Cognitive Architectures and Multi-Agent Social Simulation

Ron Sun

Rensselaer Polytechnic Institute, Troy, NY 12180, USA
rsun@rpi.edu,
WWW home page: http://www.cogsci.rpi.edu/~rsun

Abstract. As we know, a cognitive architecture is a domain-generic computational cognitive model that may be used for a broad analysis of cognition and behavior. Cognitive architectures embody theories of cognition in computer algorithms and programs. Social simulation with multi-agent systems can benefit from incorporating cognitive architectures, as they provide a realistic basis for modeling individual agents (as argued in Sun 2001). In this survey, an example cognitive architecture will be given, and its application to social simulation will be sketched.

1 Defining Cognitive Architectures

As we know, a cognitive architecture is a broadly-scoped, domain-generic computational cognitive model, capturing essential structures and processes of the mind, to be used for a broad, multiple-level, multiple-domain analysis of cognition and behavior (Newell 1990, Sun 2002).

The architecture for a building consists of its overall framework and its overall design, as well as roofs, foundations, walls, windows, floors, and so on. Furniture and appliances can be easily rearranged and/or replaced and therefore they are not part of the architecture. By the same token, a cognitive architecture includes overall structures, essential divisions of modules, relations between modules, basic representations, essential algorithms, and a variety of other aspects (Sun 2004). In general, an architecture includes those aspects of a system that are relatively invariant across time, domains, and individuals. It deals with componential processes of cognition in a structurally and mechanistically well defined way.

For cognitive science (i.e., in relation to understanding the human mind), a cognitive architecture provides a concrete framework for more detailed modeling of cognitive phenomena, through specifying essential structures, divisions of modules, relations between modules, and so on. Its function is to provide an essential framework to facilitate more detailed modeling and understanding of various components and processes of the mind. Research in computational cognitive modeling explores the essence of cognition and various cognitive functionalities through developing detailed, process-based understanding by specifying computational models of mechanisms and processes. It embodies descriptions of cognition in computer algorithms and programs. That is, it produces runnable computational models. Detailed simulations are then conducted based on the computational models. In this enterprise, a cognitive architecture may be used for a broad, multiple-level, multiple-domain analysis of cognition.

In relation to building intelligent systems, a cognitive architecture specifies the underlying infrastructure for intelligent systems, which includes a variety of capabilities, modules, and subsystems. On that basis, application systems can be more easily developed. A cognitive architecture carries also with it theories of cognition and understanding of intelligence gained from studying the human mind. Therefore, the development of intelligent systems can be more cognitively grounded, which may be advantageous in many circumstances.

2 The Importance of Cognitive Architectures

This work is specifically concerned with psychologically oriented cognitive architectures (as opposed to software engineering oriented "cognitive" architectures): their importance and their applications. Psychologically oriented cognitive architectures are particularly important because (1) they are "intelligent" systems that are cognitively realistic (relatively speaking) and therefore they are more human-like in many ways, (2) they shed new light on human cognition and therefore they are useful tools for advancing the science of cognition, (3) furthermore, they may (in part) serve as a foundation for understanding collective human behavior and social phenomena (to be detailed later). Let us examine the importance of this type of cognitive architecture.

For cognitive science, the importance of such cognitive architectures lie in the fact that they are enormously useful in terms of understanding the human mind. In understanding cognitive phenomena, the use of computational simulation on the basis of cognitive architectures forces one to think in terms of process, and in terms of detail. Instead of using vague, purely conceptual theories, cognitive architectures force theoreticians to think clearly. They are critical tools in the study of the mind. Researchers who use cognitive architectures must specify a cognitive mechanism in sufficient detail to allow the resulting models to be implemented on computers and run as simulations. This approach requires that important elements of the models be spelled out explicitly, thus aiding in developing better, conceptually clearer theories.

An architecture serves as an initial set of assumptions to be used for further modeling of cognition. These assumptions, in reality, may be based on either available scientific data (for example, psychological or biological data), philosophical thoughts and arguments, or ad hoc working hypotheses (including computationally inspired such hypotheses). An architecture is useful and important precisely because it provides a comprehensive initial framework for further modeling in a variety of task domains.

Cognitive architectures also provide a deeper level of explanation. Instead of a model specifically designed for a specific task (often in an ad hoc way), using a cognitive architecture forces modelers to think in terms of the mechanisms and processes available within a generic cognitive architecture that are not specifically designed for a particular task, and thereby to generate explanations of the task that is not centered on superficial, high-level features of a task, that is, explanations of a deeper kind. To describe a task in terms of available mechanisms and processes of a cognitive architecture is to generate explanations centered on primitives of cognition as envisioned in the cognitive architecture, and therefore such explanations are deeper explanations. Because of the nature of such deeper explanations, this style of theorizing is also more likely to lead to unified explanations for a large variety of data and/or phenomena, because potentially a large variety of task data and phenomena can be explained on the basis of the same set of primitives provided by the same cognitive architecture. Therefore, using cognitive architectures leads to comprehensive theories of the mind (Newell 1990, Anderson and Lebiere 1998, Sun 2002).

On the other hand, for the fields of artificial intelligence and computational intelligence (AI/CI), the importance of cognitive architectures lies in the fact that they support the central goal of AI/CI—building artificial systems that are as capable as human beings. Cognitive architectures help us to reverse engineer the only truly intelligent system around—the human being, and in particular, the human mind. They constitute a solid basis for building truly intelligent systems, because they are well motivated by, and properly grounded in, existing cognitive research. The use of cognitive architectures in building intelligent systems may also facilitate the interaction between humans and artificially intelligent systems because of the similarity between humans and cognitively based intelligent systems.

3 Levels of Explanations

A broader perspective on the social and behavioral sciences may lead to a view of multiple "levels" of analysis encompassing multiple disciplines in the social and cognitive sciences. That is, a set of related disciplines, may be readily cast as a set of different levels of analysis, from the most macroscopic to the most microscopic. These different *levels* include: the sociological level, the psychological level, the componential level, and the physiological level. In other words, as has been argued in Sun et al (2005), one may view different disciplines as different levels of abstraction in the process of exploring essentially the same broad set of questions (cf. Newell 1990). See Figure 1.

level	object of analysis	type of analysis	model
1	inter-agent processes	social/cultural	collections of agent models
2	agents	psychological	individual agent models
3	intra-agent processes	componential	modular constr. of agent models
4	substrates	physiological	biological realization of modules

Fig. 1. A hierarchy of four levels.

First of all, there is the sociological level, which includes collective behaviors of agents, inter-agent processes, sociocultural processes, social structures and organizations, as well as interactions between agents and their (physical and sociocultural) environments. Although studied extensively by sociology, anthropology, political science, and economics, this level has traditionally been very much ignored in cognitive science. Only recently, cognitive science, as a whole, has come to grip with the fact that cognition is, at least in part, a sociocultural process. [1]

The next level is the psychological level, which covers individual experiences, individual behaviors, individual performance, as well as beliefs, concepts, and skills employed by individual agents. In relation to the sociological level, the relationship of individual beliefs, concepts, and skills with those of the society and the culture, and the processes of change of these beliefs, concepts, and skills, independent of or in relation to those of the society and the culture, may be investigated (in inter-related and mutually influential ways). At this level, one may examine human behavioral data, compared with models (which may be based on cognitive architectures) and with insights from the sociological level and details from the lower levels.

The third level is the componential level. At this level, one studies and models cognitive agents in terms of components (e.g., in the form of a cognitive architecture), with the theoretical language of a particular paradigm (for example, symbolic computation or connectionist networks, or their combinations thereof). At this level, one may specify computationally an overall architecture consisting of multiple components therein. One may also specify some essential computational processes of each component as well as essential connections among components. That is, one imputes a computational process onto a cognitive function. Ideas and data from the psychological level (that is, the psychological constraints from above), which bear significantly on the division of components and their possible implementations, are among the most important considerations. This level may also incorporate biological/physiological facts regarding plausible divisions and their implementations (that is, it can incorporate ideas from the next level down — the physiological level, which offers the biological constraints). This level results in *mechanisms* (though they are computational and thus somewhat abstract compared with physiological-level details). [2]

Although this level is essentially in terms of intra-agent processes, computational models (cognitive architectures) developed therein may be used to capture processes at higher levels, including interaction at a sociological level whereby multiple individuals are involved. This can be accomplished, for example, by examining interactions of multiple copies of individual agent models (based on the same cognitive architecture) or those of different individual agent models (based on different cognitive architectures). One may use computation as a means for

[1] See Sun (2001) for a more detailed argument for the relevance of sociocultural processes to cognition and vice versa.

[2] The importance of this level has been argued for, for example, in Anderson and Lebiere (1998), and Sun et al (2004).

constructing cognitive architectures at a sub-agent level (the componential level), but one may go up from there to the psychological level and to the sociological level (see the discussion regarding mixing levels in Sun et al 2005).

The lowest level of analysis is the physiological level, that is, the biological substrate, or the biological implementation, of computation. This level is the focus of a range of disciplines including biology, physiology, computational neuroscience, cognitive neuroscience, and so on. Although biological substrates are not our main concern here, they may nevertheless provide useful input as to what kind of computation is likely employed and what a plausible architecture (at a higher level) should be like. The main utility of this level is to facilitate analysis at higher levels, that is, analysis using low-level information to narrow down choices in selecting computational architectures as well as choices in implementing componential computation. [3]

In this enterprise of multiple levels in cognitive and social sciences, a cognitive architecture may serve as a centerpiece, tying together various strands of research. It may serve this purpose due to the comprehensiveness of its functionality and the depth with which it has been developed (at least for some psychologically oriented/grounded cognitive architectures). Thus, detailed mechanisms are developed within a cognitive architecture, which may be tied to low-level cognitive processes, while a cognitive architecture as a whole may function at a very high level of cognitive and social processes.

4 An Example Cognitive Architecture

4.1 Overview

Below a cognitive architecture CLARION will be described. It has been described extensively before in a series of previous papers, including Sun and Peterson (1998), Sun et al (2001), and Sun (2002, 2003).

CLARION is an integrative architecture, consisting of a number of distinct subsystems, with a dual representational structure in each subsystem (implicit versus explicit representations). Its subsystems include the action-centered subsystem (the ACS), the non-action-centered subsystem (the NACS), the motivational subsystem (the MS), and the meta-cognitive subsystem (the MCS). The role of the action-centered subsystem is to control actions, regardless of whether the actions are for external physical movements or for internal mental operations. The role of the non-action-centered subsystem is to maintain general knowledge, either implicit or explicit. The role of the motivational subsystem is to provide underlying motivations for perception, action, and cognition, in terms of providing impetus and feedback (e.g., indicating whether outcomes are satisfactory or

[3] Work at this level is basically the reverse-engineering of biological systems. In such a case, what needs to be done is to pinpoint the most basic primitives that are of relevance to the higher-level functioning that is of interest. (While many low-level details are highly significant, clearly not all low-level details are significant or even relevant.) After identifying proper primitives, one may study processes that involve those primitives, in mechanistic/computational terms.

not). The role of the meta-cognitive subsystem is to monitor, direct, and modify the operations of the action-centered subsystem dynamically as well as the operations of all the other subsystems.

Each of these interacting subsystems consists of two levels of representation (i.e., a dual representational structure): Generally, in each subsystem, the top level encodes explicit knowledge and the bottom level encodes implicit knowledge. The distinction of implicit and explicit knowledge has been amply argued for before (see Reber 1989, Cleeremans et al 1998, Sun 2002). The two levels interact, for example, by cooperating in actions, through a combination of the action recommendations from the two levels respectively, as well as by cooperating in learning through a bottom-up and a top-down process (to be discussed below). Essentially, it is a dual-process theory of mind. See Figure 2.

4.2 Details

The Action-Centered Subsystem First, let us focus on the action-centered subsystem (the ACS) of CLARION. The operation of the action-centered subsystem may be described as follows:

1. Observe the current state x.
2. Compute in the bottom level the Q-values of x associated with each of all the possible actions a_i's: $Q(x, a_1)$, $Q(x, a_2)$, ..., $Q(x, a_n)$.
3. Find out all the possible actions (b_1, b_2, ..., b_m) at the top level, based on the input x (sent up from the bottom level) and the rules in place.
4. Compare or combine the values of the selected a_i's with those of b_j's (sent down from the top level), and choose an appropriate action b.
5. Perform the action b, and observe the next state y and (possibly) the reinforcement r.
6. Update Q-values at the bottom level in accordance with the *Q-Learning-Backpropagation* algorithm
7. Update the rule network at the top level using the *Rule-Extraction-Refinement* algorithm.
8. Go back to Step 1.

In the bottom level of the action-centered subsystem, implicit reactive routines are learned: A Q-value is an evaluation of the "quality" of an action in a given state: $Q(x, a)$ indicates how desirable action a is in state x (which consists of some sensory input). The agent may choose an action in any state based on Q-values. To acquire the Q-values, the *Q-learning* algorithm (Watkins 1989) may be used, which is a reinforcement learning algorithm. It basically compares the values of successive actions and adjusts an evaluation function on that basis. It thereby develops sequential behaviors (Sun 2003).

The bottom level of the action-centered subsystem is modular; that is, a number of small neural networks co-exist each of which is adapted to specific modalities, tasks, or groups of input stimuli. This coincides with the modularity claim (Fodor 1983, Hirschfield and Gelman 1994) that much processing is done by limited, encapsulated (to some extent), specialized processors that are highly

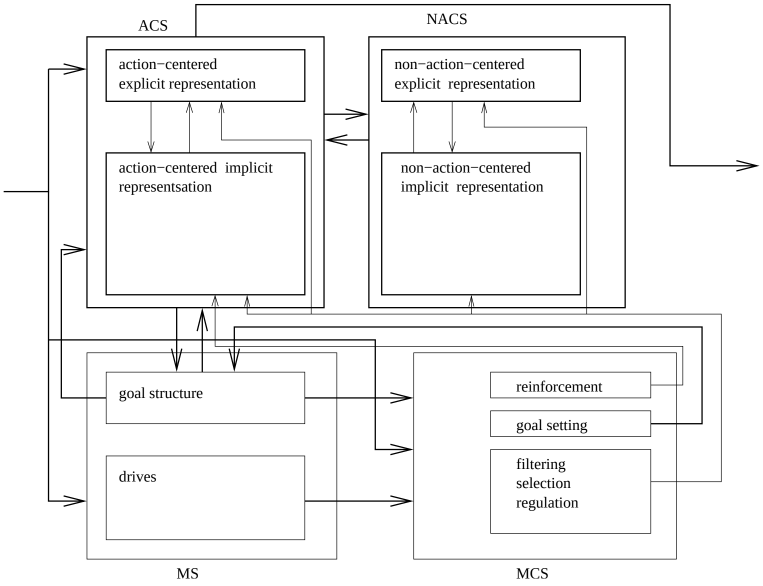
ACS
NACS
action−centered explicit representation
action−centered implicit representsation
non−action−centered explicit representation
non−action−centered implicit representation
goal structure
drives
reinforcement
goal setting
filtering selection regulation
MS
MCS

Fig. 2. The CLARION Architecture

efficient. These modules can be developed in interacting with the world (computationally through various decomposition methods; e.g., Sun and Peterson 1999). Some of them, however, are formed evolutionarily, that is, given a priori to agents, reflecting their hardwired instincts and propensities (Hirschfield and Gelman 1994).

In the top level of the action-centered subsystem, explicit conceptual knowledge is captured in the form of rules. See Sun (2003) for details. There are many ways in which explicit knowledge may be learned, including independent hypothesis-testing learning and "bottom-up learning" as discussed below.

Autonomous Generation of Explicit Conceptual Structures. Humans are generally able to learn implicit knowledge through trial and error, without necessarily utilizing a priori knowledge. On top of that, explicit knowledge can be acquired also from on-going experience in the world, through the mediation of implicit knowledge (i.e., bottom-up learning; see Sun 2002). The basic process of bottom-up learning is as follows: if an action implicitly decided by the bottom level is successful, then the agent extracts an explicit rule that corresponds to the action selected by the bottom level and adds the rule to the top level. Then, in subsequent interaction with the world, the agent verifies the extracted rule by considering the outcome of applying the rule: if the outcome is not successful, then the rule should be made more specific and exclusive of the current case; if the outcome is successful, the agent may try to generalize the rule to make it more universal (Michalski 1983). [4] After explicit rules have been learned, a variety of explicit reasoning methods may be used. Learning explicit conceptual representation at the top level can also be useful in enhancing learning of implicit reactive routines at the bottom level (e.g., Sun et al 2001).

Assimilation of Externally Given Conceptual Structures. Although CLARION can learn even when no a priori or externally provided knowledge is available, it can make use of it when such knowledge is available. To deal with instructed learning, externally provided knowledge, in the forms of explicit conceptual structures such as rules, plans, categories, and so on, can (1) be combined with existent conceptual structures at the top level (i.e., internalization), and (2) be assimilated into implicit reactive routines at the bottom level (i.e., assimilation). This process is known as top-down learning. See Sun (2003) for more details.

The Non-Action-Centered Subsystem The non-action-centered subsystem (NACS) may be used to represent general knowledge about the world, for performing various kinds of memory retrievals and inferences. Note that the non-action-centered subsystem is under the control of the action-centered subsystem (through its actions).

At the bottom level, "associative memory" networks encode non-action-centered implicit knowledge. Associations are formed by mapping an input to an output. The regular backpropagation learning algorithm can be used to establish such associations between pairs of inputs and outputs (Rumelhart et al 1986).

On the other hand, at the top level of the non-action-centered subsystem, a general knowledge store encodes explicit non-action-centered knowledge (cf. Sun 1994). In this network, chunks are specified through dimensional values. A node is set up in the top level to represent a chunk. The chunk node connects to its corresponding features represented as individual nodes in the bottom level of the non-action-centered subsystem. Additionally, links between chunks encode explicit associations between pairs of chunks, known as associative rules. Explicit associative rules may be formed (i.e., learned) in a variety of ways (Sun 2003).

In addition to applying associative rules, similarity-based reasoning may be employed in the non-action-centered subsystem. During reasoning, a known (given or inferred) chunk may be automatically compared with another chunk. If the similarity between them is sufficiently high, then the latter chunk is inferred (see Sun 2003 for details).

As in the action-centered subsystem, top-down or bottom-up learning may take place in the non-action-centered subsystem, either to extract explicit knowledge in the top level from the implicit knowledge in the bottom level or to assimilate explicit knowledge of the top level into implicit knowledge in the bottom level.

[4] The detail of the bottom-up learning algorithm can be found in Sun and Peterson (1998).

The Motivational and the Meta-Cognitive Subsystem The motivational subsystem (the MS) is concerned with drives and their interactions (Toates 1986), which leads to actions. It is concerned with why an agent does what it does. Simply saying that an agent chooses actions to maximizes gains, rewards, or payoffs leaves open the question of what determines these things. The relevance of the motivational subsystem to the action-centered subsystem lies primarily in the fact that it provides the context in which the goal and the payoff of the action-centered subsystem are set. It thereby influences the working of the action-centered subsystem, and by extension, the working of the non-action-centered subsystem.

A bipartite system of motivational representation is in place in CLARION. The explicit goals (such as "finding food") of an agent (which is tied to the working of the action-centered subsystem) may be generated based on internal drive states (for example, "being hungry"). (See Sun 2003 for details.)

Beyond low-level drives (concerning physiological needs), there are also higher-level drives. Some of them are primary, in the sense of being "hard-wired". For example, Maslow (1987) developed a set of these drives in the form of a "need hierarchy". While primary drives are built-in and relatively unalterable, there are also "derived" drives, which are secondary, changeable, and acquired mostly in the process of satisfying primary drives.

The meta-cognitive subsystem (the MCS) is closely tied to the motivational subsystem. The meta-cognitive subsystem monitors, controls, and regulates cognitive processes for the sake of improving cognitive performance (Nelson 1993). Control and regulation may be in the forms of setting goals for the action-centered subsystem, setting essential parameters of the action-centered subsystem and the non-action-centered subsystem, interrupting and changing on-going processes in the action-centered subsystem and the non-action-centered subsystem, and so on. Control and regulation can also be carried out through setting reinforcement functions for the action-centered subsystem. All of the above can be done on the basis of drive states and goals in the motivational subsystem. The meta-cognitive subsystem is also made up of two levels: th top level (explicit) and the bottom level (implicit).

5 A Cognitive Architecture in Social Simulation

One application of CLARION to social simulation is in understanding organizational decision making and the interaction between organizational structures and cognitive factors in affecting organizational decision making (Sun and Naveh 2004).

In terms of organizational structures, there are two major types: (1) teams, in which agents act autonomously, individual decisions are treated as votes, and the organizational decision is the majority decision; and (2) hierarchies, which are characterized by agents organized in a chain of command, such that information is passed from subordinates to superiors, and the decision of a superior is based solely on the recommendations of his/her subordinates. In addition, organizations are distinguished by the structure of information accessible by each agent. Two varieties of information access are: (1) distributed access, in which each agent sees a different subset of attributes (no two agents see the same subset of attributes), and (2) blocked access, in which several agents see exactly the same subset of attributes.

Several simulation models were considered in Carley et al (1998). The experiments by Carley et al (1998) were done in a 2 x 2 fashion (organization x information access). In addition, human data for the experiment were compared to the results of the four models (Carley et al 1998). [5] See Figure 3.

Agent/Org.	Team(B)	Team(D)	Hierarchy(B)	Hierarchy(D)
Human	50.0	56.7	46.7	55.0
Radar-Soar	73.3	63.3	63.3	53.3
CORP-P-ELM	78.3	71.7	40.0	36.7
CORP-ELM	88.3	85.0	45.0	50.0
CORP-SOP	81.7	85.0	81.7	85.0

Fig. 3. Human and simulation data for the organizational decision task. D indicates distributed information access, while B indicates blocked information access. All numbers are percent correct.

In their work, the agent models used were very simple, and the "intelligence" level in these models was low. Moreover, learning in these simulations was rudimentary: there was no complex learning process as one might observe in humans. With these shortcomings in mind, it is worthwhile to undertake a simulation that involves more complex agent models that more accurately capture human performance. Moreover, with the use of more cognitively realistic agent models, one may investigate individually the importance of different cognitive capacities and process details in affecting organizational performance (see Sun and Naveh 2004).

Hence, a simulation with CLARION used for modeling individual agents in an organization was conducted (Sun and Naveh 2004). The results (see Figure 4) closely accord with the patterns of the human data, with teams outperforming hierarchal structures, and distributed access proving superior to blocked access. Also, as in humans, performance is not grossly skewed towards one condition

[5] Among them, CORP-ELM produced the most probable classification based on an agent's own experience, CORP-P-ELM stochastically produced a classification in accordance with the estimate of the probability of each classification based on the agent's own experience, CORP-SOP followed organizationally prescribed standard operating procedure (which involved summing up the values of the attributes available to an agent) and thus was not adaptive, and Radar-Soar was a (somewhat) cognitive model built in Soar, which is based on explicit, elaborate search in problem spaces (Rosenbloom et al 1991).

Agent/Org.	Team(B)	Team(D)	Hierarchy(B)	Hierarchy(D)
Human	50.0	56.7	46.7	55.0
CLARION	53.2	59.3	45.0	49.4

Fig. 4. Simulation data for agents running for 3,000 cycles. The human data from Carley et al (1998) are reproduced here. Performance of CLARION is computed as percent correct over the last 1,000 cycles.

or the other, but is roughly comparable across all conditions, unlike some of the simulation results from Carley et al (1998). The match with the human data is far better than in the simulations conducted in Carley et al (1998). The better match is due, at least in part, to a higher degree of cognitive realism in our simulation. See Sun and Naveh (2004) for further details, including the interesting effects of varying cognitive parameters.

Another application of CLARION to social simulation is in capturing and explaining the essential process of publication in academic science and its relation to cognitive processes (Naveh and Sun 2006). Science develops in certain ways. In particular, it has been observed that the number of authors contributing a certain number of articles to a scientific journal follows a highly skewed distribution, corresponding to an inverse power law. In the case of scientific publication, the tendency of authorship to follow such a distribution was known as Lotka's law. Simon (1957) developed a simple stochastic process for approximating Lotka's law. One of the assumptions underlying this process is that the probability that a paper will be published by an author who has published i articles is equal to a/i^k, where a is a constant of proportionality. Using Simon's work as a starting point, Gilbert (1997) attempted to model Lotka's law. He obtains his simulation data based on some very simplified assumptions and a set of mathematical equations. To a significant extent, Gilbert's model is not cognitively realistic. The model assumes that authors are non-cognitive and interchangeable; it therefore neglects a host of cognitive phenomena that characterize scientific inquiry (e.g., learning, creativity, evolution of field expertise, etc.).

Using a more cognitively realistic model, one can address some of these omissions, as well as exploring other emergent properties of a cognitively based model and their correspondence to real-world phenomena. The results of the simulation based on CLARION (Naveh and Sun 2006) are shown in Figures 5 and 6, along with results (reported by Simon 1957) for *Chemical Abstracts* and *Econometrica*, and estimates obtained from previous simulations by Simon (1957) and Gilbert (1997). The figures in the tables indicate number of authors contributing to each journal, by number of papers each has published.

The CLARION simulation data for the two journals could be fit to the power curve $f(i) = a/i^k$, resulting in an excellent match. The results of the curve fit are shown in Figure 7, along with correlation and error measures (Naveh and Sun 2006).

# of Papers	Actual	Simon's	Gilbert's	CLARION
		estimate	simulation	simulation
1	3991	4050	4066	3803
2	1059	1160	1175	1228
3	493	522	526	637
4	287	288	302	436
5	184	179	176	245
6	131	120	122	200
7	113	86	93	154
8	85	64	63	163
9	64	49	50	55
10	65	38	45	18
11 or more	419	335	273	145

Fig. 5. Number of authors contributing to *Chemical Abstracts*.

# of Papers	Actual	Simon's	Gilbert's	CLARION
		estimate	simulation	simulation
1	436	453	458	418
2	107	119	120	135
3	61	51	51	70
4	40	27	27	48
5	14	16	17	27
6	23	11	9	22
7	6	7	7	17
8	11	5	6	18
9	1	4	4	6
10	0	3	2	2
11 or more	22	25	18	16

Fig. 6. Number of authors contributing to *Econometrica*.

Journal	a	k	Pearson R	R-square	RMSE
CA	3806	1.63	0.999	0.998	37.62
E	418	1.64	0.999	0.999	4.15

Fig. 7. Results of fitting CLARION data to power curves. CA stands for Chemical Abstracts and E stands for Econometrica.

Note that, in our simulation, the number of papers per author reflected the cognitive ability of an author, as opposed to being based on auxiliary assumptions such as those made by Gilbert (1997). This explains, in part, the greater divergence of our results from the human data: whereas Gilbert's simulation consists of equations selected to match the human data, our approach relies on much more detailed and lower-level mechanisms—namely, a cognitive agent model that is generic rather than task-specific. The result of the CLARION based simulation is therefore emergent, and not a result of specific and direct attempts to match the human data. That is, we put more distance between mechanisms and outcomes, which makes it harder to obtain a match with the human data. Thus, the fact that we were able to match the human data reasonably well shows the power of our cognitive architecture based approach.

6 Challenges Facing Cognitive Social Simulation

An important development in the social sciences has been that of agent-based social simulation (ABSS). This approach consists of instantiating a population of agents, allowing the agents to run, and observing the interactions between them. The use of agent-based social simulation as a means for computational study of societies mirrors the development of cognitive architectures in cognitive science. Thus, it is time to tackle sociality and social processes through cognitive architectures. So far, however, the two fields of social simulation and cognitive architectures have developed separately from each other (with some exceptions; e.g., Sun 2006). Thus, most of the work in social simulation assumes very rudimentary cognition on the part of the agents.

The two fields of social simulation and cognitive architectures can be profitably integrated. This is an important challenge. As has been argued before (Sun and Naveh 2004), social processes ultimately rest on the choices and decisions of individuals, and thus understanding the mechanisms of individual cognition can lead to better theories describing the behavior of aggregates of individuals. Although most agent models in social simulation have been extremely simple, a more realistic cognitive agent model, incorporating realistic tendencies, inclinations and capabilities of individual cognitive agents can serve as a more realistic basis for understanding the interaction of individuals. [6]

At the same time, by integrating social simulation and cognitive modeling, one can arrive at a better understanding of individual cognition. Traditional approaches to cognitive modeling have largely ignored the potentially decisive effects of socially acquired and disseminated knowledge (including language, norms, and so on). By modeling cognitive agents in a social context, one can learn more about the sociocultural processes that influence individual cognition.

The most fundamental challenge in this regard is to develop better ways of conducting detailed social simulation based on cognitive architectures as basic

[6] Although some cognitive details may ultimately prove to be irrelevant, this cannot be determined *a priori*, and thus simulations are useful in determining which aspects of cognition can be safely abstracted away.

building blocks. This is not an easy task. Although some initial work has been done (e.g., Sun and Naveh 2004, Sun 2006), much more work is needed.

One specific challenge is how to enhance cognitive architectures for the purpose of accounting for sociality in individual cognitive agents. There are many questions in this regard. For example, what are the characteristics of a proper cognitive architecture for modeling the interaction of cognitive agents? What additional sociocultural representations (for example, "motive", "obligation", or "norm") are needed in cognitive modeling of multi-agent interaction? See, for example, Sun (2006) for further discussions.

There is also the challenge of computational complexity and thus scalability that needs to be addressed. Social simulation could involve a large number of agents, up to thousands. Computational complexity is thus already high, even without involving cognitive architectures as agent models. To incorporate cognitive architectures into social simulation, one has to deal with a great deal of added complexity. Thus, scalability is a significant issue.

7 Conclusion

The field of cognitive architectures will have a profound impact on cognitive science as well as on social simulations, both in terms of better understanding cognition and in terms of better understanding sociality. As such, a significant amount of collective research effort should be put into it.

Acknowledgments

The work surveyed here was supported in part by ARI grants DASW01-00-K-0012 and W74V8H-05-K-0002 (to Ron Sun and Bob Mathews). Thanks are due to Isaac (Yizchak) Naveh, Xi Zhang, and Bob Mathews for their collaborations on related research.

References

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

K. Carley, M. Prietula, and Z. Lin, (1998). Design versus cognition: The interaction of agent cognition and organizational design on organizational performance. *Journal of Artificial Societies and Social Simulation*, 1 (3).

A. Cleeremans, A. Destrebecqz and M. Boyer, (1998). Implicit learning: News from the front. *Trends in Cognitive Sciences*, 2 (10), 406-416.

J. Fodor, (1983). *The Modularity of Mind.* MIT Press, Cambridge, MA.

N. Gilbert, (1997), A simulation of the structure of academic science. *Sociological Research Online*, 2(2). Available online at
http://www.socresonline.org.uk/socresonline/2/2/3.html.

L. Hirschfield and S. Gelman (eds.), (1994). *Mapping the Mind: Domain Specificity in Cognition and Culture*. Cambridge University Press, Cambridge, UK.

D. Marr, (1982). *Vision*. W.H. Freeman: New York.

A. Maslow, (1987). *Motivation and Personality*. 3rd Edition. Harper and Row, New York.

I. Naveh and R. Sun, (2006). A cognitively based simulation of academic science. *Computational and Mathematical Organization Theory*, Vol.12, No.4, pp.313-337.

T. Nelson, (Ed.) (1993). *Metacognition: Core Readings*. Allyn and Bacon.

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

H. Simon, (1957), *Models of Man, Social and Rational*. Wiley, NY.

R. Sun, (1994). *Integrating Rules and Connectionism for Robust Commonsense Reasoning*. John Wiley and Sons, New York, NY.

R. Sun, (2001). Cognitive science meets multi-agent systems: A prolegomenon. *Philosophical Psychology*, 14 (1), 5-28.

R. Sun, (2002). *Duality of the Mind*. Lawrence Erlbaum Associates, Mahwah, NJ.

R. Sun, (2003). *A Tutorial on CLARION*. Technical report, Cognitive Science Department, Rensselaer Polytechnic Institute.
http://www.cogsci.rpi.edu/∼rsun/sun.tutorial.pdf

R. Sun, (2004). Desiderata for cognitive architectures. *Philosophical Psychology*, 17 (3), 341-373.

R. Sun, (2006). Prolegomena to integrating cognitive modeling and social simulation. In: R. Sun (ed.), *Cognition and Multi-Agent Interaction: From Cognitive Modeling to Social Simulation*. Cambridge University Press, New York.

R. Sun, L. A. Coward, and M. J. Zenzen, (2005). On levels of cognitive modeling. *Philosophical Psychology*, 18 (5), 613-637.

R. Sun and I. Naveh, (2004). Simulating organizational decision-making using a cognitively realistic agent model. *Journal of Artificial Societies and Social Simulation*, 7 (3). http://jasss.soc.surrey.ac.uk/7/3/5.html

R. Sun and T. Peterson, (1998). Autonomous learning of sequential tasks: experiments and analyses. *IEEE Transactions on Neural Networks*, 9 (6), 1217-1234.

R. Sun and T. Peterson, (1999). Multi-agent reinforcement learning: Weighting and partitioning. *Neural Networks*, 12 (4-5). 127-153.

C. Watkins, (1989). *Learning with Delayed Rewards*. Ph.D Thesis, Cambridge University, Cambridge, UK.