only when the conditions are met. The second step selects among allowable alternatives, often by computing some numeric score and choosing one or more with better scores. Such con ict resolution takes quite diþerent forms in diþerent architectures.

Finally, an ideal cognitive architecture should incorporate some way to improve its decisions through learning. Although this can, in principle, involve learning new alternatives, most mechanisms focus on learning or revising either the conditions under which an existing action is considered allowable or altering the numeric functions used during the con ict resolution stage. The resulting improvements in decision making will then be re ected in the agent's overall behavior.

3.3 Perception and Situation Assessment

Cognition does not occur in isolation; an intelligent agent exists in the context of some external environment that it must sense, perceive, and interpret. An agent may sense the world through diþerent modalities, just as a human has access to sight, hearing, and touch. The sensors may range from simple devices like a thermometer, which generates a single continuous value, to more complex mechanisms like stereoscopic vision or sonar that generate a depth map for the local environment within the agent's þeld of view. Perception can also involve the integration of results from diþerent modalities into a single assessment or description of the environmental situation, which an architecture can represent for utilization by other cognitive processes.

Perception is a broad term that covers many types of processing, from inexpensive ones that an architecture can support automatically to ones that require limited resources and so must be invoked through conscious intentions. For example, the human visual system can detect motion in the periphery without special eÞort, but the fovea can extract details only from the small region at which it is pointed. A cognitive architecture that includes the second form of sensor must confront the issue of attention, that is, deciding how to allocate and direct its limited perceptual resources to detect relevant information in a complex environment.

An architecture that supports perception should also deal with the issue that sensors are often noisy and provide at most an inaccurate and partial picture of the agent's surroundings. Dynamic environments further complicate matters in that the agent must track changes that sometimes occur at a rapid rate. These challenges can be oÞset with perceptual knowledge about what sensors to invoke, where and when to focus them, and what inferences are plausible. An architecture can also acquire and improve this knowledge by learning from previous perceptual experiences.

An intelligent agent should also be able to move beyond perception of isolated objects and events to understand and interpret the broader environmental situation. For example, a þre control oŽcer on a ship must understand the location, severity, and trajectory of þres in order to respond eÞectively, whereas a general must be aware of an enemy's encampments, numbers, and resources to defend against them successfully. Thus, situation assessment requires an intelligent agent to combine perceptual information about many entities and events, possibly obtained from many sources, to compose a large-scale model of the current environment. As such, it relies both on the recognition and categorization of familiar patterns in the environment, which we discussed earlier, and on inferential mechanisms, which we will consider shortly.

3.4 Prediction and Monitoring

Cognitive architectures exist over time, which means they can beneþt from an ability to predict future situations and events accurately. For example, a good driver knows approximately when his car will run out of gas, a successful student can predict how much he must study to ace a þnal, and a skilled pilot can judge how close he can y to the ground without crashing. Perfect prediction may not be possible in many situations, but perfection is seldom necessary to make predictions that are useful to an intelligent system.

Prediction requires some model of the environment and the eÞect actions have on it, and the architecture must represent this model in memory. One general approach involves storing some mapping from a description of the current

situation and an action onto a description of the resulting situation. Another approach encodes the eÞects of actions or events in terms of changes to the environment. In either case, the architecture also requires some mechanism that uses these knowledge structures to predict future situations, say by recognizing a class of situations in which an action will have certain eÞects. An ideal architecture should also include the ability to learn predictive models from experience and to reþne them over time.

Once an architecture has a mechanism for making predictions, it can also utilize them to monitor the environment. For example, a pilot may suspect that his tank has a leak if the fuel gauge goes down more rapidly than usual, and a commander may suspect enemy action if a reconnaissance team fails to report on time. Because monitoring relates sensing to prediction, it raises issues of attentional focus when an architecture has limited perceptual resources. Monitoring also provides natural support for learning, since errors can help an agent improve its model of the environment.

3.5 Problem Solving and Planning

Because intelligent systems must achieve their goals in novel situations, the cognitive architectures that support them must be able to generate plans and solve problems. For example, an unmanned air vehicle beneþts from having a sensible ight plan, a project manager desires a schedule that allocates tasks to speciþc people at speciþc times, and a general seldom moves into enemy territory without at least an abstract course of action. When executed, plans often go awry, but that does not make them any less useful to an intelligent agent's thinking about the future.

Planning is only possible when the agent has an environmental model that predicts the eÞects of its actions. To support planning, a cognitive architecture must be able to represent a plan as an (at least partially) ordered set of actions, their expected eÞects, and the manner in which these eÞects enable later actions. The plan need not be complete to guide behavior, in that it may extend only a short time into the future or refer to abstract actions that can be expanded in diÞerent ways. The structure may also include conditional actions and branches that depend on the outcome of earlier events as noted by the agent.

An intelligent agent should also be able to construct a plan from components available in memory. These components may refer to low-level motor and sensory actions but, often, they will be more abstract structures, including prestored subplans. There exist many mechanisms for generating plans from components, as well as ones for adapting plans that have been retrieved from

memory. What these methods have in common is that they involve problem solving or search. That is, they carry out steps through a space of problem states, on each step considering applicable operators, selecting one or more operator, and applying it to produce a new problem state. This search process continues until the system has found an acceptable plan or decides to give up.

The notion of problem solving is somewhat more general than planning, though they are typically viewed as closely related. In particular, planning usually refers to cognitive activities within the agent's head, whereas problem solving can also occur in the world. Especially when a situation is complex and the architecture has memory limitations, an agent may carry out search by applying operators or actions in the environment, rather than trying to construct a plan internally. Problem solving can also rely on a mixture of internal planning and external behavior, but it generally involves the multi-step construction of a problem solution. Like planning, problem solving is often characterized in terms of search through a problem space that applies operators to generate new states, selects promising candidates, and continues until reaching a recognized goal.

Planning and problem solving can also beneþt from learning. Naturally, improved predictive models for actions can lead to more eÞective plans, but learning can also occur at the level of problem space search, whether this activity takes place in the agent's head or in the physical world. Such learning can rely on a variety of information sources. In addition to learning from direct instruction, an architecture can learn from the results of problem-space search (Sleeman et al., 1982), by observing another agent's behavior or behavioral cloning (Sammut, 1996), and from delayed rewards via reinforcement learning (Sutton & Barto, 1998). Learning can aim to improve problem solving behavior in two ways (Langley, 1995a). One focuses on reducing the branching factor of search, either through adding heuristic conditions to problem space operators or reþning a numeric evaluation function to guide choice. Another focuses on forming macro-operators or stored plans that reduce the eÞective depth of search by taking larger steps in the problem space.

Intelligent agents that operate in and monitor dynamic environments must often modify existing plans in response to unanticipated changes. This can occur in several contexts. For instance, an agent should update its plan when it detects a changed situation that makes some planned activities inapplicable, and thus requires other actions. Another context occurs when a new situation suggests some more desirable way of accomplishing the agent's goal; such opportunistic planning can take advantage of these unexpected changes. Monitoring a plan's execution can also lead to revised estimates about the plan's eÞectiveness, and, ultimately, to a decision to pursue some other course of action with greater potential. Replanning can draw on the same mechanisms as generating a plan from scratch, but requires additional operators for

removing actions or replacing them with other steps. Similar methods can also adapt to the current situation a known plan the agent has retrieved from memory.

3.6 Reasoning and Belief Maintenance

Problem solving is closely related to reasoning, another central cognitive activity that lets an agent augment its knowledge state. Whereas planning is concerned primarily with achieving objectives in the world by taking actions, reasoning draws mental conclusions from other beliefs or assumptions that the agent already holds. For example, a pilot might conclude that, if another plane changes its course to intersect his own, it is probably an enemy þghter. Similarly, a geometry student might deduce that two triangles are congruent because they share certain sides and vertices, and a general might infer that, since he has received no recent reports of enemy movement, a nearby opposing force is still camped where it was the day before.

To support such reasoning, a cognitive architecture must þrst be able to represent relationships among beliefs. A common formalism for encoding such relationships is þrst-order logic, but many other notations have also been used, ranging from production rules to neural networks to Bayesian networks. The relations represented in this manner may be logically or probabilistically sound, but this is not required; knowledge about reasoning can also be heuristic or approximate and still prove quite useful to an intelligent agent. Equally important, the formalism may be more or less expressive (e.g., limited to propositional logic) or computationally eŽcient.

Naturally, a cognitive architecture also requires mechanisms that draw inferences using these knowledge structures. Deductive reasoning is an important and widely studied form of inference that lets one combine general and speciþc beliefs to conclude others that they entail logically. However, an agent can also engage in inductive reasoning, which moves from speciþc beliefs to more general ones and which can be viewed as a form of learning. An architecture may also support abductive inference, which combines general knowledge and speciþc beliefs to hypothesize other speciþc beliefs, as occurs in medical diagnosis. In constrained situations, an agent can simply draw all conclusions that follow from its knowledge base, but more often it must select which inferential knowledge to apply. This raises issues of search closely akin to those in planning tasks, along with issues of learning to make that search more eÞective.

Reasoning plays an important role not only when inferring new beliefs but when deciding whether to maintain existing ones. To the extent that cer-

tain beliefs depend on others, an agent should track the latter to determine whether it should continue to believe the former, abandon it, or otherwise alter its conþdence. Such belief maintenance is especially important for dynamic environments in which situations may change in unexpected ways, with implications for the agent's behavior. One general response to this issue involves maintaining dependency structures in memory that connect beliefs, which the architecture can use to propagate changes as they occur.

3.7 Execution and Action

Cognition occurs to support and drive activity in the environment. To this end, a cognitive architecture must be able to represent and store motor skills that enable such activity. For example, a mobile ground robot or unmanned air vehicle should have skills or policies for navigating from one place to another, for manipulating its surroundings with eÞectors, and for coordinating its behavior with other agents on its team. These may be encoded solely in terms of primitive or component actions, but they may also specify more complex multi-step skills or procedures. The latter may take the form of plans that the agent has generated or retrieved from memory, especially in architectures that have grown out of work on problem solving and planning. However, other formulations of motor skill execution, such as closed-loop controllers, have also been explored.

A cognitive architecture must also be able to execute skills and actions in the environment. In some frameworks, this happens in a completely reactive manner, with the agent selecting one or more primitive actions on each decision cycle, executing them, and repeating the process on the next cycle. This approach is associated with closed-loop strategies for execution, since the agent can also sense the environment on each time step. The utilization of more complex skills supports open-loop execution, in which the agent calls upon a stored procedure across many cycles without checking the environment. However, a exible architecture should support the entire continuum from fully reactive, closed-loop behavior to automatized, open-loop behavior, as can humans.

Ideally, a cognitive architecture should also be able learn about skills and execution policies from instruction and experience. Such learning can take diÞerent forms, many of which parallel those that arise in planning and problem solving. For example, an agent can learn by observing another agent's behavior, by successfully achieving its goals, and from delayed rewards. Similarly, it can learn or reþne its knowledge for selecting primitive actions, either in terms of heuristic conditions on their application or as a numeric evaluation function that re ects their utility. Alternatively, an agent can acquire or revise more complex skills in terms of known skills or actions.

3.8 Interaction and Communication

Sometimes the most eÞective way for an agent to obtain knowledge is from another agent, making communication another important ability that an architecture should support. For example, a commander may give orders to, and receive reports from, her subordinates, while a shopper in a ea market may dicker about an item's price with its owner. Similarly, a traveler may ask and receive directions on a street corner, while an attorney may query a defendant about where he was on a particular night. Agents exist in environments with other agents, and there are many occasions in which they must transfer knowledge from one to another.

Whatever the modality through which this occurs, a communicating agent must represent the knowledge that it aims to convey or that it believes another agent intends for it. The content so transferred can involve any of the cognitive activities we have discussed so far. Thus, two agents can communicate about categories recognized and decisions made, about perceptions and actions, about predictions and anomalies, and about plans and inferences. One natural approach is to draw on the representations that result from these activities as the input to, and the output from, interagent communication.

A cognitive architecture should also support mechanisms for transforming knowledge into the form and medium through which it will be communicated. The most common form is spoken or written language, which follows established conventions for semantics, syntax, and pragmatics onto which an agent must map the content it wants to convey. Even when entities communicate with purely artiþcial languages, they do not have exactly the same mental structures and they must translate content into some external format. One can view language generation as a form of planning and execution, whereas language understanding involves inference and reasoning. However, the specialized nature of language processing makes these views misleading, since the task raises many additional issues.

An important form of communication occurs in conversational dialogues, which require both generation and understanding of natural language, as well as coordination with the other agent in the form of turn taking. Learning is also an important issue in language and other forms of communication, since an architecture should be able to acquire syntactic and semantic knowledge for use at both the sentence and dialogue levels. Moreover, some communicative tasks, like question answering, require access to memory for past events and cognitive activities, which in turn beneþts from episodic storage.

3.9 Remembering, Re ection, and Learning

A cognitive architecture can also beneþt from capabilities that cut across those described in the previous sections, in that they operate on mental structures produced or utilized by them. Such abilities, which Sloman (2001) refers to as metamanagement mechanisms, are not strictly required for an intelligent agent, but their inclusion can extend considerably the exibility and robustness of an architecture.

One capacity of this sort involves remembering { the ability to encode and store the results of cognitive processing in memory and to retrieve or access them later. An agent cannot directly remember external situations or its own physical actions; it can only recall cognitive structures that describe those events or inferences about them. This idea extends naturally to memories of problem solving, reasoning, and communication. To remember any cognitive activity, the architecture must store the cognitive structures generated during that activity, index them in memory, and retrieve them when needed. The resulting content is often referred to as episodic memories.

Another capability that requires access to traces of cognitive activity is re ection. This may involve processing of either recent mental structures that are still available or older structures that the agent must retrieve from its episodic store. One type of re ective activity concerns the justiþcation or explanation of an agent's inferences, plans, decisions, or actions in terms of cognitive steps that led to them. Another revolves around meta-reasoning about other cognitive activities, which an architecture can apply to the same areas as explanation, but which emphasizes their generation (e.g., forming inferences or making plans) rather than their justiþcation. To the extent that re ective processes lay down their own cognitive traces, they may themselves be subject to re ection. However, an architecture can also support re ection through less transparent mechanisms, such as statistical analyses, that are not themselves inspectable by the agent.

A þnal important ability that applies to many cognitive activities is learning. We have discussed previously the various forms this can take, in the context of diÞerent architectural capacities, but we should also consider broader issues. Learning usually involves generalization beyond speciþc beliefs and events. Although most architectures carry out this generalization at storage time and enter generalized knowledge structures in memory, some learning mechanisms store speciþc situations and generalization occurs at retrieval time through analogical or case-based reasoning. Either approach can lead to diÞerent degrees of generalization or transfer, ranging from very similar tasks, to other tasks within the same domain, and even to tasks within related but distinct domains. Many architectures treat learning as an automatic process that is

not subject to inspection or conscious control, but they can also use meta-reasoning to support learning in a more deliberate manner. The data on which learning operates may come from many sources, including observation of another agent, an agent's own problem solving behavior, or practice of known skills. But whatever the source of experience, all involve processing of memory structures to improve the agent's capabilities.

4 Properties of Cognitive Architectures

We can also characterize cognitive architectures in terms of the internal properties that produce the capabilities described in the previous section. These divide naturally into the architecture's representation of knowledge, the organization it places on that knowledge, the manner in which the system utilizes its knowledge, and the mechanisms that support acquisition and revision of knowledge through learning. Below we consider a number of design decisions that arise within each of these facets of an intelligent system, casting them in terms of the data structures and algorithms that are supported at the architectural level. Although we present most issues in terms of oppositions, many of the alternatives we discuss are complementary and can exist within the same framework.

4.1 Representation of Knowledge

One important class of architectural properties revolves around the representation of knowledge. Recall that knowledge itself is not built into an architecture, in that it can change across domains and over time. However, the representational formalism in which an agent encodes its knowledge constitutes a central aspect of a cognitive architecture.

Perhaps the most basic representational choice involves whether an architecture commits to a single, uniform notation for encoding its knowledge or whether it employs a mixture of formalisms. Selecting a single formalism has advantages of simplicity and elegance, and it may support more easily abilities like learning and re ection, since they must operate on only one type of structure. However, as we discuss below, diPerent representational options have advantages and disadvantages, so that focusing on one framework can force an architecture into awkward approaches to certain problems. On the other hand, even mixed architectures are typically limited to a few types of knowledge structures to avoid complexity.

One common tradition distinguishes declarative from procedural representations. Declarative encodings of knowledge can be manipulated by cognitive

mechanisms independent of their content. For instance, a notation for describing devices might support design, diagnosis, and control. First-order logic (Genesereth & Nilsson, 1987) is a classic example of such a representation. Generally speaking, declarative representations support very flexible use, but they may lead to inefficient processing. In contrast, procedural formalisms encode knowledge about how to accomplish some task. For instance, an agent might have a procedure that lets it solve an algebra problem or drive a vehicle, but not recognize such an activity when done by others. Production rules (Neches et al., 1987) are a common means of representing procedural knowledge. In general, procedural representations let an agent apply knowledge efficiently, but typically in an inflexible manner.

We should clarify that a cognitive architecture can support both declarative and procedural representations, so they are not mutually exclusive. Also, all architectures have some declarative and procedural aspects, in that they require some data structures to recognize and some interpreter to control behavior. However, we typically reserve the term knowledge to refer to structures that are fairly stable (not changing on every cycle) and that are not built into the architecture. Moreover, whether knowledge is viewed as declarative or procedural depends less on its format than on what architectural mechanisms can access it. For example, production rules can be viewed as declarative if other production rules can inspect them.

Although much of an agent's knowledge must consist of skills, concepts, and facts about the world it inhabits, an architecture may also support meta-knowledge about the agent's own capabilities. Such higher-level knowledge can support meta-reasoning, let the agent \know what it knows", and provide a natural way to achieve cognitive penetrability, that is, an understanding of the cognitive steps taken during the agent's activities and the reasons for them. Encoding knowledge in a declarative manner is one way to achieve meta-knowledge, but an emphasis on procedural representations does not mean an architecture cannot achieve these ends through other means.

Another contrast parallels the common distinction between activities and the entities on which they operate. Most cognitive architectures, because they evolved from theories of problem solving and planning, focus on skill knowledge about how to generate or execute sequences of actions, whether in the agent's head or in the environment. However, an equally important facet of cognition is conceptual knowledge, which deals with categories of objects, situations, and other less action-oriented concepts. All cognitive architectures refer to such categories, but they often relegate them to opaque symbols, rather than representing their meaning explicitly. There has been considerable work on formalisms and methods for conceptual memory, but seldom in the context of cognitive architectures.

Yet another distinction (Tulving, 1972) involves whether stored knowledge supports a semantic memory of generic concepts, procedures, and the like, or whether it encodes an episodic memory of speciþc entities and events the agent has encountered in the environment. Most cognitive architectures focus on semantic memory, partly because this is a natural approach to obtaining the generalized behavior needed by an intelligent agent, whereas an episodic memory seems well suited for retrieval of speciþc facts and occurrences. However, methods for analogical and case-based reasoning can produce the eÞect of generalized behavior at retrieval time, so an architecture's commitment to semantic or episodic memory does not, by itself, limit its capabilities. Neither must memory be restricted to one framework or the other.

Researchers in artiþcial intelligence and cognitive science have explored these design decisions through a variety of speciþc representational formalisms. An early notation, known as semantic networks (Ali & Sowa, 1993; Sowa, 1991), encodes both generic and speciþc knowledge in a declarative format that consists of nodes (for concepts or entities) and links (for relations between them). First-order logic was another early representational framework that still sees considerable use; this encodes knowledge as logical expressions, each cast in terms of predicates and arguments, along with statements that relate these expressions in terms of logical operators like conjunction, disjunction, implication, and negation. Production systems (Neches, Langley, & Klahr, 1987) provide a more procedural notation, retaining the modularity of logic, which represent knowledge as a set of condition-action rules that describe plausible responses to diÞerent situations. Frames (Minsky, 1975) and schemas oÞer structured declarative formats that specify concepts in terms of attributes (slots) and their values (þllers), whereas plans (Hendler et al., 1990) provide a structured framework for encoding courses of action. In addition, some approaches augment symbolic structures with strengths (as in neural networks) or probabilities (as in Bayesian networks), although, as typically implemented, these have limited expressiveness.

4.2 Organization of Knowledge

Another important set of properties concerns the manner in which a cognitive architecture organizes knowledge in its memory. One choice that arises here is whether the underlying knowledge representation scheme directly supports ' at' or hierarchical structures. Production systems and propositional logic are two examples of at frameworks, in that the stored memory elements make no direct reference to each other. This does not mean they cannot in uence one another; clearly, application of one production rule can lead to another one's selection on the next cycle, but this happens indirectly through operation of the architecture's interpreter.

In contrast, stored elements in structured frameworks make direct reference to other elements. One such approach involves a task hierarchy, in which one plan or skill calls directly on component tasks, much as in subroutine calls. Similarly, a part-of hierarchy describes a complex object or situation in terms of its components and relations among them. A somewhat diﬀerent organization occurs with an is-a hierarchy, in which a category refers to more general concepts (its parents) and more specialized ones (its children). Most architectures commit to either a flat or structured scheme, but task, part-of, and is-a hierarchies are complementary rather than mutually exclusive.

A second organizational property involves the granularity of the knowledge stored in memory. For example, both production systems and first-order logic constitute fairly fine-grained forms of knowledge. An architecture that encodes knowledge in this manner must use its interpreter to compose them in order to achieve complex behavior. Another option is to store more coarse-grained structures, such as plans and macro-operators, that effectively describe multi-step behavior in single knowledge structures. This approach places fewer burdens on the interpreter, but also provides less flexibility and generality in the application of knowledge. A structured framework offers one compromise by describing coarse memory elements in terms of fine-grained ones, thus giving the agent access to both.

Another organizational issue concerns the number of distinct memories that an architecture supports and their relations to each other. An intelligent agent requires some form of long-term memory to store its generic skills and concepts; this should be relatively stable over time, though it can change with instruction and learning. An agent also requires some short-term memory that contains more dynamic and short-lived beliefs and goals. In most production system architectures, these two memories are structurally distinct but related through the matching process, which compares the conditions of long-term production rules with short-term structures. Other frameworks treat short-term memory as the active portion of the long-term store, whereas others replace a single short-term memory with a number of modality-specific perceptual buffers. A cognitive architecture may also allocate its stable knowledge to distinct long-term memories, say for procedural, conceptual, and episodic structures, as appears to occur in humans.

4.3 Utilization of Knowledge

A third class of properties concerns the utilization of knowledge stored in long-term memories. As we have seen, this can range from low-level activities like recognition and decision making to high-level ones like communication and reflection. We cannot hope to cover all the design choices that arise in

knowledge utilization, so we focus here on issues which deal with cognitive behavior that occurs across cycles, which is typically a central concern of architectural developers.

One such design decision involves whether problem solving relies primarily on heuristic search through problem spaces or on retrieval of solutions or plans from long-term memory. As usual, this issue should not be viewed as a strict dichotomy, in that problem space search itself requires retrieval of relevant operators, but a cognitive architecture may emphasize one approach over the other. For instance, production system architectures typically construct solutions through heuristic search, whereas case-based systems retrieve solutions from memory, though the latter must often adapt the retrieved structure, which itself can require search.

When a cognitive architecture supports multi-step problem solving and inference, it can accomplish this in diþerent ways. One approach, known as **forward chaining**, applies relevant operators and inference rules to the current problem state and current beliefs to produce new states and beliefs. We can view forward chaining as progressing from a known mental state toward some goal state or description. In contrast, **backward chaining** applies relevant operators and inference rules to current goals in order to generate new subgoals, which involves progression from some goal state or description toward current states or beliefs. A third alternative, **means-ends analysis** (e.g., Carbonell et al., 1990; Ernst & Newell, 1969), combines these two approaches by selecting operators through backward chaining but executing them whenever their preconditions are satisþed.

To clarify this dimension, production system architectures typically operate in a forward chaining fashion, while Prolog (Clocksin & Mellish, 1981) provides a good example of backward chaining. However, it is important to distinguish between problem solving techniques that are supported directly by an architecture and ones that are implemented by knowledge stated within that architecture. For instance, backward-chaining behavior can arise within a forward-chaining production system through rules that match against goals and, upon þring, add subgoals to short-term memory (e.g., Anderson & Lebiere, 1998). Such knowledge-driven behavior does not make the architecture itself any less committed to one position or another.

Computer scientists often make a strong distinction between sequential and parallel processing, but this dichotomy, as typically stated, is misleading in the context of cognitive architectures. Because an intelligent agent exists over time, it cannot avoid some sequential processing, in that it must take some cognitive and physical steps before others are possible. On the other hand, most research on cognitive architectures assumes that retrieval of structures from long-term memory occurs in parallel or at least that it happens so rapidly

it has the same eÞect. However, frameworks can genuinely diÞer in the number of cognitive structures they select and apply on each cycle. For example, early production system architectures (Newell, 1973b) found all matching instantiations of rules on each cycle, but then selected only one for application; in contrast, some more recent architectures like Soar (Newell, 1990) apply all matching rules, but introduce constraints elsewhere, as in the number of goals an agent can simultaneously pursue. Thus, architectures diÞer not so much in whether they support sequential or parallel processing, but in where they place sequential bottlenecks and the details of those constraints. Some architectures, like ACT-R (Anderson et al., 2004) model cognitive bottlenecks in order to simulate limitations on human performance.

Given that a cognitive architecture has some resource limitations which require selection among alternative goals, rules, or other knowledge structures, it needs some way to make this selection. Early production system architectures handled this through a process known as con ict resolution, which selected one or more matched rules to apply based on criteria like the recency of their matched elements, the rules' speciþcities, or their strength. Computer programs for game playing instead select moves with some numeric evaluation function that combines features of predicted states, whereas systems that incorporate analogical or case-based reasoning typically select structures that are most similar to some target. Again, it is important to distinguish the general mechanism an architecture uses to select among alternative decisions or actions from the knowledge it uses to implement that strategy, which may diÞer across tasks or change with learning.

Another central issue for the utilization of knowledge concerns the relation between cognition and action. A deliberative architecture is one that plans or reasons out a course of action before it begins execution, whereas a reactive architecture simply selects its actions on each decision cycle based on its understanding of the current situation. Deliberation has advantages in predictable environments, but it requires an accurate model of actions' eÞects and forces the agent to construct a plan for each new problem it encounters. Reaction has advantages in dynamic and unpredictable environments, but requires the presence of control knowledge for many diÞerent situations. Some architectures (e.g., Carbonell et al., 1990) lean toward deliberation because they grew out of research on problem solving and planning, whereas other frameworks (e.g., Brooks, 1986) emphasize reactive execution to the exclusion of deliberation. Both positions constitute extremes along a continuum that, in principle, should be controlled by agent knowledge rather than built into the architecture. [2]

[2] Another response is to support deliberation and reactive control in separate modules, as done in Bonasso et al.'s (1997) 3T framework.

A similar issue arises with respect to the relation between perception and action (Schmidt, 1975). A closed-loop control system senses the environment on every cycle, thus giving an agent the opportunity to respond to recent changes. In contrast, an open-loop system carries out an extended action sequence over multiple cycles, without bothering to sense the environment. Closed-loop approaches are often associated with reactive systems and open-loop methods with deliberative ones, but they really involve distinct issues. Closed-loop control has the advantage of rapid response in dynamic domains, but requires constant monitoring that may exceed an agent's perceptual resources. Open-loop behavior requires no sensing and supports eŽcient execution, but it seems most appropriate only for complex skills that necessitate little interaction with the environment. Again, these two extremes deþne a continuum, and an architecture can utilize domain knowledge to determine where its behavior falls, rather than committing to one or the other.

4.4 Acquisition and Reþnement of Knowledge

A þnal important class of properties concerns the acquisition of knowledge from instruction or experience. Although such learning mechanisms can be called intentionally by the agent and carried out in a deliberative fashion, both their invocation and execution are typically handled at the architectural level, though the details vary greatly. One important issue is whether a cognitive architecture supports many such mechanisms or whether it relies on a single learning process that (ideally) interacts with knowledge and experience to achieve many diÞerent eÞects. For instance, early versions of ACT included þve distinct learning processes, whereas early versions of Soar included only one such mechanism.

The literature on cognitive architectures commonly distinguishes between processes that learn entirely new knowledge structures, such as production rules or plans, and ones that þne tune existing structures, say through adjusting weights or numeric functions. For example, Soar learns new selection, rejection, or preference rules when it creates results in a subgoal, whereas ACT-R updates the utilities associated with production rules based on their outcomes. An architectural learning mechanism may also revise existing structures by adding or removing components. For instance, early versions of ACT included a discrimination method that added conditions to production rules and a generalization method that removed them.

Another common distinction involves whether a given learning process is analytical or empirical in nature (Schlimmer & Langley, 1992). Analytical methods rely on some form of form of reasoning about the learning experience in terms of knowledge available to the agent. In contrast, empirical methods rely

on inductive operations that transform experience into usable knowledge based on detected regularities. In general, analytical methods are more explanatory in favor and empirical methods are more descriptive. This is actually a continuum rather than a dichotomy, in which the critical variable is the amount of knowledge-based processing the learner carries out. Architectures can certainly utilize hybrid methods that incorporate ideas from both frameworks, and they can also combine them through different learning mechanisms. For example, Prodigy utilizes an analytic method to construct new rules and an empirical method to estimate their utility after gaining experience with them.

A fourth issue concerns whether an architecture's learning mechanisms operate in an eager or a lazy fashion. Most frameworks take an eager approach that forms generalized knowledge structures from experience at the time the latter enter memory. The interpreter can then process the resulting generalized rules, plans, or other structures without further transformation. Methods for rule induction and macro-operator construction are good examples of this approach. However, some architectures take a lazy approach (Aha, 1997) that stores experiences in memory untransformed, then carry out implicit generalization at the time of retrieval and utilization. Analogical and case-based methods (e.g., Veloso, & Carbonell, 1993) are important examples of this approach.

A final property revolves around whether learning occurs in an incremental or nonincremental manner. Incremental methods incorporate training cases one at a time, with limited memory for previous cases, and update their knowledge bases after processing each experience. In contrast, nonincremental methods process all training cases in a single step that operates in a batch procedure. Because agents exist over time, they accumulate experience in an online fashion, and their learning mechanisms must deal with this constraint. Incremental methods provide a natural response, but the order of presentation can influence their behavior (Langley, 1995b). Nonincremental approaches avoid this drawback, but only at the expense of retaining and reprocessing all experiences. Most architectural research takes an incremental approach to learning, though room remains for hybrid methods that operate over limited subsets of experience.

5 Evaluation Criteria for Cognitive Architectures

As with any scientific theory or engineered artifact, cognitive architectures require evaluation. However, because architectural research occurs at the systems level, it poses more challenges than does the evaluation of component knowledge structures and methods. In this section, we consider some dimensions along which one can evaluate cognitive architectures. In general, these involve matters of degree, which suggests the use of quantitative measures

rather than all-or-none tests. Langley and Messina (2004) discuss additional issues that arise in the evaluation of integrated intelligent systems.

Recall that ability to explain psychological phenomena is an important dimension along which to evaluate architectures. For example, in recent years, research within a number of architectural frameworks (Anderson et al., 2004; Sun et al., 2001) has emphasized þtting timing and error data from detailed psychological experiments, but that is not our focus here. However, it is equally important to demonstrate that an architecture supports the same qualitative robustness that humans exhibit. The criteria we discuss in this section are based directly on such qualitative aspects of human behavior, even when a system may produce them through entirely diÞerent means.

Cognitive architectures also provide a distinctive approach to constructing integrated intelligent systems. The convential wisdom of software engineering is that one should develop independent modules that have minimal interaction. In contrast, a cognitive architecture oÞers a uniþed theory of cognition (Newell, 1990) with tightly interleaved modules that support synergistic effects. However, claims about synergy in cognitive systems are diŽcult to test empirically,[3] so here we focus on other criteria that are linked directly to functionality.

5.1 Generality, Versatility, and Taskability

Recall that cognitive architectures are intended to support general intelligent behavior. Thus, generality is a key dimension along which to evaluate a candidate framework. We can measure an architecture's generality by using it to construct intelligent systems that are designed for a diverse set of tasks and environments, then testing its behavior in those domains. The more environments in which the architecture supports intelligent behavior, and the broader the range of those environments, the greater its generality.

However, demonstrating the generality of an architecture may require more or less eÞort on the part of the system developer. For each domain, we might implement a new system in low-level assembly code, which makes few theoretical commitments or high-level mechanisms, but this approach would take much too long. We can deþne the versatility of a cognitive architecture in terms of the diŽculty encountered in constructing intelligent systems across a given set of tasks and environments. The less eÞort it takes to get an architecture to produce intelligent behavior in those environments, the greater its versatility.

[3] Langley and Choi (2006) provide qualitative arguments that their Icarus framework beneþts from interactions among its modules, but even evidence of this sort is rare.

Generality and versatility are related to a third notion, the taskability of an architecture, which acknowledges that long-term knowledge is not the only determinant of an agent's behavior in a domain. Brie y, this concerns an architecture's ability to carry out diPerent tasks in response to goals or other external commands from a human or from some other agent. The more tasks an architecture can perform in response to such commands, and the greater their diversity, the greater its taskability. This in turn can in uence generality and versatility, since it can let the framework cover a wider range of tasks with less ePort on the developer's part.

5.2 Rationality and Optimality

We usually consider an agent to be intelligent when it pursues a behavior for some reason, which makes the rationality of an architecture another important dimension for its evaluation. We can measure a framework's rationality by examining the relationship among its goals, its knowledge, and its actions. For instance, Newell (1982) states \If an agent has knowledge that one of its actions will lead to one of its goals, then the agent will select that action". Since an architecture makes many decisions about action over time, we can estimate this sense of rationality by noting the percentage of times that its behavior satisPes the criterion.

Note that this notion of rationality takes no position about how to select among multiple actions that are relevant to the agent's goals. One response to this issue comes from Anderson (1991), who states \The cognitive system optimizes the adaptation of the behavior of the organism". The notion of optimality assumes some numeric function over the space of behaviors, with the optimal behavior being the one that produces the best value on this function. Although optimality is an all-or-none criterion, we can measure the degree to which an architecture approaches optimality by noting the percentage of times its behavior is optimal across many decision cycles or the ratio of actual to optimal value it achives averaged over time.

However, Simon (1957) has argued that, because intelligent agents have limited cognitive resources, the notion of bounded rationality is more appropriate than optimality for characterizing their behavior. In his view, an agent has bounded rationality if it behaves in a manner that is as nearly optimal with respect to its goals as its resources will allow. We can measure the degree to which a cognitive architecture exhibits bounded rationality in the same manner as for optimality, provided we can incorporate some measure of the resources it has available for each decision.

5.3 Efficiency and Scalability

Because cognitive architectures must be used in practice, they must be able to perform tasks within certain time and space constraints. Thus, efficiency is another important metric to utilize when evaluating an architecture. We can measure efficiency in quantitative terms, as the time and space required by the system, or in all-or-none terms, based on whether the system satisfies hard constraints on time and space, as in work on real-time systems. We can also measure efficiency either at the level of the architecture's recognize-act cycle or at the level of complete tasks, which may give very different results.

However, because architectures must handle tasks and situations of different difficulty, we also want to know their scalability. This metric is closely related to the notion of complexity as used in the formal analysis of algorithms. Thus, we can measure an architecture's space and time efficiency in terms of how they are influenced by task difficulty, environmental uncertainty, length of operation, and other complicating factors. We can examine an architecture's complexity profile across a range of problems and amounts of knowledge. The less an architecture's efficiency is affected by these factors, the greater its scalability.

A special case of scalability that has received considerable attention arises with cognitive architectures that learn over time. As learning mechanisms add knowledge to their long-term memory, many such systems become slower in their problem-solving behavior, since they have more alternatives from which to choose. This utility problem (Minton, 1990) has arisen in different architectures that employ a variety of representational formalisms and retrieval mechanisms. Making architectures more scalable with respect to such increased knowledge remains an open research issue.

5.4 Reactivity and Persistence

Many cognitive architectures aim to support agents that operate in external environments that can change in unpredictable ways. Thus, the ability to react to such changes is another dimension on which to evaluate candidate frameworks. We can measure an architecture's reactivity in terms of the speed with which it responds to unexpected situations or events, or in terms of the probability that it will respond on a given recognize-act cycle. The more rapidly an architecture responds, or the greater its chances of responding, the greater its reactivity. [4]

[4] The notion of interruptability is closely related to reactivity, but is associated primarily with architectures that deliberate or pursue explicit plans, which can be

Of course, this deþnition must take into account the relation between the environment and the agent's model of that environment. If the model predicts accurately what transpires, then reactivity becomes less of an issue. But if the environment is an uncertain one or if the agent has a weak model, then reactivity becomes crucial to achievement of the agent's goals. Alternative cognitive architectures can take diþerent positions along this spectrum, and we must understand that position when evaluating their reactivities.

An issue related to reactivity that has received substantial attention is known as the frame problem (McCarthy, 1963). This arises in any dynamic environment where an agent must keep its model of the world aligned with the world itself, despite the inability of the agent to sense the world in its entirety. Even when it is not hard to detect environmental changes themselves, propagating the eþect of these changes on knowledge, goals, and actions can be diŽcult. Many research eþorts have addressed the frame problem, but making architectures more robust on this front remains an open area for research.

Despite the importance of reactivity, we should note that, in many contexts, persistence is equally crucial. An architecture that always responds immediately to small environmental changes may lose sight of its longer-term objectives and oscillate from one activity to another, with no higher purpose. We can measure persistence as the degree to which an architecture continues to pursue its goals despite changes in the environment. Reactivity and persistence are not opposites, although they may appear so at þrst glance. An agent can react to short-term changes while still continuing to pursue its long-term objectives.

5.5 Improvability

We expect intelligent agents to improve their behavior over time. One means to this end involves direct addition of knowledge by the system's programmer or user. The key question here is not whether such additions are possible, but how eþective they are at improving the agent's behavior. Thus, we can measure improvability of this type in terms of the agent's ability to perform tasks that it could not handle before the addition of knowledge. More speciþcally, we can measure the rate at which performance improves as a function of programmer time, since some architectures may require less eþort to improve than others.

Another path to improvement involves the agent learning from its experience with the environment or with its own internal processes. We can measure an architecture's capacity for learning in the same way that we can measure its capacity for adding knowledge { in terms of its ability to perform new tasks.

---

interrupted when unexpected events occur.

Since cognitive agents exist over time, this means measuring their improvement in performance as a function of experience. Thus, the method commonly used in machine learning of separating training from test cases makes little sense here, and we must instead collect learning curves that plot performance against experience in an online setting.

We should note that diþerent forms of learning focus on diþerent types of knowledge, so we should not expect a given mechanism to improve behavior on all fronts. For example, some learning processes are designed to improve an agent's ability to recognize objects or situations accurately, others focus on acquisition of new skills, and still others aim to make those skills more eŽcient. We should use diþerent tests to evaluate an architecture's ability to learn diþerent types of knowledge, although we would expect a well-rounded architecture to exhibit them all.

Because learning is based on experience with speciþc objects or events, evaluating the generality, transfer, and reusability of learned knowledge is also crucial. We want learning to involve more than memorizing speciþc experiences, though such episodic memory also has its uses. We can determine the degree of generalization and transfer by exposing the agent to situations and tasks that diþer from its previous experience in various ways and measuring its performance on them. Again, a key issue concerns the rate of learning or the amount of acquired knowledge that the architecture needs to support the desired behavior.

5.6 Autonomy and Extended Operation

Although we want intelligent agents that can follow instructions, sometimes we also expect them to operate on their own over extended periods. To this end, the architectures that support them must be able to create their own tasks and goals. Moreover, they must be robust enough to keep from failing when they encounter unexpected situations and to keep from slowing down as they accumulate experience over long periods of time. In other words, a robust architecture should provide both autonomy and extended operation.

We can measure an architecture's support for autonomy by presenting agents with high-level tasks that require autonomous decision making for success and that beneþt from knowledge about the domain. For example, we can provide an agent with the ability to ask for instructions when it does not know how to proceed, then measure the frequency with which it requests assistance as a function of its knowledge. We can measure the related ability for extended operation by placing an agent in open-ended environments, such as a simulated planetary expedition, and noting how long, on average, it continues before failing or falling into inaction. We can also measure an agent's eŽciency as a

function of its time in the þeld, to determine whether it scales well along this dimension.

6 Open Issues in Cognitive Architectures

Despite the many conceptual advances that have occurred during three decades of research on cognitive architectures, and despite the practical use that some architectures have seen on real-world problems, there remains considerable need for additional work on this important topic. In this section, we note some open issues that deserve attention from researchers in the area.

The most obvious arena for improvement concerns the introduction of new capabilities. Existing architectures exhibit many of the capacities described in Section 3, but few support all of them, and even those achieve certain functionalities only with substantial programmer eÞort. Some progress has been made on architectures that combine deliberative problem solving with reactive control, but we need increased eÞorts at uniþcation along a number of other fronts:

- Most architectures emphasize the generation of solutions to problems or the execution of actions, but categorization and understanding are also crucial aspects of cognition, and we need increased attention to these abilities.

- The focus on problem solving and procedural skills has drawn attention away from episodic knowledge. We need more research on architectures that directly support both episodic memory and re ective processes that operate on the structures it contains.

- Most architectures emphasize logic or closely related formalisms for representing knowledge, whereas humans also appear to utilize visual, auditory, diagrammatic, and other specialized representational schemes. We need extended frameworks that can encode knowledge in a variety of formalisms, relate them to each other, and use them to support intelligent behavior more exibly and eÞectively.

- Although natural language processing has been demonstrated within some architectures, few intelligent systems have combined this with the ability to communicate about their own decisions, plans, and other cognitive activities in a general manner.

- Physical agents have limited resources for perceiving the world and aÞecting it, yet few architectures address this issue. We need expanded frame-

works that manage an agent's resources to selectively focus its perceptual attention, its eÞectors, and the tasks it pursues.

- Although many architectures interface with complex environments, they rarely confront the interactions between body and mind that arise with real embodiment. For instance, we should examine the manner in which physical embodiment impacts thinking and consider the origin of agents' primary goals in terms of internal drives.

- Emotions play a central role in human behavior, yet few systems oÞer any account of their purposes or mechanisms. We need new architectures that exhibit emotion in ways that link directly to other cognitive processes and that modulate intelligent behavior.

- From an engineering standpoint, architectures are interesting if they ease development of intelligent agents through reuse, but we need research on whether this is best accomplished through specialized functional capabilities that are utilized repeatedly or through reusable knowledge that supports multiple tasks.

- From an engineering standpoint, architectures are primarily interesting if they can ease development of intelligent agents. To that end, reusability is key, but it is not clear if the architectures themselves need to support specialized capabilities that can be reused, or if it is possible to develop reusable knowledge that supports multiple tasks.

Architectures that demonstrate these new capabilities will support a broader class of intelligent systems than the þeld has yet been able to develop.

We also need additional research on the structures and processes that support such capabilities. Existing cognitive architectures incorporate many of the underlying properties that we described in Section 4, but a number of issues remain unaddressed.

- Certain representational frameworks { production systems and plans { have dominated architectural research. To explore the space of architectures more fully, we should also examine designs that draw on other representational frameworks like frames (Minsky, 1975), case bases (Aamodt & Plaza, 1994), description logics (Nardi & Brachman, 2002), and probabilistic formalisms (Richardson & Domingos, 2006).

- Many architectures commit to a single position on properties related to knowledge utilization, but this is not the only alternative. We should also explore frameworks that change their location on a given spectrum (e.g., deliberative vs. reactive behavior) dynamically based on their situation.

- Most architectures incorporate some form of learning, but none have shown the richness of improvement that humans demonstrate. We need more robust and flexible learning mechanisms that are designed for extended operation in complex, unfamiliar domains and that build in a cumulative manner on the results of previous learning over long periods of time.

These additional structures and processes should both increase our understanding of the space of cognitive architectures and provide capabilities that are not currently available.

The research community should also devote more serious attention to methods for the thoughtful evaluation of cognitive architectures. Metrics like those we proposed in Section 5 are necessary but not sufficient to understand scientifically the mapping from architectural properties to the capabilities they support. In addition, we must identify or create complex environments, both physical and simulated, that exercise these capabilities and provide realistic opportunities for measurement.

We will also need an experimental method that recognizes the fact that cognitive architectures involve integration of many components which may have synergistic effects, rather than consisting of independent but unrelated modules (Langley & Messina, 2004). Experimental comparisons among architectures have an important role to play, but these must control carefully for the task being handled and the amount of knowledge encoded, and they must measure dependent variables in unbiased and informative ways. Systematic experiments that are designed to identify sources of power will tell us far more about the nature of cognitive architectures than simplistic competitions.

Our field still has far to travel before we understand fully the space of cognitive architectures and the principles that underlie their successful design and utilization. However, we now have over two decades' experience with constructing and using a variety such architectures for a wide range of problems, along with a number of challenges that have arisen in this pursuit. If the scenery revealed by these initial steps are any indication, the journey ahead promises even more interesting and intriguing sites and attractions.

Acknowledgments

Production of this paper was supported, in part, by Grant HR0011-04-1-0008 from DARPA IPTO and by Grant IIS-0335353 from NSF. The document borrows greatly, in both organization and content, from the University of Michigan Web site at http://ai.eecs.umich.edu/cogarch0/, which was authored by R. Wray, R. Chong, J. Phillips, S. Rogers, B. Walsh, and J. E. Laird. We thank Ron Brachman for convincing us that the paper was needed and encouraging us to write it.

References

Aamodt, A., & Plaza, E. (1994). Case-based reasoning: Foundational issues, methodological variations, and system approaches. AI Communications, 7, 39{59.

Aha, D. W. (1997). Lazy learning. Dordrecht, Germany: Kluwer.

Albus, J. S., Pape, C. L., Robinson, I. N., Chiueh, T.-C., McAulay, A. D., Pao, Y.-H., & Takefuji, Y. (1992). RCS: A reference model for intelligent control. IEEE Computer, 25, 56{79.

Ali, S. S. & Shapiro, S. C. (1993). Natural language processing using a propositional semantic network with structured variables. Minds and Machines, 3, 421{451.

Anderson, J. R. (1991). Cognitive architectures in a rational analysis. In K. VanLehn (Ed.), Architectures for intelligence. Hillsdale, NJ: Lawrence Erlbaum.

Anderson, J. R. (2007). How can the human mind exist in the physical universe?. New York: Oxford University Press.

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

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. Psychological Review, 111, (4). 1036{1060.

Bonasso, R. P., Firby, R. J., Gat, E., Kortenkamp, D., Miller, D., & Slack, M. (1997). Experiences with an architecture for intelligent, reactive agents. Journal of Experimental and Theoretical Artiþcial Intelligence, 9, 237{256.

Bratman, M. E. (1987). Intentions, plans, and practical reason. Cambridge, MA: Harvard University Press.

Brooks, R. A. (1986). A robust layered control system for a mobile robot. IEEE Journal of Robotics and Automation, RA-2, 14{23.

Carbonell, J. G., Knoblock, C. A., & Minton, S. (1990). Prodigy: An integrated architecture for planning and learning. In K. Van Lehn (Ed.), Architectures for intelligence. Hillsdale, NJ: Lawrence Erlbaum.

Choi, D., Konik, T., Nejati, N., Park, C., & Langley, P. (2007). A believable agent for þrst-person shooter games. Proceedings of the Third Annual Artiþcial Intelligence and Interactive Digital Entertainment Conference (pp.

71{73). Stanford, CA: AAAI Press.

Clocksin, W. F., & Mellish, C. S. (1981). Programming in Prolog. Berlin: Springer-Verlag.

Drummond, M., Bresina, J., & Kedar, S. (1991). The Entropy Reduction Engine: Integrating planning, scheduling, and control. SIGART Bulletin, 2, 61{65.

Ernst, G., & Newell, A. (1969). GPS: A case study in generality and problem solving. New York: Academic Press.

Fikes, R., Hart, P. E., & Nilsson, N. J. (1972). Learning and executing generalized robot plans. Artiþcial Intelligence, 3, 251{288.

Firby, R. J. (1994). Task networks for controlling continuous processes. Proceedings of the Second International Conference on AI Planning Systems (pp. 49{54). Chicago: AAAI Press.

Freed, M. (1998). Managing multiple tasks in complex, dynamic environments. Proceedings of the Fifteenth National Conference on Artiþcial Intelligence (pp. 921{927). Madison, WI: AAAI Press.

Gat, E. (1991). Integrating planning and reacting in a heterogeneous asynchronous architecture for mobile robots. SIGART Bulletin, 2, 17{74.

Genesereth, M. R., & Nilsson, N. J. (1987). Logical foundations of artiþcial intelligence. Los Altos, CA: Morgan Kaufmann.

Gratch, J. (2000). Emile: Marshalling passions in training and education. Proceedings of the Fourth International Conference on Autonomous Agents (pp. 325{332). Barcelona, Spain.

Haigh, K., & Veloso, M. (1996). Interleaving planning and robot execution for asynchronous user requests. Proceedings of the International Conference on Intelligent Robots and Systems (pp. 148{155). Osaka, Japan: IEEE Press.

Hayes-Roth, B., P eger, K., Lalanda, P., Morignot, P., & Balabanovic, M. (1995). A domain-speciþc software architecture for adaptive intelligent systems. IEEE Transactions on Software Engineering, 21, 288{301.

Hendler, J., Tate, A., & Drummond, M. (1990). AI planning: Systems and techniques. AI Magazine, 11, 61{77.

Ingrand, F. F., GeorgeÞ, M. P., & Rao, A. S. (1992). An architecture for real-time reasoning and system control. IEEE Expert, 7, 34{44.

Koedinger, K. R., Anderson, J. R., Hadley, W. H., & Mark, M. (1997). In-

telligent tutoring goes to school in the big city. International Journal of Artiþcial Intelligence in Education, 8, 30{43.

Konolige, K., Myers, K. L., Ruspini, E. H., & SaŽotti, A. (1997). The Saþra architecture: A design for autonomy. Journal of Experimental & Theoretical Artiþcial Intelligence, 9, 215{235.

Laird, J. E. (1991). Preface for special section on integrated cognitive architectures. SIGART Bulletin, 2, 12{123.

Laird, J. E. (2008). Extending the Soar cognitive architecture. Proceedings of the Artiþcial General Intelligence Conference. Memphis, TN: IOS Press.

Laird, J. E., Rosenbloom, P. S., & Newell, A. (1986). Chunking in Soar: The anatomy of a general learning mechanism. Machine Learning, 1, 11{46.

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

Langley, P. (1995a). Elements of machine learning. San Francisco: Morgan Kaufmann.

Langley, P. (1995b). Order eÞects in incremental learning. In P. Reimann & H. Spada (Eds.), Learning in humans and machines: Towards and interdisciplinary learning science. Oxford: Elsevier.

Langley, P. (2006). Intelligent behavior in humans and machines (Technical Report). Computational Learning Laboratory, CSLI, Stanford University, CA.

Langley, P., & Choi, D. (2006a). A uniþed cognitive architecture for physical agents. Proceedings of the Twenty-First AAAI Conference on Artiþcial Intelligence. Boston: AAAI Press.

Langley, P., & Choi, D. (2006b). Learning recursive control programs from problem solving. Journal of Machine Learning Research, 7, 493{518.

Langley, P., Cummings, K., & Shapiro, D. (2004). Hierarchical skills and cognitive architectures. Proceedings of the Twenty-Sixth Annual Conference of the Cognitive Science Society (pp. 779{784). Chicago, IL.

Langley, P., & Messina, E. (2004). Experimental studies of integrated cognitive systems. Proceedings of the Performance Metrics for Intelligent Systems Workshop. Gaithersburg, MD.

Lewis, R. L. (1993). An architecturally-based theory of sentence comprehension. Proceedings of the Fifteenth Annual Conference of the Cognitive Science Society (pp. 108{113). Boulder, CO: Lawrence Erlbaum.

Magerko, B., Laird, J. E., Assanie, M., Kerfoot, A., & Stokes, D. (2004). AI characters and directors for interactive computer games. Proceedings of the Sixteenth Innovative Applications of Artiþcial Intelligence Conference (pp. 877-884). San Jose, CA: AAAI Press.

McCarthy, J. (1963). Situations, actions and causal laws (Memo 2). Artiþcial Intelligence Project, Stanford University, Stanford, CA.

Meyer, M., & Kieras, D. (1997). A computational theory of executive control processes and human multiple-task performance: Part 1. Basic mechanisms. Psychological Review, 104, 3{65.

Miller, C. S., & Laird, J. E. (1996). Accounting for graded performance within a discrete search framework. Cognitive Science, 20, 499{537.

Minsky, M. (1975). A framework for representing knowledge. In P. Winston (Ed.), The psychology of computer vision. New York: McGraw-Hill.

Minton, S. N. (1990). Quantitative results concerning the utility of explanation-based learning. Artiþcial Intelligence, 42, 363{391.

Muscettola, N., Nayak, P. P., Pell, B., & Williams, B. (1998). Remote Agent: To boldly go where no AI system has gone before. Artiþcial Intelligence, 103, 5{48.

Musliner, D. J., Goldman, R. P., & Pelican, M. J. (2001). Planning with increasingly complex models. Proceedings of the International Conference on Intelligent Robots and Systems.

Nardi, D., & Brachman, R. J. (2002). An introduction to description logics. In F. Baader et al. (Eds.), Description logic handbook. Cambridge: Cambridge University Press.

Nason, S., & Laird, J. E. (2004). Soar-RL: Integrating reinforcement learning with Soar. Proceedings of the Sixth International Conference on Cognitive Modeling (pp. 220{225).

Neches, R., Langley, P., & Klahr, D. (1987). Learning, development, and production systems. In D. Klahr, P. Langley, & R. Neches (Eds.), Production system models of learning and development. Cambridge, MA: MIT Press.

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

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

Newell, A. (1982). The knowledge level. Artiþcial Intelligence, 18, 87{127.

Newell, A. (1990). Uniþed theories of cognition. Cambridge, MA: Harvard University Press.

Nuxoll, A. M. & Laird, J. E. (2007). Extending cognitive architecture with episodic memory. Proceedings of the Twenty-Second AAAI Conference on Artiþcial Intelligence. Vancouver, BC: AAAI Press.

Pell, B., Bernard, D. E., Chien, S. A., Gat, E., Muscettola, N., Nayak, P. P., Wagner, M. D., & Williams, B. C. (1997). An autonomous spacecraft agent prototype. Proceedings of the First International Conference on Autonomous Agents (pp. 253{261). Marina del Rey, CA: ACM Press.

Perez, M. A. & Carbonell, J. G. (1994). Control knowledge to improve plan quality. Proceedings of the Second International Conference on AI Planning Systems (pp. 323{328). Chicago: AAAI Press.

Remington, R., Matessa, M., Freed, M., & Lee, S. (2003). Using Apex / CPM-GOMS to develop human-like software agents. Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems. Melbourne: ACM Press.

Richardson, M., & Domingos, P. (2006). Markov logic networks. Machine Learning, 62, 107{136.

Sammut, C. (1996). Automatic construction of reactive control systems using symbolic machine learning. Knowledge Engineering Review, 11, 27{42.

Schlimmer, J. C., & Langley, P. (1992). Machine learning. In S. Shapiro (Ed.), Encyclopedia of artiþcial intelligence (2nd ed.). New York: John Wiley & Sons.

Schmidt, R. A. (1975). A schema theory of discrete motor skill learning. Psychological Review, 82, 225{260.

Shapiro, D., & Langley, P. (2004). Symposium on learning and motivation in cognitive architectures: Final report. Institute for the Study of Learning and Expertise, Palo Alto, CA.
http://www.isle.org/symposia/cogarch/arch.þnal.pdf

Simon, H. A. (1957). Models of man. New York: John Wiley.

Sleeman, D., Langley, P., & Mitchell, T. (1982). Learning from solution paths: An approach to the credit assignment problem. AI Magazine, 3, 48{52.

Sloman, A. (2001). Varieties of aÞect and the CogAÞ architecture schema. Proceedings of the AISB'01 Symposium on Emotion, Cognition, and AÞective

Computing. York, UK.

Sowa, J. F. (Ed.). (1991). Principles of semantic networks: Explorations in the representation of knowledge. San Mateo, CA: Morgan Kaufmann.

Sun, R. (Ed.). (2005). Cognition and multi-agent interaction: Extending cognitive mdoeling to social simulation. Cambridge University Press.

Sun, R. (2007). The importance of cognitive architectures: An analysis based on CLARION. Journal of Experimental and Theoretical Artiþcial Intelligence, 19, 159{193.

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

Sutton, R. S. & Barto, A. G. (1998). Reinforcement learning: An introduction. Cambridge, MA: MIT Press.

Taatgen, N. A. (2005). Modeling parallelization and speed improvement in skill acquisition: From dual tasks to complex dynamic skills. Cognitive Science, 29, 421{455.

Tambe, M., Johnson, W. L., Jones, R. M., Koss, F., Laird, J. E., Rosenbloom, P. S., & Schwamb, K. B. (1995). Intelligent agents for interactive simulation environments. AI Magazine, 16, 15{39.

Trafton, J. G., Cassimatis, N. L., Bugajska, M., Brock, D., Mintz, F., & Schultz, A. (2005). Enabling eÞective human-robot interaction using perspective-taking in robots. IEEE Transactions on Systems, Man and Cybernetics, 25, 460{470.

Tulving, E. (1972). Episodic and semantic memory. In E. Tulving & W. Donaldson (Eds.), Organization of memory. New York: Academic Press.

VanLehn, K. (Ed.) (1991). Architectures for intelligence. Hillsdale, NJ: Lawrence Erlbaum.

Veloso, M. M., & Carbonell, J. G. (1993). Derivational analogy in Prodigy: Automating case acquisition, storage, and utilization. Machine Learning, 10, 249{278.

Wang, X. (1995). Learning by observation and practice: An incremental approach for planning operator acquisition. Proceedings of the Twelfth International Conference on Machine Learning (pp. 549{557). Lake Tahoe, CA: Morgan Kaufmann.

Appendix. Representative Cognitive Architectures

Many researchers have proposed and studied cognitive architectures over the past three decades. Some have been only thought experiments, while others have been implemented and utilized as tools by people at many institutions. Here we review brie y a number of architectures that have appeared in the literature. We have not attempted to be exhaustive, but this set should give readers an idea of the great diversity of research in this area.

- ACT-R (Anderson, 2007; Anderson et al., 2004), the most recent instantiation of the ACT family, includes a declarative memory for facts and a procedural memory consisting of production rules. The architecture operates by matching productions on perceptions and facts, mediated by the real-valued activation levels of objects, and executing them to aÞect the environment or alter declarative memory. Learning in ACT-R involves creating new facts and productions, as well as updating base activations and utilities associated with these structures.

- The AIS architecture (Hayes-Roth et al., 1995) stores procedural knowledge as a set of behaviors, each with associated triggering conditions, and control plans, which specify temporal patterns of plan steps. These match against, modify, and interact through a declarative memory that stores factual knowledge, intended activities, and traces of the agent's experience. On each cycle, a meta-controller selects among enabled behaviors and selects which ones to execute. The architecture includes a deliberative cognitive layer, which is responsible for situation assessment and planning, and a more rapid physical layer, which handles perception and action in the environment.

- APEX (Freed, 1998) organizes knowledge in hierarchical procedures, with higher-level elements indexed by the task they address and referring to subtasks they invoke. These match against the contents of a perceptual memory, with an agenda selecting tasks that it adds to an agenda. The architecture associates cognitive, perceptual, and motor resources; this can lead to con icts among tasks on the agenda, which the system resolves by selecting those with highest priority. This can lead to interruption of tasks and later return to them when resources become available.

- CIRCA (Musliner et al., 2001) incorporates a stable memory for possible action, temporal, and event transitions, along with a dynamic memory for speciþc plans and events. The cognitive subsystem generate a planned course of action, encoded as a nondeterministic þnite state graph, starting þrst with an abstract plan and reþning it as appropriate. The architecture passes this structure to a real-time subsystem that operates in parallel

with the cognitive subsystem, letting the former execute the plan while the latter attempts to improve it.

- CLARION (Sun et al., 2001) stores both action-centered and non-action knowledge in implicit form, using multi-layer neural networks, and in explicit form, using symbolic production rules. Corresponding short-term memories contain activations on nodes and symbolic elements that the architecture matches against long-term structures. Performance involves passing sensory information to the implicit layer, which generates alternative high-value actions, and to the explicit layer, which uses rules to propose actions; the architecture then selects the candidate with the highest expected value. Learning involves weight revision in the implicit system, using a combination of reinforcement learning and backpropagation to estimate value functions, and construction of production rules by extraction from the implicit layer, error-driven revision, and instantiation of rule templates.

- CogAÞ (Sloman, 2001) is an architectural schema or framework designed to support interaction between cognition and aÞect. Although it does not commit to speciþc representations, it does posit three distinct levels of processing. A reactive level uses condition-action associations that respond to immediate environmental situations. A deliberative layer operates over mental goals, states, and plans to reason about future scenarios. Finally, metamanagement mechanisms let an agent think about its own thoughts and experiences. AÞective experience is linked to interruption of some layers by others, with more sophisticated emotions occurring at higher levels.

- Emile (Gratch, 2000) provides an architectural account of emotions and their eÞect on behavior. Long-term knowledge includes Strips operators for use in plan generation and construal frames that specify conditions (relating events, expectations, goals, and standards) for eliciting diÞerent emotions. As the agent acquires new information about expected events, an appraisal module generates emotions in response, with initial intensity being a function of their probability and importance, but decaying over time. The agent's own emotions focuses eÞorts of the planning module and biases action selection, while inferences about other agents' emotions guide its dialogue choices.

- The Entropy Reduction Engine (Drummond et al., 1991) includes long-term memories for domain operators that describe the eÞects of actions, domain and behavioral constraints, situated control rules that propose actions to achieve goals, and reduction rules that decompose complex problems into simpler ones. The architecture uses its operators and constraints to produce temporal projections, which it then compiles into control rules

that a recognize-act cycles uses to determine which actions to execute. The projection process is supplemented by a problem reduction module, which uses the decomposition rules to constrain its search. Successful projections lead the system to learn new control rules, whereas prediction failures lead to revision of operators and domain constraints.

- EPIC (Meyer & Kieras, 1997) encodes long-term knowledge as production rules, organized as methods for accomplishing goals, that match against short-term elements in a variety of memories, including visual, auditory, and tactile buþers. Performance involves selecting matched rules and applying them in parallel to move eyes, control hands, or alter the contents of memory. Research on EPIC has included a strong emphasis on achieving quantitative þts to human behavior, especially on tasks that involve interacting with complex devices.

- FORR (Epstein, 1992) includes a declarative memory for facts and a procedural memory represented as a hierarchy of weighted heuristics. The architecture matches perceptions and facts against the conditions of heuristics, with matched structures proposing and rating candidate actions. Execution aþects the environment or changes the contents of declarative memory. Learning involves creating new facts and heuristics, adjusting weights, and restructuring the hierarchy based on facts and metaheuristics for accuracy, utility, risk, and speed.

- GLAIR (Shapiro & Ismail, 2003) stores content at a knowledge or cognitive level, a perceptual-motor level, and a sensori-actuator level. The highest layer includes generalized structures that deþne predicates in logical terms, with abstract concepts and procedures ultimately being grounded in perceptual features and behavioral routines at the middle layer. The system supports inference, belief revision, planning, execution, and natural language processing, with high-level beliefs being inferred from perceptions and with commands at the sensori-actuator level being derived from the agent's goals and plans.

- Icarus (Langley & Choi, 2006a; Langley et al., 2004) represents long-term knowledge in separate memories for hierarchical skills and concepts, with short-term beliefs, goals, and intentions cast as instances of these general structures. The performance element þrst infers all beliefs implied by its concepts and its perceptions of the environment, then selects an applicable path through the skill hierarchy to execute. Means-ends problem solving occurs when no skills relevant to the current goal are applicable, whereas learning creates new skills based on traces of successful problem solving.

- PolyScheme (Cassimatis et al., 2004) is a cognitive architecture designed to achieve human-level intelligence by integrating multiple representations,

reasoning methods, and problem-solving techniques. Each representation has an associated specialist module that supports forward inference, subgoaling, and other basic operations, which match against a shared dynamic memory with elements that are grounded in perception and action. PolyScheme make a stronger semantic commitment than most architectures, encoding all structures with a basic set of relations about time, space, events, identity, causality, and belief.

- Prodigy (Carbonell et al., 1990) encodes two kinds of long-term structures { domain operators that describe the eÞects of actions and control rules that specify when the system should select, reject, or prefer a given operator, binding, state, or goal. Short-term structures include descriptions of states and contents of a goal stack. Problem solving involves means-ends analysis, which repeatedly selects an operator to reduce differences between the current goal and state until it þnds a sequence that achieves the top-level goal. An explanation-based learning module analyzes problem-solving traces and creates new selection, rejection, and preference rules to reduce search on future tasks. Other modules control search by analogy with earlier solutions, learn operator descriptions from experimentation, and learn to improve the quality of solutions.

- PRS (Ingrand et al., 1992), which stands for Procedural Reasoning System, was an early architecture in the Beliefs-Desires-Intentions paradigm. The framework stores hierarchical procedures with conditions, eÞects, and ordered steps that invoke subprocedures. Dynamic structures include beliefs about the environment, desirs the agent wants to achieve, and intentions the agent plans to carry out. On each cycle, PRS decides whether to continue executing its current intention or to select a new intention to pursue.

- The Remote Agent architecture (Pell et al., 1998) was developed to control autonomous, mission-oriented spacecraft. Long-term structures include mission goals, possible activities and constraints on their execution, and qualitative models of the spacecraft's components, whereas dynamic structures include plans about which activities to pursue, schedules about when to carry them out, and inferences about the operating or failure modes. The architecture incorporates processes which retrieve high-level goals, generate plans and schedules that should achieve them, execute these schedules by calling low-level commands, monitor the modes of each spacecraft component, and recover in case of failures.

- RCS (Albus et al., 1992) is an architectural framework for developing intelligent physical agents. Expertise resides in a hierachical set of knowledge modules, each with its own long-term and short-term memories. Knowledge representation is heterogeneous, including frames, rules, images, and

maps. Modules operate in parallel, with a sensory interpreter examining the current state, a world model predicting future states, value judgement selecting among alternatives, and behavior generation carrying out tasks. Higher-level modules in uence their children in a top-down manner, whereas children pass information back up to their parent modules.

- Soar (Laird et al. 1987, Newell, 1990) encodes procedural long-term memory as production rules, whereas working memory memory contains a set of elements with attributes and values. The performance system matches productions against elements in working memory, and generates subgoals automatically when it cannot continue. When processing in the subgoal lets the agent overcome this impasse, the architecture adds a new chunk to long-term memory that summarizes the subgoal processing. In recent versions, episodic and semantic learning store working memory elements as structures in long-term memory, while reinforcement learning alters weights associated with rules that select operators.

- 3T (Bonasso et al., 1997) stores long-term knowledge in three layers or tiers. The lowest level consists of sensori-motor behaviors, which the architecture executes reactively, whereas the middle layer stores reactive action packages (Firby, 1994) that sequence these behaviors. The highest layer contains abstract operators, which a deliberative planner uses to generate a partial-order plan that the middle layer serializes and executes. In addition to this high-level plan, each skill and reactive action package has its own short-term memory. A predecessor of 3T, the Atlantis architecture (Gat, 1991), organized its knowledge and behavior in a very similar manner.