Modeling Meta-Cognition in a Cognitive Architecture

Ron Sun

Cognitive Science Department, Rensselaer Polytechnic Institute, Troy, NY 12180, USA

rsun@rpi.edu

Xi Zhang

Department of Computer Science, University of Missouri, Columbia, MO 65211, USA

Robert Mathews

Psychology Department, Louisiana State University, Baton Rouge, LA 70803, USA.

October 27, 2005

Keywords: cognitive modeling, metacognition, neural networks

Abstract

This paper describes how meta-cognitive processes (i.e., the self monitoring and regulating of cognitive processes) may be captured within a cognitive architecture CLARION. Some currently popular cognitive architectures lack sufficiently complex built-in meta-cognitive mechanisms. However, a sufficiently complex meta-cognitive mechanism is important, in that it is an essential part of cognition and without it, human cognition may not function properly. We contend that such a meta-cognitive mechanism should be an integral part of a cognitive architecture. Thus such a mechanism has been developed within the CLARION cognitive architecture. The paper demonstrates how human data of two meta-cognitive experiments are simulated using CLARION. The simulations show that the meta-cognitive processes represented by the experimental data (and beyond) can be adequately captured within the CLARION framework.

1 Introduction

According to Flavell (1976), meta-cognition refers to "one's knowledge concerning one's own cognitive processes and products or anything related to them." Meta-cognition includes "the active monitoring and consequent regulation and orchestration of these processes in relation to the cognitive objects or data on which they bear, usually in the service of some concrete goal or objective." It has been extensively studied in cognitive psychology and in other related fields.

In cognitive psychology, meta-cognitive processes have been routinely conceived as being carried out by specialized mechanisms that are separate and standalone for the specific purpose of monitoring and controlling regular cognitive processes. Such a conception has been explicitly stated in some theoretical treatments (e.g., Nelson and Narens 1990, Darling et al 1998), as well as implied in many experimental designs (e.g., Metcalfe 1994, Schneider 1998).

Moreover, meta-cognitive processes have often been portrayed as explicit processes that involve deliberate reasoning (Metcalfe and Shimamura 1994, Mazzoni and Nelson 1998). However, evidence has been mounting that meta-cognitive processes may not be entirely explicit. For example, Reder and Schunn (1996) argued that there were likely to be implicit processes, for the simple reason of avoiding using up limited cognitive resources (such as attention) and interfering with regular processes. Thus, they argued that, while meta-cognitive strategies themselves might be explicit, and/or explicitly learned, the selection (and use) of meta-cognitive strategies was implicit. We have reasons to believe that meta-cognitive knowledge is neither necessarily explicit, nor necessarily implicit (Sun and Mathews 2003). Meta-cognition is likely a combination of implicit and explicit processes, the same as regular cognitive processes, as has been argued amply before (Sun 1999, 2002, Sun et al 2001, Sun et al 2005, Mathews et al 1989, Wegner and Bargh 1998).

If meta-cognitive processes are not entirely explicit, then can they still be separate and standalone processes? Although it is not inconceivable that meta-cognition constitutes a distinct and separate process of cognition, it may not be necessary to emphasize such a separation, at least not with regard to the meta-cognitive functioning, which includes monitoring, controlling, verbal reporting, and goal setting, all of which require the interaction of different types of processes (more discussions later).

In relation to computational cognitive modeling, it is worth noting that in many currently popular cognitive architectures, there is the lack of a sophisticated and sufficiently complex built-in meta-cognitive mechanism. However, a sophisticated and sufficiently complex meta-cognitive mechanism is important. It is an essential part of cognition, and without it, human cognition may not function properly (Mazzoni and Nelson 1998, Metcalfe and Shimamura 1994). Therefore, we believe that such a complex meta-cognitive mechanism should be an integral part of a cognitive architecture.

In this paper, we will develop a theoretical framework of meta-cognition in the context of an overall architecture of the mind — the Clarion cognitive architecture. We will then use the architecture to construct models of specific meta-cognitive processes, which are then used to capture experimental data related to meta-cognition. Such simulations serve to validate the models used in the simulations (to some preliminary extent).

In the remainder of this paper, section 2 describes a set of pertinent meta-cognitive experiments. Section 3 then turns to describe the cognitive architecture Clarion, to be used for simulating the data, which is rather novel compared with most other popular cognitive architectures in that it explicitly includes a meta-cognitive subsystem. Section 3 describes model setups for simulations, and results of simulation runs, in relation to the experimental tasks described earlier. A discussion section follows after that. A summary section then completes the paper.

2 Meta-Cognitive Experiments and Data

Let us look into two experiments as examples of meta-cognition related human experiments. The first experiment captures meta-cognitive monitoring, while the second further captures both meta-cognitive monitoring and meta-cognitive intervention (control and regulation).

2.1 Meta-Cognitive Monitoring

In the task of Metcalfe (1986), subjects were given a sheet of paper that described a story. They were asked to solve the puzzle in the story. They were told to write down a number between 0 and 10, whereby 0 meant that they were "cold" about the solution (i.e., they had no idea at all about the solution) and 10 meant that they were certain that they had the right solution. They were supposed to do so every 10s at the sound of a click. When the subjects had achieved a solution, they were to write it down on a piece of paper. 134 subjects (undergraduate introductory psychology students) were tested.

43 subjects got the right solution and 44 subjects came up with wrong answers. In general, subjects who came up with the correct solutions gave lower warmth ratings than did subjects with the incorrect

solutions. ANOVA (correct × incorrect) showed $F(1, 50) = 2.81, p = 0.09$, when the last three ratings were used. However, if we only look at the last two warmth ratings before reaching a solution, this effect was significant, $F(1, 50) = 6.48, p < 0.05$. If we only look at the last warmth rating before reaching a solution, this effect was also significant, $F(1, 68) = 15.00, p < 0.05$.

Warmth rating reflects meta-cognitive monitoring — keeping an eye on one's own cognitive processes. However, the difference in warmth rating was highly counter-intuitive — we would normally expect that subjects who came up with the correct solutions gave higher warmth ratings than did subjects with the incorrect solutions, but the result was the exact opposite. The question is how this result should be explained; in particular, we would want to know how this result should be explained mechanistically (computationally), in a process sense, within the general framework of a cognitive architecture.

2.2 "Lack of Knowledge" Inferences

In this case (Gentner and Collins 1991), instead of dealing with numerical data, we deal with protocols that indicated meta-cognitive reasoning. Here is an example from Gentner and Collins (1991):

> Q: Have you ever shaken hands with Richard Nixon?
>
> A: No. ...... How do I know? It's not something that one would forget. I don't think I've ever seen him live, in person. I'm sure I haven't." (He went on describing meetings with some other presidents.)

Another essentially similar example was from Collins (1978):

> Q: Is the Nile longer than the Mekong river?
>
> A: I think so. ...... Because in junior high, I read a book on rivers. ...... the Amazon was in there and the Nile was in there, and they were big, and long, and important. The Mekong wasn't in there.

Yet another example was also from Collins (1978):

> Q: Is Kissinger 6'6" tall?
>
> A: If Kissinger were 6'6" tall, I would know he is very tall. I don't, so he must not be that tall.

In all of these examples, inferences were made based on (1) the lack of knowledge about something, as well as (2) the importance/significance of that knowledge. In each of these cases, because of the combined reason of the lack of knowledge of a particular event/fact on the one hand and the significance of that event/fact on the other, an inference was made that the event/fact was not true.

In order to make such inferences, first of all, meta-cognitive monitoring of one's own reasoning process is necessary, the same as in the previous task. However, beyond meta-cognitive monitoring (as in the previous task), active meta-cognitive intervention is also necessary. Based on information from monitoring one's own cognitive processes (such as the lack of a crucial piece of information), a meta-cognitive process intervenes and (re)-directs the reasoning process, leading to the conclusion that something is not true.

Beside the above protocol data, Gentner and Collins (1981) presented experimental data from a third-person view of this meta-cognitive reasoning process. Assuming that a protagonist in a story forgot about an event, they asked subjects to rate the likelihood of the event, which was either of low or high importance. This scenario was essentially identical to the above protocol segments, except that in this experiment, there was an additional projection of one's own meta-cognitive process onto others. In this experiment, the likelihood ratings were found to be inversely correlated with the importance of events, reflecting meta-cognitive monitoring and intervention during reasoning.

3 Meta-Cognition in Clarion

3.1 The Overall Architecture

Overall, Clarion is an integrative cognitive architecture, consisting of a number of distinct subsystems, with a dual representational structure (implicit and explicit) in each. 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 ACS is to control actions, regardless of whether actions are for external physical movements or internal mental operations. The role of the NACS is to maintain general knowledge, either implicit or explicit (conceptual). The role of the MS 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 not). The role of the MCS is to monitor, direct, and modify the operations of the ACS dynamically as well as the operations of all the other subsystems.

Each of these subsystems consists in turn of two levels of representation (a dual representational structure): Generally, the top level encodes explicit knowledge and the bottom level encodes implicit knowledge. Let us consider the representational forms that need to be present for encoding these two different types of knowledge. Notice the fact that the relatively inaccessible nature of implicit knowledge may be captured by subsymbolic, distributed representation provided, for example, by a backpropagation network (Rumelhart et al 1986). This is because distributed representational units (in the hidden layers of a backpropagation network) are capable of accomplishing computations but are subsymbolic and generally not individually meaningful (Rumelhart et al 1986, Sun 1994). This characteristic of distributed representation, which renders this representational form less accessible, accords well with the relative inaccessibility of implicit knowledge (Reber 1989, Seger 1994, Cleeremans et al 1998). In contrast, explicit knowledge may be captured in computational modeling by symbolic or localist representation (Clark and Karmiloff-Smith 1993), in which each unit is more easily interpretable and has a clearer conceptual meaning. This characteristic of symbolic or localist representation captures the characteristic of explicit knowledge being more accessible and more manipulable (Sun 1994). [1]

Let us turn to learning. First, there is the learning of implicit knowledge at the bottom level. One way of implementing a mapping function to capture implicit knowledge is to use a multi-layer neural network (e.g., a three-layer backpropagation network). Adjusting parameters of this mapping function (to change input/output mappings; that is, to learn implicit knowledge) may be carried out in ways consistent with the nature of distributed representation (e.g., as involved in backpropagation networks). Often, reinforcement learning can be used (Sun et al 2001), especially Q-learning (Watkins 1989), implemented using backpropagation networks (Rumelhart et al 1986). In this learning setting, there is no need for external teachers providing desired input/output mappings. This (implicit) learning method may be cognitively justified. For instance, Cleeremans (1997) argued at length that implicit learning could not be captured by symbolic models but neural networks. Sun (1999) made similar arguments.

Explicit knowledge at the top level can also be learned in a variety of ways (in accordance with localist/symbolic representation used there). Because of the representational characteristics, one-shot

[1] Accessibility here refers to the direct and immediate availability of mental content for the major operations that are responsible for, or concomitant with, consciousness, such as introspection, forming higher-order thoughts, and verbal reporting, and so on. The dichotomous difference in the representations of the two different types of knowledge leads to a two-level architecture, whereby each level uses one kind of representation and captures one corresponding type of process (either implicit or explicit). See Sun (2002) for further arguments.

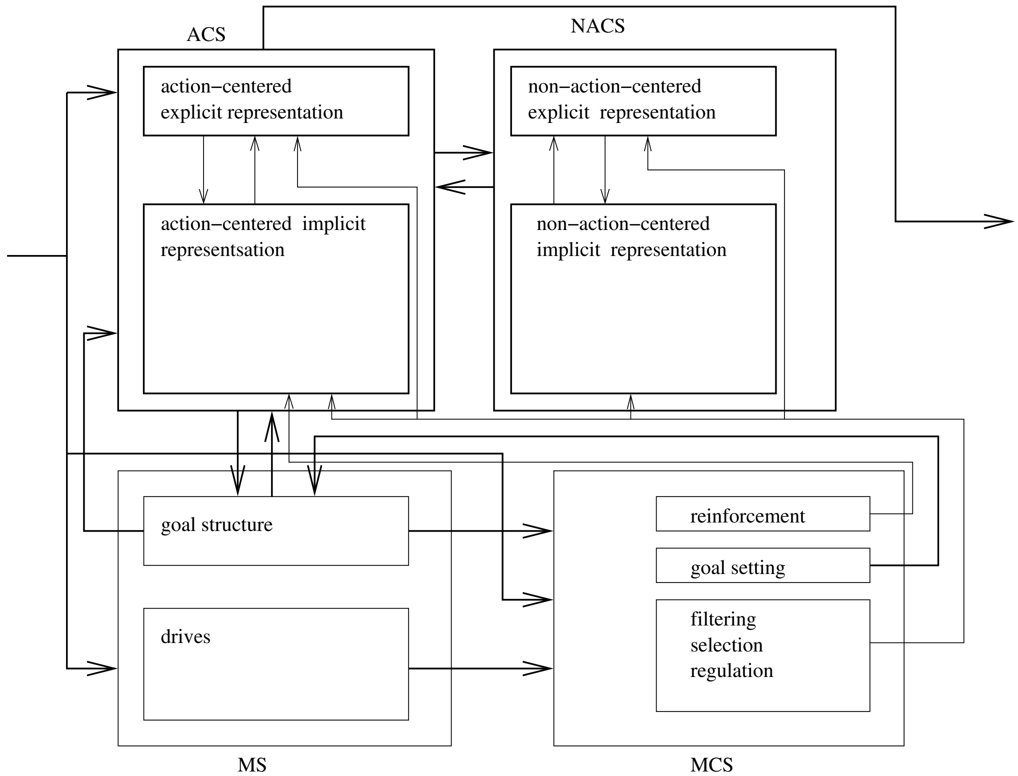

Figure 1: The CLARION architecture.

learning (for example, based on hypothesis testing) is more likely (Bruner et al 1956, Busemeyer and Myung 1992, Sun et al 2001). With such learning, an agent explores the world, and dynamically acquires explicit representations and modify them as needed.

Note that the implicit knowledge already acquired in the bottom level may be utilized in learning explicit knowledge at the top level, through *bottom-up learning* (Sun et al 2001). That is, information accumulated in the bottom level through interacting with the world may be used for extracting and then refining explicit knowledge. Conceivably, other types of learning of explicit knowledge are also possible, such as explicit hypothesis testing without the help of the bottom level. Moreover, once explicit knowledge is established at the top level, it may be assimilated into the bottom level. The assimilation process, known as *top-down learning* (as opposed to bottom-up learning), may be carried out in a variety of ways. See Sun (2003) for details.

Figure 1 contains a sketch of this basic architecture of a cognitive agent, which includes the four major subsystems. The following three sections will sketch, one by one and in some more detail, these subsystems of Clarion. Due to lengths, only major aspects of Clarion will be described; other details will not be covered.

3.2 The Action-Centered Subsystem

The overall algorithm for action decision making in the action-centered subsystem (the ACS) of CLARION is as follows, where the bottom level is named the IDNs (the Implicit Decision Networks) and the top level the ARS (the Action Rule Store):

1. Observe the current state x.
2. Compute in the bottom level (the IDNs) the "value" of each of the possible actions (a_i's) associated with the state x: $Q(x, a_1)$, $Q(x, a_2)$, ..., $Q(x, a_n)$. Stochastically choose one action according to Q values.
3. Find out all the possible actions (b_1, b_2, ..., b_m) at the top level (the ARS), based on the the current state x (which goes up from the bottom level) and the existing rules in place at the top level. Stochastically choose one action.
4. Choose an appropriate action a, by stochastically selecting the outcome of either the top level or the bottom level.
5. Perform the action a, and observe the next state y.
6. Update the bottom level in accordance with an appropriate algorithm (to be detailed later), based on the feedback information.
7. Update the top level using an appropriate algorithm (for extracting, refining, and deleting rules, to be detailed later).
8. Go back to Step 1.

The input (x) to the bottom level consists of three groups of information: (1) sensory input, (2) working memory items, (3) the selected item of the goal structure. The sensory input is divided into a number of input dimensions, each of which has a number of possible values. The goal input is also divided into a number of dimensions (one of which is the goal dimension, the values of which are possible goals and at most one of them can be activated at one time). The working memory is divided into dimensions as well. Thus, input state x is represented as a set of dimension-value pairs: $(dim_1, val_1)(dim_2, val_2)....(dim_n, val_n)$.

The output of the bottom level is the action choice. It consists of three groups of actions: working memory actions, goal actions, and external actions. (These three groups may be computed by separate networks.)

In each network (encoding implicit knowledge), actions are selected based on Q values. 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. At each step, given the state x, we compute the Q values of all the actions (i.e., $Q(x, a)$ for all a's). We then use the Q values to decide probabilistically on an action to be performed, through a Boltzmann distribution of Q values:

$$p(a|x) = \frac{e^{Q(x,a)/\alpha}}{\sum_i e^{Q(x,a_i)/\alpha}} \tag{1}$$

where α controls the degree of randomness ("temperature") of the decision-making process. (This method is also known as Luce's choice axiom; Watkins 1989.) The *Q-learning* algorithm (Watkins 1989), a reinforcement learning algorithm, is used for learning Q values (i.e., learning implicit knowledge at the bottom level). Q values are gradually tuned, on-line, through successive updating, to enable sequential behavior to emerge. Q-learning is implemented in backpropagation networks (Sun and Peterson 1998). Since Q-learning is not directly relevant to this work, see Sun (2003) for further details.

Next, explicit knowledge at the top level (the ARS) is captured by chunks and action rules. The condition of an action rule, similar to the input to the bottom level, consists of three groups: sensory input, working memory items, and the current goal. The output of a rule, similar to the output from the bottom level, is an action choice. It may be one of the three types: working memory actions, goal actions, and external actions, or their combinations thereof.

Specifically, rules are of the following form: *current-state-specification* $\longrightarrow$ *action*. The left-hand side of a rule is a conjunction (i.e., logic AND) of individual elements. Each element refers to a dimension x_i of the input x, specifying a value range, in the form of $x_i \in (v_{i1}, v_{i2}, ..., v_{in})$. The right-hand side of a rule is an action recommendation a.

We translate the structure of a set of rules into that of a network at the top level. Note that each value of each input dimension is represented by an individual node at the bottom level. Those nodes relevant to the condition of a rule are connected to the node at the top level representing that condition, known as a *chunk* node. Given a set of rules, a rule network can be wired up at the top level, in which conditions and conclusions of rules are represented by respective chunk nodes, and links representing rules are established that connect corresponding pairs of chunk nodes. For action decision making, at each step, one rule is randomly selected out of all the action rules that match the current state x. The selected rule gets to decide the action recommendation from the top level.

To capture the bottom-up learning process (Stanley et al 1989, Karmiloff-Smith 1996), the *Rule-Extraction-Refinement* algorithm (RER) learns action rules at the top level using information in the bottom level. The basic idea of bottom-up learning is as follows: If an action chosen (by the bottom level) is successful (i.e., it satisfies a certain criterion), then an explicit action rule is extracted. Then, in subsequent interactions with the world, the rule is refined by considering the outcome of applying the rule: If the outcome is successful, the condition of the rule may be generalized to make it more universal; if the outcome is not successful, then the condition of the rule should be made more specific and exclusive of the current case. An agent needs a rational basis for making these above decisions, which is the *information gain* measure (Sun and Peterson 1998). The details of the operations used in the algorithm (including extraction, generalization, and specialization) and the numerical criteria measuring whether a result is successful or not (used in deciding whether or not to apply some of these operators) can be found in Sun (2003). Since they are not directly relevant to this work, we will not get into the details.

There are also fixed rules (or FRs for short, which are not from bottom-up learning), representing prior explicit knowledge acquired through prior experience in the world or given from external sources (for example, via instructions or textbooks). Externally given fixed rules may lead to top-down learning (assimilation of explicit knowledge into an implicit form). For example, with explicit knowledge (in the form of action rules) in place at the top level, the bottom level learns under the guidance of the rules. That is, initially, the agent relies mostly on the rules at the top level for its action decision making. But gradually, when more and more knowledge is acquired by the bottom level through "observing" actions directed by the top-level rules, the agent becomes more and more reliant on the bottom level (given that the inter-level stochastic selection mechanism is adaptable). Hence, top-down learning takes place.

For combining the outcomes of the two levels, at each step, with probability P_{TL}, if there is at least one rule indicating an action in the current state, we use the outcome from the rule set; otherwise, we use the outcome of the bottom level (which is always available). With the probability $P_{BL}(= 1 - P_{TL})$, we use the outcome of the bottom level. The selection probabilities may be variable, determined through a process known as "probability matching": that is, the probability of selecting a component is determined based on the relative success ratio of that component.

Note that afore-mentioned working memory (WM for short) is for storing information temporarily for the purpose of facilitating subsequent decision making (Baddeley 1986). Working memory actions are used either for storing an item in the working memory, or for removing an item from the working memory. Goal structures (GS for short) are special cases of working memory, and are for storing goal information specifically.

3.3 The Non-Action-Centered Subsystem

At the bottom level of the non-action-centered subsystem, the "associative memory" networks (the AMNs for short) 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 (the GKS) encodes explicit non-action-centered knowledge (cf. Sun 1994). In this network, *chunks* are specified through dimensional values (as in the ACS). A node is set up in the GKS to represent a chunk. The node connects to its corresponding features represented as individual nodes in the bottom level. Additionally, links between chunks encode associations between pairs of chunks, known as *associative rules*. Associative rules may be formed (i.e., learned) in a variety of ways (see Sun 2003).

The support for an associative rule is calculated as follows (Sun 1994):

$$S_j^a = \sum_i S_i^c * W_i^a \tag{2}$$

where j indicates the jth rule at the top level, S_j^a is the support for associative rule j, S_i^c is the strength of the ith chunk in the condition of the rule, i ranges over all the chunks in the condition of rule j, and W_i^a is the weight of the ith chunk in the condition of rule j (which, by default, is $W_i^a = 1/n$, where n is the number of chunks in the condition of the associative rule).

The chunk representing the conclusion of the rule has a strength level that is determined by the maximum of the support from all the relevant rules (pointing to that conclusion chunk):

$$S_{c_k}^{c,a} = \max_{j:\text{all associative rules leading to } c_k} S_j^a \tag{3}$$

where $S_{c_k}^{c,a}$ is the strength of chunk c_k (resulting from associative rules), and j ranges over all the associative rules pointing to c_k.

On top of such rule-based reasoning, similarity-based reasoning (SBR) may be employed in the NACS. An agent may compare a known (given or inferred) chunk with another chunk. If the similarity between them is sufficiently high, then the latter chunk is inferred. The strength of a chunk c_j as a result of such similarity-based reasoning is:

$$S^{c,s}_{c_j} = \max_i (S_{c_i \sim c_j} \times S^c_{c_i}) \tag{4}$$

where $S^{c,s}_{c_j}$ denotes the strength of chunk j resulting from similarity, $S_{c_i \sim c_j}$ measures the similarity from c_i to c_j, $S_{c_i \sim c_j} \times S^c_{c_i}$ measures the support to c_j from the similarity with c_i, and i ranges over all chunks.

The similarity measure by default is:

$$S_{c_i \sim c_j} = \frac{N_{c_i \cap c_j}}{f(N_{c_j})} \tag{5}$$

where $S_{c_i \sim c_j}$ denotes the similarity from c_i to c_j, $N_{c_i \cap c_j}$ is the number of identically valued dimensions of c_i and c_j (among all the specified dimensions), N_{c_j} is the number of the (specified) dimensions of c_j, and f is a superlinear (but close to linear) function, such as $f(x) = x^{1.10}$. Thus, the similarity measure is limited to $[0, 1)$. Essentially, this equation measures similarities based on the proportion of overlapping features (Tversky 1977).

Similarity-based and rule-based reasoning (SBR and RBR) can be inter-mixed. When similarity-based reasoning is employed, extending the strength calculation of rule-based reasoning explained earlier, we have:

$$\begin{aligned} S^c_{c_i} &= \max(S^{c,a}_{c_i}, S^{c,s}_{c_i}) \\ &= \max\Big(\max_{j:\text{all associative rules leading to } c_i} S^a_j, \\ &\qquad \max_{j:\text{all chunks similar to } c_i} (S_{c_j \sim c_i} \times S^c_{c_j})\Big) \end{aligned} \tag{6}$$

where $S_{c_j \sim c_i}$ is the similarity measure. That is, rule-based reasoning and similarity-based reasoning compete with each other (Sun 1994). [2]

Note that a chunk in the GKS may be "activated" by both a source within the GKS and the result of the AMNs. When the result from the output side of the AMNs is sent bottom-up to the GKS, it "activates" all chunks compatible with it. This activation combines with the activation from within the GKS by a MAX function.

As in the ACS, top-down or bottom-up learning may take place in the NACS, either to extract explicit knowledge of the GKS from the implicit knowledge in the AMNs or to assimilate explicit knowledge of the GKS into implicit knowledge in the AMNs (see Sun 2003).

[2] As a result of mixing similarity-based and rule-based reasoning, complex patterns of reasoning can emerge. As shown by Sun (1994), different sequences of mixed similarity-based and rule-based reasoning capture essential patterns of human everyday (mundane, commonsense) reasoning, including various forms of "inheritance reasoning".

3.4 The Meta-Cognitive and the Motivational Subsystem

Supervisory processes in Clarion are made up of two subsystems: the motivational subsystem and the meta-cognitive subsystem. The motivational subsystem (the MS) is concerned with drives and their interactions (Toates 1986). That is, it is concerned with why an agent does what it does. Simply saying that an agent chooses actions to maximizes gains, rewards, payoffs, or reinforcements leaves open the question of what determines gains, rewards, payoffs, or reinforcements. The relevance of the motivational subsystem to the main component, the ACS, lies primarily in the fact that it provides the context in which the goal and the reinforcement of the ACS are set. It thereby influences the working of the ACS, and by extension, the working of the NACS. See Sun (2003) for details. Since this part is not directly relevant to this work, we will not get into more details (but see Sun 2003 for further details).

On the other hand, meta-cognition concerns one's knowledge of, and control over, one's own cognitive processes and outcomes. It involves active monitoring as well as active regulation/control of these processes, usually in the service of some goal (Flavell 1976). This notion of meta-cognition is operationalized within Clarion. The meta-cognitive subsystem (the MCS) monitors and controls/regulates cognitive processes for the sake of improving cognitive performance in various circumstances. Control and regulation may be in the forms of setting goals for the ACS, setting essential parameters of the ACS and the NACS, interrupting and changing on-going processes in the ACS and the NACS, and so on. Control and regulation may also be carried out through setting reinforcement functions (on the basis of motivational states).

Specifically, in this subsystem, several types of meta-cognitive processes are available, for different meta-cognitive monitoring and control/regulation purposes. Among them, there are the following types:

(1) behavioral aiming:
setting of reinforcement functions
setting of goals

(2) information filtering:
focusing of input dimensions in the ACS
focusing of input dimensions in the NACS

(3) information acquisition:

selection of learning methods in the ACS

selection of learning methods in the NACS

(4) information utilization:

selection of reasoning methods in the ACS

selection of reasoning methods in the NACS

(5) outcome selection:

selection of output dimensions in the ACS

selection of output dimensions in the NACS

(6) cognitive mode selection:

selection of explicit processing, implicit processing, or a combination thereof (with proper integration parameters), in the ACS

(7) setting parameters of the ACS and the NACS:

setting of parameters for the IDNs

setting of parameters for the ARS

setting of parameters for the AMNs

setting of parameters for the GKS

Structurally, the subsystem may be subdivided into a number of modules. The bottom level consists of the following (separate) networks: the goal setting network, the reinforcement network, the input selection network, the output selection network, the parameter setting network (for setting learning rates, temperatures, etc.), and so on. In a similar fashion, the rules at the top level (if they exist) can be correspondingly subdivided. See Figure 2 for a diagram of the MCS.

The monitoring buffer contains several sections of information: the ACS performance section, the NACS performance section, the ACS learning section, the NACS learning section, and other sections. Each section contains information about both the bottom level and the top level of a subsystem. Most relevant to this work, in each "performance" section, the information about a subsystem includes not only the strengths of the conclusions, but also the *relative strengths* of the conclusions, which concern how distinguished or certain the conclusions are in relation to other competing ones. [3] A relative

[3]Details of other sections can be found in Sun (2003).

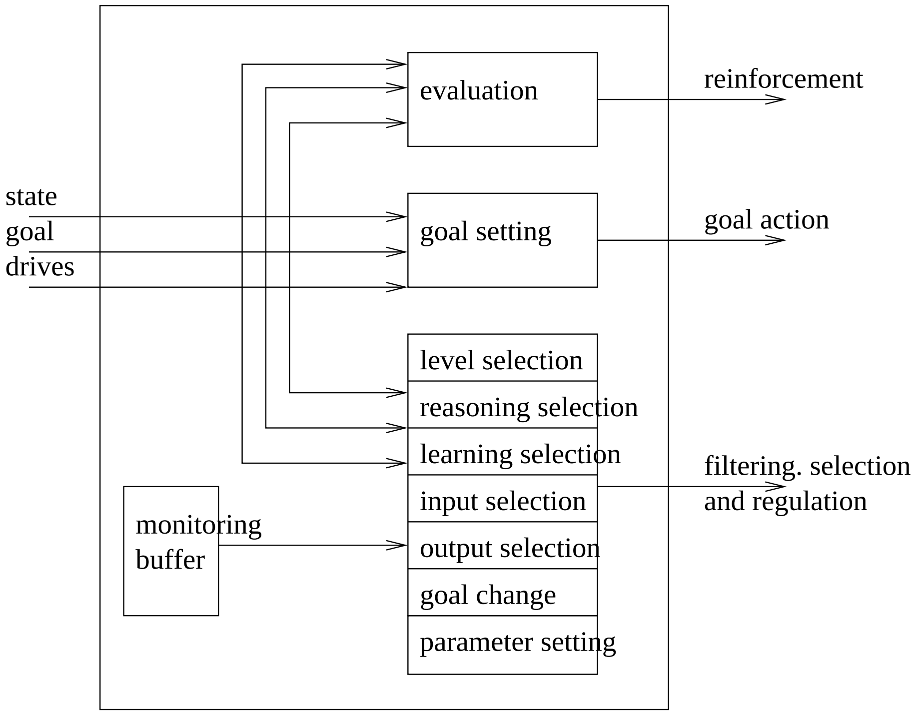

Figure 2: The structure of the meta-cognitive subsystem.

strength is defined as:

$$RS_i = \frac{S_i}{\sum_j S_j} \tag{7}$$

Other aspects of the MCS, such as setting reinforcement functions (based on motivational states), setting goals (based on motivational states), information selection, and so on, can be found in Sun (2003). This subsystem may be pre-trained before the simulation of any particular task (to capture evolutionary pre-wired instincts or previously acquired knowledge and skills).

4 Simulations of Meta-Cognitive Data

In this section, we describe each of the two simulations of the two experiments described earlier, which capture meta-cognitive monitoring as well as control/regulation.

However, before that, it is worth noting that CLARION has been successful in simulating a variety of cognitive tasks. These tasks include serial reaction time tasks, artificial grammar learning tasks, process control tasks, categorical inference tasks, alphabetical arithmetic tasks, discovery tasks, and Tower of Hanoi (Sun 2002, Sun et al 2005). In addition, extensive work have been done on a complex minefield navigation task (Sun et al 2001). Therefore, we are now in a good position to extend the effort on Clarion to the capturing of a range of meta-cognitive phenomena.

4.1 Simulations of Meta-Cognitive Monitoring

In this simulation, we capture meta-cognitive monitoring, and provide a detailed computational explanation of the (counter-intuitive) experimental results of Metcalfe (1986).

Model Setup. The explanation of the experiment of Metcalfe (1986) on which this simulation was based was that when a subject came up with multiple plausible explanations and had to evaluate their relative merits, his/her subjective certainty on the conclusions (a meta-cognitive judgment) would be relatively low due to the co-existence of multiple plausible explanations. Hence a lower warmth rating was produced. But, in this way, the subject was more likely to come up with a correct (the most plausible) explanation eventually. On the other hand, when a subject came up with only one plausible explanation, there was no need to evaluate multiple possibilities, and thus his/her subjective certainty, and also his/her "warmth" rating, would be higher, but that sole explanation was more likely to be wrong, because of the ambiguity of the situation and the lack of careful evaluation of all possibilities on the part of the subject (Metcalfe 1986).

In Clarion, the ACS, the NACS, and the MCS were involved in the simulation of this task. The NACS performed inferences under the control of the ACS. Through the monitoring buffer, the MCS monitored the progress of inferences in the NACS (and might also perform meta-cognitive control when needed, although not in this task).

More specifically, the goal of performing "regular inference" was set up first by the MCS (before it all began). The MCS then selected relevant input dimensions to be used in reasoning in the NACS, which excluded all information not relevant to the task at hand (for example, there was contextual input information, such as time and location, that was not relevant to the task). The MCS also selected the reasoning method to be used in the NACS, in this case, "forward chaining with SBR".

Among other things, the monitoring buffer in the MCS kept track of how clear-cut the conclusions reached by the NACS were. The NACS section of the buffer recorded the relative strengths of n most highly activated conclusions (as explained earlier). When that part of the buffer reported that there was one conclusion that stood out with a high relative strength, the conclusion was considered certain and its "warmth" level was high. Otherwise, the conclusions were less certain, and the "warmth" levels were lower. Hence, "warmth" was captured in this simulation by the relative strengths in the monitoring buffer.

The ACS directed the reasoning of the NACS (with the pre-selected reasoning method). The following rules were implemented in the top level of the ACS:

> If goal= regular-inference, then perform one step of inference in the NACS (using the method selected by the MCS and the information filtered by the MCS).
>
> If goal= regular-inference, and chunk i is a conclusion chunk with $S_i^c > threshold_S$ and $\forall_j S_i^c > S_j^c$, then retrieve chunk i.
>
> If goal= warmth-reporting, then report the "warmth" (RS) of the chosen chunk from the monitoring buffer in the MCS.

where RS stood for relative strength, and S for strength. The threshold for strengths was set at $threshold_S = 0.1$. These rules constituted a priori knowledge in this simulation.

Although the bottom level of the ACS (the IDNs) was present, it had little effect. This was due to the stochastic selection of levels in favor of the top level (due to the setting of the cross-level integration parameters), which was the result of the task instructions, which led to performing a rather explicit inference task. (Therefore, details of the IDNs were irrelevant and are not describe further.)

At the top level of the NACS (the GKS), relevant knowledge was encoded as associative rules. Some subjects (those who turned out to have higher warmth ratings) had few of these rules, while other subjects (those who turned out to have lower warmth ratings) had more of these rules. In relation to the domain of this experiment, rules were of the following form:

> If event A1 happens, then B11 might be the answer
>
> If event A1 happens, then B12 might be the answer
>
> If event A2 happens, then B21 might be the answer
>
> If event A2 happens, then B22 might be the answer

Note that in the top level of the NACS (the GKS), similarity-based reasoning (as explained earlier) was occurring throughout the simulation in addition to rule-based reasoning.

At the bottom level of the NACS, one AMN was present. The network was trained with the same knowledge as embodied by the associative rules in the GKS.

Simulation Results. In our simulation, in general, those variants of the models that generated the correct solutions gave lower warmth ratings than those that generated the incorrect solutions. Thus, the simulation model (as explained earlier) within the general framework of CLARION accounted for the counter-intuitive findings from the experimental data of Metcalfe (1986).

In the simulation data, during the last two ratings, there were significant differences between the two groups of simulated subjects. At the last rating, the average warmth rating of the simulated subjects with the correct solutions was 3.327, while that of the simulated subjects with the incorrect solutions was 5.241. At the rating preceding the last, the average warmth rating of the simulated subjects with the correct solutions was 3.299, while that of the simulated subjects with the incorrect solutions was 5.085. An ANOVA of the warmth data of the last rating showed that there was a significant difference between correct versus incorrect ($F(1, 298) = 92.808, MSE = 273.196, p < 0.0001$). Similarly, an ANOVA of the warmth data of the penultimate rating showed that there was also a significant difference between correct versus incorrect ($F(1, 292) = 78.049, MSE = 232.681, p < 0.0001$).

The earlier explanation of the data pattern of this experiment was confirmed by the simulation. That is, when a subject initially came up with multiple plausible explanations (when multiple relevant rules were available), his/her subjective certainty on the conclusions was low due to the co-existence of multiple plausible explanations. Thus a lower warmth rating was produced. However, the subject in this case was more likely to come up with a correct explanation, based on evaluations of the relative merits of different explanations. On the other hand, if a subject initially came up with only one plausible explanation (when only one relevant rule was available), there was no need to evaluate multiple possibilities, and thus his/her subjective certainty was higher, which led to a higher "warmth" rating. But that sole explanation produced was more likely to be wrong, because of the ambiguity of the situation and the lack of careful evaluation of all possibilities.

4.2 Simulating "Lack of Knowledge" Inferences

In this simulation, we intended to also capture a kind of meta-cognitive reasoning from the lack of a certain piece of knowledge, that is, to capture both meta-cognitive monitoring and meta-cognitive intervention (control/regulation).

As we know, a negative conclusion is often drawn from the lack of knowledge. It is reasonable to assume that a negative conclusion is drawn, if subjects think that they know enough about a domain and yet they do not know about a particular fact in that domain. If they do not know enough about a domain when they do not know about a particular fact in that domain, a negative conclusion is not likely to be drawn.

Model Setup. The ACS, the NACS, and the MCS were involved in this simulation. The NACS performed inferences under the control of the ACS. The MCS selected information to be used and reasoning methods to be applied. More importantly, the MCS also monitored the progress of inferences in the NACS, and performed meta-cognitive intervention accordingly, including starting "lack-of-knowledge" inferences.

Specifically, the goal of performing "regular inference" was first set up by the MCS. The MCS then selected relevant input dimensions to be used in reasoning within the NACS, which excluded all information not relevant to the task at hand (as before, there was contextual input information, such as time and location, that was not relevant to the task). Then the MCS selected the reasoning method to be used in the NACS, in this case, "forward chaining with SBR".

When the "lack-of-knowledge" condition was detected (as indicated by the uniformly low activation in the NACS performance section of the monitoring buffer of the MCS), the MCS initiated the "lack-of-knowledge" inferences, by pushing the goal of *LOK-inference* onto the goal structure. The LOK inferences were carried out by the ACS in consultation with the NACS.

As always, the ACS directed the reasoning of the NACS (using the pre-selected reasoning method). The following rules were used in the top level of the ACS for directing reasoning of the NACS:

> If goal= regular-inference, the perform one step of inference in the NACS (using the method selected by the MCS and the information filtered by the MCS).
>
> If goal= regular-inference, and chunk i is a conclusion chunk with $S_i^c > threshold_S$ and $\forall_j S_i^c > S_j^c$, then retrieve chunk i.
>
> If goal = LOK-inference, there is no conclusion chunk with $S_i^c > threshold_S$ but there are many associative rules pointing to the conclusion chunk, then indicate that the conclusion is negative.
>
> If goal = LOK-inference, there is no conclusion chunk with $S_i^c > threshold_S$ and there are not many associative rules pointing to the conclusion chunk, then indicate that the conclusion is indeterminate.

where S represented strength. The threshold for S was set at $threshold_S = 0.1$. Another threshold determined how many rules constituted "many", which was domain specific. In this simulation, we assumed a value of 2. These rules constituted a priori knowledge for our simulation.

In the ACS, although the bottom level of the ACS (the IDNs) was present, it had little effect. Stochastic selection in favor of the top level (the ARS) was used, which was the result of the task instructions, which led to performing a rather explicit inference task. (Therefore, details of the IDNs were irrelevant and are not described here.)

At the top level of the NACS (the GKS), all relevant knowledge was encoded as associative rules. The associative rules relevant to this task domain were generally of the following form:

River A $\longrightarrow$ long river

River B $\longrightarrow$ long river

River C $\longrightarrow$ long river

plus many other rules that were not relevant to this task.

At the bottom level of the NACS, one AMN was present. As before, the AMN was trained with the same knowledge as the associative rules in the GKS.

Simulation Results. Our simulation successfully captured the lack-of-knowledge inference as exhibited by the human subjects in the protocols described earlier.

As predicted, when a simulated subject had a (relatively) large amount of knowledge about a conclusion but could not reach that conclusion in a particular instance, then the lack-of-knowledge inference was initiated and a negative answer was produced. On the other hand, when a simulated subject had a (relatively) small amount of knowledge about a conclusion and could not reach that conclusion in a particular instance, then the lack-of-knowledge inference was not employed and no conclusion was given. A large number of simulation runs testified to this phenomenon.

In all of these cases, meta-cognitive monitoring enabled meta-cognitive intervention, which led to the lack-of-knowledge inferences. Thus, this simulation demonstrated not only meta-cognitive monitoring but also meta-cognitive intervention (control and regulation).

5 Discussions

The simulations in this work help to better understand issues related to meta-cognition, which may well be one of the central issues of cognition and may have significant implications for further advances in cognitive theories and for further development of cognitive architectures (Reder 1996, Anderson and Lebiere 1998, Lovett et al 2000, Sun et al 2001). Notably, some currently popularly cognitive architectures do not include a sufficiently complex meta-cognitive component (such as ACT-R; Anderson and Lebiere 1998). Thus, work in this area is not only useful but very much needed. The meta-cognitive subsystem developed in this work may also be applied to some other existing cognitive architectures.

In Clarion, we view the relationship among different types of cognitive processes as highly interactive. For example, regular cognitive processes in the ACS and the NACS interact continuously with the meta-cognitive subsystem. In particular, the MCS receives information from other subsystems on a constant basis and may intervene in those other subsystems at any time. The ACS and the NACS also interact constantly with each other and the operations in the NACS are mostly directed by the ACS. Given this highly interactive relationship among subsystems, we may as well view meta-cognitive processes as one with regular processes, since they are intimately tied together. Therefore, the issue of whether there is a separate and standalone meta-cognitive process, according to Clarion, is a red herring.

Compared with other existing cognitive architectures, Clarion clearly has much better developed meta-cognitive processes. For example, we may compare Clarion with ACT-R (Anderson and Lebiere 1998). In ACT-R, meta-cognitive control and regulation are often accomplished through manually adjusting parameter settings (Lovett et al 2000). Although meta-cognitive reasoning processes may be implemented using ACT-R's production rules, the range of meta-cognitive processes that can be implemented using ACT-R is limited, because there is no built-in mechanism specifically designed to capture meta-cognitive processes in a precise way. Our simulation results, for instance, would be difficult for ACT-R to replicate. An ACT-R model of this sort of task would end up calculating the relative strengths of the relevant productions (or chunks, depending on implementations), but the ACT-R model, by design, would not have access to the results of this calculation, making it very difficult to show the same behavior.

We can also compare Clarion with SOAR (Rosenbloom et al 1993). In SOAR, like in ACT-R, some meta-cognitive processes may be implemented using production rules, but they would be very limited. They are also not properly distinguished from regular processes and delimited in terms of their specific characteristics.

In contrast, Clarion has a set of specifically designed meta-cognitive mechanisms for monitoring, controlling, and regulating cognitive processes. The meta-cognitive subsystem may filter/select information, adjust cognitive parameters, or intervene in regular cognitive processes. Also, in contrast to these two other cognitive architectures, in Clarion, meta-cognitive processes are architecturally specified to a large extent. They are specifically modeled to the extent that we believe is appropriate: They are not totally undelimited, as in the other architectures. But they are also not completely fixed and thus they are not inflexible. As our understanding of meta-cognitive processes grows, these meta-cognitive mechanisms in Clarion may be further refined, and tailored to capture the exact range and scope of human meta-cognitive processes.

One counter-argument that may arise would be that, although the other architectures do not have built-in meta-cognitive mechanisms, at least some of those architectures allow meta-cognition to occur on the basis of the regular cognitive mechanisms, and it is the more constrained architectures that provide deeper explanations, rather than those with a larger pool of specific mechanisms for specific classes of phenomena. However, this counter-argument ignored the fact that there are severe limitations in those other architectures in terms of the range of meta-cognitive phenomena those other architectures can capture. If an architecture fails to capture the breadth of meta-cognitive phenomena, then there is very little to be gained in being "constrained", because no "deep" explanations come out of it when it cannot capture most of the phenomena.

As indicated by our earlier discussion of simulation results, Clarion, through simulation, succeeded in explaining, computationally in a process-based way, the counter-intuitive results in the experimental data of Metcalfe (1986) (the lower warmth ratings from the subjects who found the correct answers compared with the subjects who failed to do so). The explanation (based on the amount of relevant knowledge) naturally fell out of the processes embodied by Clarion. Similarly, the lack-of-knowledge inferences also naturally fell out of the processes embodied by Clarion.

Lastly, Norman and Shallice's (1986) view is somewhat akin to our view here. They posited the co-existence of two kinds of meta-cognitive processes: (1) fast, automatic processes, which are triggered by stimuli and are inflexible; (2) slow, conscious processes, which are independent of stimuli and are flexible. The former is used in skilled performance, while the latter deals mostly with novel situations. Although our main conception of the explicit/implicit dichotomy (which applies to the meta-cognitive subsystem as well as the other subsystems) is somewhat akin to Norman and Shallice's (1986) view, we further addressed more complex forms of implicit and explicit meta-cognitive mechanisms and processes, which they did not specify, and advocated intermeshed meta-cognitive and regular cognitive processes, which differ somewhat from their view.

6 Concluding Remarks

In summary, simulations of meta-cognitive monitoring and control/regulation have been conducted based on the built-in meta-cognitive mechanisms in the cognitive architecture CLARION. The cognitive architecture contains rather detailed meta-cognitive mechanisms and thus makes simulations of meta-cognitive processes easier to construct, less ad hoc, and more uniform. That is, meta-cognitive processes are, more or less, architecturally specified in CLARION. This approach has been shown to be viable for cognitive modeling.

The afore-described simulations captured rather accurately the experimental data of Metcalfe (1986) and Gentner and Collins (1981). To some extent, the afore-described simulations validated our approach as embodied by CLARION.

One last point that may be derived from the CLARION cognitive architecture and the simulations based on the architecture is that meta-cognitive processes are intermeshed with regular processes. They interact with each other on a constant basis. Thus, they may be viewed either as separate or as integrated. The two views together are descriptive of the CLARION perspective on meta-cognition.

ACKNOWLEDGMENT: This work is supported in part by Army Research Institute contract DASW01-00-K-0012 and W74V8H-04-K-0002. Thanks to the reviewers who provided comments on an earlier version.

References

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

A. Baddeley, (1986). *Working Memory.* Oxford University Press, New York.

J. Bruner, J. Goodnow, and J. Austin, (1956). *A Study of Thinking.* Wiley, New York.

J. Busemeyer and I. Myung, (1992). An adaptive approach to human decision making: Learning theory, decision theory, and human performance. *Journal of Experimental Psychology: General*, 121 (2), 177-194.

A. Clark and A. Karmiloff-Smith, (1993). The cognizer's innards: a psychological and philosophical perspective on the development of thought. *Mind and Language.* 8 (4), 487-519.

A. Cleeremans, (1997). Principles for implicit learning. In: D. Berry (Ed.), *How Implicit Is Implicit Learning?* 195-234. Oxford, England: Oxford University Press.

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

A. Collins, (1978). Fragments of a theory of human plausible reasoning. In: D. Waltz (ed), *Theoretical Issues in Natural Language Processing II.* Ablex, Norwood, NJ.

S. Darling, S. Sala, C. Gray, and C. Trivelli, (1998). Putative functions of the prefrontal cortex. In: G. Mazzoni and T. Nelson, (eds.) *Metacognition and Cognitive Neuropsychology.* Erlbaum, Mahwah, NJ.

J. Flavell, (1976). Metacognitive aspects of problem solving. In: B. Resnick (ed.), *The Nature of Intelligence.* Erlbaum, Hillsdale, NJ.

D. Gentner and A. Collins, (1981). Studies of inference from lack of knowledge. *Memory and Cognition*, 9, 434-443.

A. Karmiloff-Smith, (1986). From meta-processes to conscious access: Evidence from children's metalinguistic and repair data. *Cognition.* 23. 95-147.

M. Lovett, L. Daily and L. Reder, (2000). A source activation theory of working memory: cross-task prediction of performance in ACT-R. *Cognitive Systems Research*, 1 (2), 99-118.

R. Mathews, R. Buss, W. Stanley, F. Blanchard-Fields, J. Cho, and B. Druhan, (1989). Role of implicit and explicit processes in learning from examples: a synergistic effect. *Journal of Experimental Psychology: Learning, Memory and Cognition.* 15, 1083-1100.

G. Mazzoni and T. Nelson, (eds.) (1998). *Metacognition and Cognitive Neuropsychology.* Erlbaum, Mahwah, NJ.

J. Metcalfe, (1986). Dynamic metacognitive monitoring during problem solving. *Journal of Experimental Psychology: Learning, Memory and Cognition*, 12, 623-634.

J. Metcalfe, (1994). A computational modeling approach to novelty monitoring, metacognition, and frontal lobe dysfunction. In: J. Metcalfe and A. Shimamura, (eds.) *Metacognition: Knowing about Knowing.* MIT Press, Cambridge, MA.

J. Metcalfe and A. Shimamura, (eds.) (1994). *Metacognition: Knowing about Knowing.* MIT Press, Cambridge, MA.

T. Nelson and L. Narens, (1990). Meta-memory: a theoretical treatment and new findings. In: G.

Bower, (ed.) *The Psychology of Learning and Motivation.* Vol.26, 125-140. Academic Press, New York.

R. Nisbett and T. Wilson, (1977). Telling more than we can know: Verbal reports on mental processes. *Psychological Review.* 84 (3), 1977.

D. Norman and T. Shallice, (1986). Attention to action: willed and automatic control of behavior. In: G. Schwartz and D. Shapiro, (eds.) *Consciousness and Self Regulation: Advances in Research and Theory*, vol.4, 1-18. Plenum, New York.

A. Reber, (1989). Implicit learning and tacit knowledge. *Journal of Experimental Psychology: General.* 118 (3), 219-235.

L. Reder, (ed.) (1996). *Implicit Memory and Metacognition.* Erlbaum, Mahwah, NJ.

L. Reder and C. Schunn, (1996). Metacognition does not imply awareness: strategy choice is governed by implicit learning and memory. In: L. Reder, (ed.) *Implicit Memory and Metacognition.* Erlbaum, Mahwah, NJ.

P. Rosenbloom, J. Laird, and A. Newell, (1993). *The SOAR Papers: Research on Integrated Intelligence.* MIT Press, Cambridge, MA.

D. Rumelhart, J. McClelland and the PDP Research Group, (1986). *Parallel Distributed Processing: Explorations in the Microstructures of Cognition*, MIT Press, Cambridge, MA.

W. Schneider, (1998). The development of procedural metamemory in childhood and adolescence. In: G. Mazzoni and T. Nelson, (eds.) *Metacognition and Cognitive Neuropsychology.* Erlbaum, Mahwah, NJ.

C. Seger, (1994). Implicit learning. *Psychological Bulletin.* 115 (2), 163-196.

T. Shallice and W. Burgess, (1991). Higher order cognitive impairments and frontal lobe damage in man. In: H. Levin, H. Eisenberg, and A. Benton, (eds.) *Frontal Lobe Function and Dysfunction*, 125-138. Oxford University Press, New York.

W. Stanley, R. Mathews, R. Buss, and S. Kotler-Cope, (1989). Insight without awareness: On the interaction of verbalization, instruction and practice in a simulated process control task. *Quarterly Journal of Experimental Psychology.* 41A (3), 553-577.

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

R. Sun, (1995). Robust reasoning: integrating rule-based and similarity-based reasoning. *Artificial*

Intelligence, 75 (2), 241-296.

R. Sun, (1999). Accounting for the computational basis of consciousness: A connectionist approach. *Consciousness and Cognition*, 8, 529-565.

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, Troy, New York. http://www.cogsci.rpi.edu/∼rsun/tutorial.pdf

R. Sun and R. Mathews, (2003). Explicit and implicit processes of metacognition. In: S. Shohov (eds.), *Advances in Psychology Research*, Vol.22, pp.3-18. Nova Science Publishers, Hauppauge, NY.

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

R. Sun and T. Peterson, (1998). Autonomous learning of sequential tasks: Experiments and analyses. *IEEE Transactions on Neural Networks*, 9 (6), 1215-1234.

R. Sun, P. Slusarz, and C. Terry, (2005). The interaction of the explicit and the implicit in skill learning: A dual-process approach. *Psychological Review*, 112 (1), 159-192.

F. Toates, (1986). *Motivational Systems*. Cambridge University Press, Cambridge, UK.

A. Tversky, (1977). Features of similarity. *Psychological Review*, 84(4), 327-352.

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

D. Wegner and J. Bargh, (1998). Control and automaticity in social life. In: D. Gilbert, S. T. Fiske, and G. Lindzey (eds.), *Handbook of Social Psychology* (4th ed), Vol. 1, pp.446-496. New York: McGraw-Hill.