Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Papers

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Acquisition and Transfer of Declarative and Procedural Knowledge

Todd R. Johnson
Department of Pathology
The Ohio State University
Columbus, OH 43228
614-292-3284
johnson.25@osu.edu

ABSTRACT

Recent results in cognitive skill acquisition suggest that task speed-up can be due to either speed-up of procedural knowledge or speed-up of the retrieval of declarative knowledge. This paper presents a single Act-R model that closely fits the data of two learning and transfer experiments conducted by Rabinowitz and Goldberg (1995). These experiments test three main hypotheses: 1) access to procedural and declarative knowledge speeds up as separate power laws of practice; 2) training on a large variety of problems leads to strengthening of procedural knowledge, whereas training on a small set of problems leads to the acquisition and strengthening of declarative knowledge; and 3) procedural knowledge operates in one direction only—from condition to action—whereas declarative knowledge can be cued by any of its elements. The model provides a good fit to the data, further validating Act-R as a model of the human cognitive architecture

Keywords

Declarative memory, procedural memory, learning, transfer, knowledge compilation, Act-R, Soar.

INTRODUCTION

One common view of cognitive skill acquisition is that it progresses from an interpretive stage to a procedural stage using some kind of knowledge compilation mechanism (Stillings et al., 1995; VanLehn, 1989). Such a mechanism produces procedural knowledge from the results of more deliberate, interpretive problem solving. This view has received a lot of empirical support. Several researchers have shown that knowledge compilation can model the transition from novice to expert behavior (Larkin, 1981; Newell & Rosenbloom, 1981). One major research effort, the Soar architecture, even asserts that knowledge compilation is the only mechanism required to account for all human learning (Newell, 1990). Researchers using Soar have been able to model a wide range of learning strategies (Miller, 1993; Rosenbloom & Aasman, 1990; Steier et al., 1987). Knowledge compilation mechanisms can also sometimes account for the ubiquitous power law of learning (Newell & Rosenbloom, 1981).

Recent results on the characteristics of declarative and procedural knowledge, however, threaten the simplicity of this view of skill acquisition, because they suggest that cognitive skill can also improve through the acquisition and strengthening of declarative memory elements (for a review see (VanLehn, 1996)). A number of experiments have suggested that the retrieval of declarative knowledge and the application of procedural knowledge speed up as separate power laws of practice. In other words, the time to retrieve a declarative memory speeds up as a power function of the number of retrievals, whereas the time to apply a procedure speeds up as a power function of the number of applications. This implies that cognitive skill can improve by acquiring and strengthening procedural or declarative knowledge, or some combination of the two.

Despite the intuitive nature of the distinction between declarative and procedural knowledge, the hypothesis that there are separate long-term memory stores for declarative and procedural knowledge remains a controversial issue in cognitive science. The controversy arises because, in theory, anything that can be modeled with two distinct long-term stores can also be modeled using only a procedural long-term store. For example, long-term procedural knowledge might add "Washington, DC" to working memory whenever working memory encodes a goal to determine the capitol of the United States. Working memory is widely thought to be a declarative store, so the declarative-procedural distinction applies only to long-term memory.

There is, however, mounting evidence in favor of the distinction. Cognitive neuroscientists have found a double dissociation between declarative and procedural knowledge—some patients can acquire new declarative knowledge, but not procedural, whereas other patients can acquire procedural, but not declarative. There is also evidence that the two kinds of knowledge have different retrieval characteristics: declarative knowledge can be primed by any of its components, but procedural knowledge only works in one direction: from a specific set of cues to an action. A review of these issues can be found in (Anderson, 1993).

Rabinowitz and Goldberg (1995) conducted two experiments that nicely illustrate many of the recent phenomena concerning skill acquisition and the distinction between declarative and procedural knowledge. These experiments use a learning and transfer paradigm to examine learning of declarative and procedural knowledge, and their different retrieval characteristics.

This paper presents a single Act-R model that accounts for the data in the two Rabinowitz and Goldberg experiments. In addition, the paper presents protocol results from a newly conducted experiment designed to

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

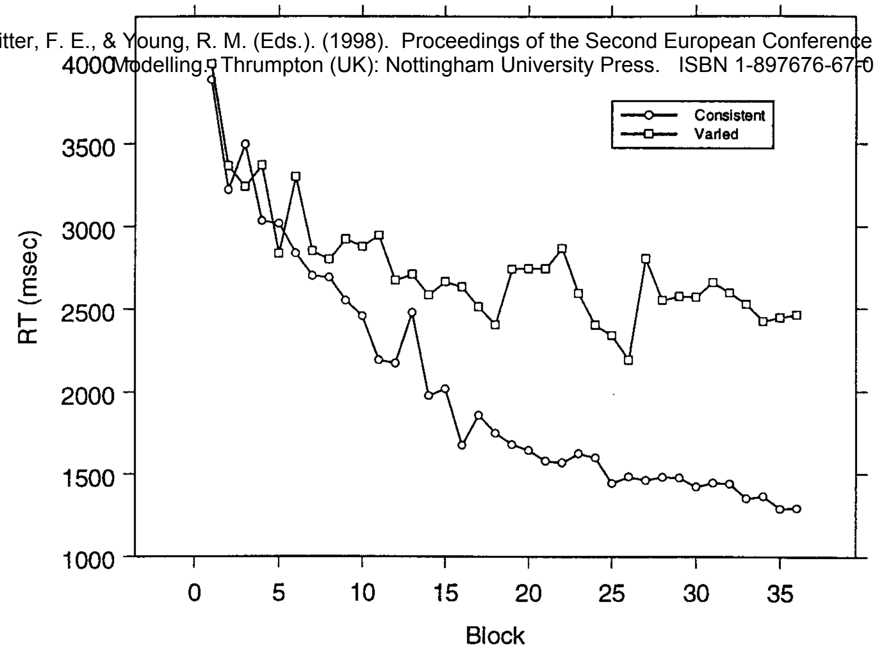


Figure 1: Mean response times during alphabet arithmetic training as a function of training group and practice block. Data plotted from original data by Rabinowitz and Goldberg (1995).

further test the assumptions of the experiments and the model.

THE RABINOWITZ AND GOLDBERG EXPERIMENTS

Both experiments used an alphabet arithmetic task, which consists of problems of the form *letter1* + *number* = *letter2*, where *letter2* is *number* letters after *letter1*. For example, A+2=C, because C is 2 letters after A.

In Experiment 1, one group of participants (the consistent group) received training on 36 blocks of problems, where each block consisted of the same 12 problems. Another group of participants (the varied group) received training on 6 blocks of problems, where each block consisted of the same 72 problems. Thus, both groups received 432 training trials, but the consistent group practiced each problem 36 times, whereas the varied group practiced each problem only 6 times. The problems used addends from 1 to 6. Consistent problems had two occurrences of each addend, whereas varied problems had 12 occurrences.

In the transfer phase, both groups received 12 new addition problems, repeated 3 times. Rabinowitz and Goldberg reasoned that during training the consistent group would quickly acquire declarative knowledge of the answers and switch to retrieval, whereas the varied group would continue to count up the alphabet. Thus the consistent group would get a lot of practice at retrieving the answers to the same 12 problems, but relatively little practice on the procedural knowledge needed to count up the alphabet. In contrast, the varied group would receive little or no practice retrieving declarative knowledge, but a great deal of practice counting up the alphabet. When transferred to the 12 new addition problems, the consistent group should revert to counting up the alphabet, resulting in a dramatic decrease in speed. However, the varied group should show perfect transfer from the training problems to the new problems.

The training results are shown in Figure 1. Each point on the graph is the mean of the median response times for all subjects on a block of 12 problems. The different asymptotes support the assertion that varied participants practice procedural knowledge, while consistent participants switch to and then practice retrieval.

The transfer results, shown in Figure 2, support the predictions: the varied group shows perfect transfer, but the consistent group shows considerable slow-down.

Although Experiment 1 supports the predictions, it is also consistent with a procedural-only long-term store. The consistent subjects might have acquired problem-specific procedural knowledge that directly produces the answer to each problem. For example, knowledge of the form "If problem is A+2, then type C." Since this knowledge is specific to the 12 training problems, it would not have helped the participants during the transfer phase. This issue is examined in Rabinowitz and Goldberg's second experiment.

The second experiment attempts to determine whether consistent training leads to specific procedural knowledge, or to declarative knowledge. It is based on the hypothesis that declarative and procedural knowledge have different retrieval characteristics. Declarative knowledge is thought to be subject to symmetric retrieval, meaning that any part of a declarative memory element can act as a cue for the retrieval of that element. Procedural knowledge is thought to be subject to symmetric access, meaning that a procedure operates in only one direction: from condition to action.

Training in Experiment 2 was identical to Experiment 1, however, in the transfer phase, both groups were given 12 subtraction problems repeated 3 times. A subtraction

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

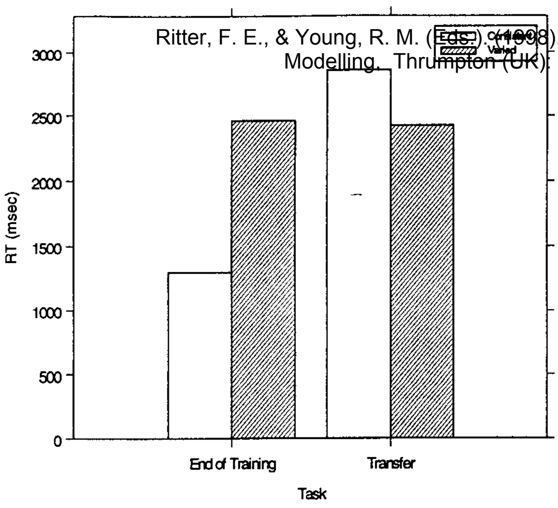


Figure 2: Mean response times for Experiment 1 as a function of task and group.

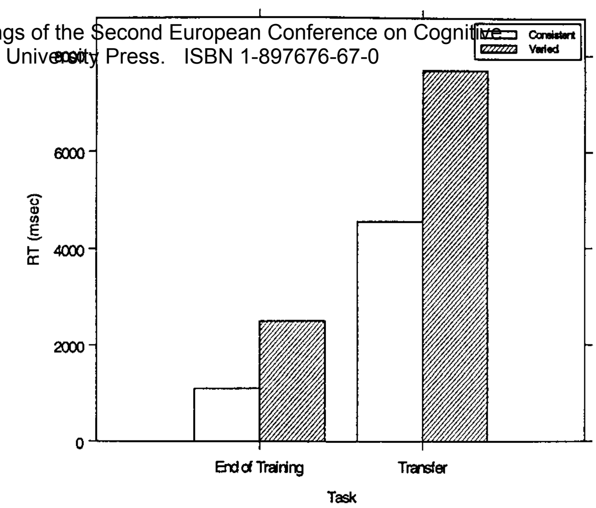


Figure 3: Mean response time for Experiment 2 as a function of task and group.

problem is of the form *letter1 - number = letter2*. For example, C-2=A. The 12 subtraction problems were inverted versions of the addition problems that both groups had seen during training. If the consistent group acquires declarative knowledge of the addition problems, the participants in this group should be able to solve the subtraction problems by retrieving and inverting addition problems. However, if this group has acquired problem-specific procedural knowledge, they will need to develop a new procedural for counting down the alphabet, as will the varied participants—who presumably strengthen their procedural knowledge during training.

Training results are similar to those for Experiment 1, so they are not reproduced here. Figure 3 shows that the transfer results are consistent with the predictions: the varied group requires considerably more time than the consistent group.

Taken together, Experiments 1 and 2 support the speed-up of both declarative knowledge retrieval and procedural knowledge application, as well as symmetric access to declarative knowledge and asymmetric access to procedural knowledge.

AN ACT-R MODEL

Act-R (Anderson, 1993) seems well suited for modeling these results, because it contains procedural and declarative long-term stores, along with learning mechanisms that alter the speed of elements in the two stores as a function of experience. Trafton (1996) has described an Act-R model for Experiment 1, but a bigger challenge is to construct a single Act-R model that can account for the results from both experiments. Such a model will serve three purposes. First, it will act as an additional test for several of Act-R's theoretical assumptions. Second, although each of Act-R's mechanisms has been tested in isolation, this model will test the interaction of several mechanisms. Third, the model will provide an explicit account of declarative and procedural learning and transfer that might then be used to analyze a wide range of more complex cognitive tasks.

The model presented here uses Act-R 4.0 (Anderson & Lebiere, in press).

Act-R is a parallel matching, serial firing rule-based system. It contains two long-term stores: procedural memory, represented by production rules, and declarative memory, represented by an associative network of declarative memory elements (DMEs). Working memory is viewed as the highly active portion of long-term declarative memory.

The alphabet arithmetic model has six production rules for the main goal. These are described in Table 1. READ-DISPLAY and ENCODE-DISPLAY simply read and look up the meaning of the textual symbols in the problem. REPORT-ANSWER reports the answer and signals that the goal has been achieved.

The remaining three rules—RETRIEVE-PLUS-RESULT, RETRIEVE-MINUS-RESULT, AND SUBGOAL-COUNT—are the most important rules in the model. RETRIEVE-PLUS-RESULT attempts to solve an addition problem by retrieving a fact from declarative memory that matches the problem, but also contains the answer. If successful, it uses the retrieved answer as the solution. RETRIEVE-MINUS-RESULT attempts to solve a subtraction problem by retrieving an addition DME that is the inverse of the subtraction problem. In other words, if the current problem is C-2=?, this rule will attempt to retrieve a fact of the form *letter* + 2 = C. SUBGOAL-COUNT creates a subgoal to solve the current problem by counting up or down the alphabet.

The model is designed so that Act-R will first try to retrieve an answer by using one of the retrieve rules. If the retrieval fails, then SUBGOAL-COUNT will fire to create the computation subgoal.

The model switches from computation to retrieval by acquiring declarative representations of problems that it has solved. When the model begins to solve problems it does not have any DMEs of past problems to retrieve, so it always uses SUBGOAL-COUNT. However, each time it solves a problem, it automatically remembers the

Ritter, F. E. & Young, R. M. (Eds.) (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

problem and solution as a DME. These DMEs are then available for recall in future trials. Details of this memorization process are given below following the description of the computation subgoal.

The computation subgoal works by counting either up or down the alphabet. It uses a set of declarative memory elements that represent the alphabet using chunks thought to be common to people raised in United States:

ABCD EFG HIJK LMNOP QRS TUV WXYZ

Each chunk is a DME containing up to five letters and a pointer to the next chunk. For example, the second chunk in the alphabet (named alpha2) is represented as:

```
alpha2
        ISA item
        FIRST e
        SECOND f
        THIRD g
        NEXT alpha3
```

The subgoal contains 26 rules that implement counting forward and backward through the alphabet. To do this, it must first retrieve the alphabet chunk that contains the starting letter. Next it steps forward along the chunk until it finds the starting letter. Finally, it counts along the alphabet (either forward or backward) the required number of letters. If it reaches a chunk boundary, it must retrieve either the next or previous chunk before continuing the count.

The subgoal automatically produces a declarative memory trace of the problem and its solution. Goals in Act-R are DMEs that have been pushed onto the goal stack. You can think of a goal as a kind of goal-specific working memory, because it encodes the problem, the solution, and any partial results. When the subgoal has computed an answer, a rule pops the goal off of Act-R's goal stack. This removes the goal from the stack, but it remains in declarative memory as a DME representing the problem and its solution. For example, the DME representing A+2=C is:

```
Add-fact-10
        ISA problem
        ARG1 a
        OP plus
        ARG2 2
        COUNT 2
        RESULT c
```

Here, Add-fact-10 is an arbitrary name for the DME, and COUNT is used during processing to keep track of how many letters were counted.

Every time the subgoal solves a new problem, it leads to a new DME representing the problem and its solution. These DMEs are then available for retrieval by the two retrieval rules described above.

The model accounts for the experimental data by using three of Act-R's mechanisms: base-level learning, which speeds up access to commonly retrieved DMEs, strength learning, which speeds up rules that are commonly used, and the memory retrieval threshold, which prevents the retrieval of DMEs below a specified activation.

To understand how these mechanisms produce the speed-up and transfer shown in the data, you must first understand how Act-R predicts latencies. The total time for a trial in Act-R is the sum of the times needed to fire each production rule during that trial. The time to fire a rule is the sum of the time needed to retrieve the DMEs it matches plus the time to execute the rule's action. The time to retrieve a DME depends on its activation and the strength of the production rule that is retrieving it. Intuitively, latency of retrieval is inversely proportional to production strength and DME activation. The time to match DME i is given by Equation 1:

$$t_i = Fe^{-f(A_i+S_p)} \qquad \text{Equation 1}$$

Here, F and f are constants. Ai is the activation of DME i, and Sp is the strength of production p.

The activation of a DME is the sum of its base level activation and the spreading activation from other DMEs:

$$A_i = B_i + \sum_j W_j S_{ji} \qquad \text{Equation 2}$$

where Bi is the base level activation, Wj is the source activation of DME j, and Sji is the strength of association from j to i. A single unit of source activation is divided among all DMEs that fill slots of the current goal. For the present model, this means that elements of the current problem (i.e., the letter, operator, and number) will spread activation to DMEs representing past solutions.

Read-Display

IF the goal is to do an alphabet arithmetic problem, but the problem text has not yet been read
THEN read the problem text from the display

Encode-Display

IF the goal is to do an alphabet arithmetic problem, and the problem text has been read, but its meaning has not been determined
THEN encode the meaning of each textual symbol

Retrieve-Plus-Result

IF the goal is to do an alphabet ADDITION arithmetic problem of the form letter1 + number =, but the answer has not been determined, and there is a fact in memory stating that letter1 + number = letter2
THEN note letter2 as the answer

Retrieve-Minus-Result

IF the goal is to do an alphabet SUBTRACTION arithmetic problem of the form letter1 - number =, but the answer has not been determined, and there is a fact in memory stating that letter2 + number = letter1
THEN note letter2 as the answer

Subgoal-Count

IF the goal is to do an alphabet arithmetic problem, but the answer has not been determined
THEN set a subgoal to compute the answer by counting

Report-Answer

IF the goal is to do an alphabet arithmetic problem, and the answer has been determined
THEN report the answer and pop the goal

Table 1: The English version of the model's main production rules

Ritter, F. E. & Young, R. M. (Eds.) (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0.

For example, if the current goal is to solve A+2, then A will spread activation to all traces of previous problems that contain A either as the first letter or as the answer. The same is true for the operator and the number. Hence, the DME that represents the past solution to the current problem will receive activation from all three elements and will, most likely, be the most active DME.

The base level activation of a DME reflects the log prior odds that the DME will be matched by a production rule. Act-R assumes that these odds increase as a function of use and decrease as a function of delay. This is given by the optimized base-level learning equation.

$$B_i = \ln\left(\frac{nL^{-d}}{1-d}\right) + \beta \qquad \text{Equation 3}$$

where β represents the initial base-level, d is the decay rate, L is the time since the DME was created, and n is the number of times the DME has been used. This equation assumes that the uses of the DME are evenly spaced in time. This is a reasonable assumption for the present model, because each trial occurs only once in a given block. Act-R's exact base-level learning equation does not make this assumption, but is much more expensive to compute.

A use count of a DME is incremented whenever the DME is retrieved by a rule or when a duplicate DME is created. As noted above, when a goal is popped from the stack it remains in declarative memory. However, if Act-R detects that a newly created DME is identical to an existing DME, then it destroys the new DME and increments the use count of the old DME. This is important during initial skill acquisition, because a newly created DME might be too inactive to recall after a brief delay. When this happens, the model must recompute the answer. Since the subgoal creates a duplicate DME, the original DME is strengthened, increasing the chances of recall in future trials.

A DME that matches a rule's condition will be successfully retrieved whenever its activation exceeds the global retrieval threshold. Act-R assumes that DME activation contains permanent noise with mean 0 and variance σ_1^2. When a DME is first created, its base-level activation is set to a base level constant plus the permanent activation noise.

We can now see how the model might learn to retrieve declarative traces in the consistent training condition, but not in the varied training condition. In the consistent condition, the model is exposed to each problem 36 times. These frequent exposures boost the base-level activation of the memory traces, allowing the retrieval rules to directly recall the solutions. In contrast, in the varied condition the model is exposed to each problem only six times. In addition, the varied condition takes longer because the first 72 trials can only be solved by counting. In the consistent condition there is a chance of recalling one or more answers after the first 12 trials.

The speed-up of participants in the consistent condition is predicted by Equation 1, which governs retrieval latency. It predicts that retrieval latency is inversely proportional to activation and rule strength. Without considering rule strength we can see that an increase in DME activation will lead to lower predicted retrieval times and hence lower trial times in the consistent condition.

The model predicts that speed-up in the varied condition and part of the speed up in the consistent condition is due to speed-up of procedural knowledge. As discussed earlier in this section, Act-R assumes that the latency of a rule application is inversely proportional to its strength and the activation of the DMEs that it matches (see the discussion surrounding Equations 1 and 2). Rule strength is governed by the same equation that governs base-level learning (Equation 3) except that L is the time since the rule was created, d is a separate strength decay constant, and n is the number of times the rule has been fired.

Strength learning, combined with the latency equations (Equations 1 and 2), predict the speed-up in the varied condition and why varied training produces perfect transfer to new addition problems, whereas consistent training shows no transfer. In the varied condition, the model receives a lot of practice counting up the alphabet. Thus, the rules for counting, which are not specific to a single problem, are strengthened throughout training, and this strengthening continues during the transfer phase. In contrast, when the model is given consistent training, it learns to retrieve the answers to the 12 problems, so it rarely uses the counting rules. Once the model reaches the transfer phase it must begin to use the counting rules again, but their strengths will be either at or below their initial values, producing the dramatic slowdown observed in the data.

The model also accounts for the subtraction transfer results. In the consistent condition, the model acquires and strengthens DMEs representing each problem and its solution. When transferred to subtraction, these DMEs have a high enough activation to be retrieved and inverted by RETRIEVE-MINUS-RESULT. The model predicts that performance will be slower than at the end of training, because it has not yet strengthened RETRIEVE-MINUS-RESULT. In contrast, when the model is in the varied training condition, the DMEs rarely become active enough to retrieve, so they are not available during transfer. Although the model has strengthened its rules for counting up the alphabet, very few of these rules are used to count down, so the model must use counting down rules that have not yet been used, and hence are much slower to fire.

Four parameters were estimated to fit the model to the data. These were the base-level learning decay parameter (d in Equation 3), production strength decay parameter, retrieval threshold, and permanent activation noise. Transient noise was not used. These four parameters are critical to fitting the data. The rule strength decay parameter affects the learning rate of procedural knowledge. The interaction of the retrieval threshold with the three other parameters determines the amount of practice needed before the model can switch from computation to retrieval. To fit the data, these parameters must be set so that consistent training leads the model to retrieve the answers, whereas varied training leads the model to continue to compute the answers. In addition, the parameters must also produce the right learning

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

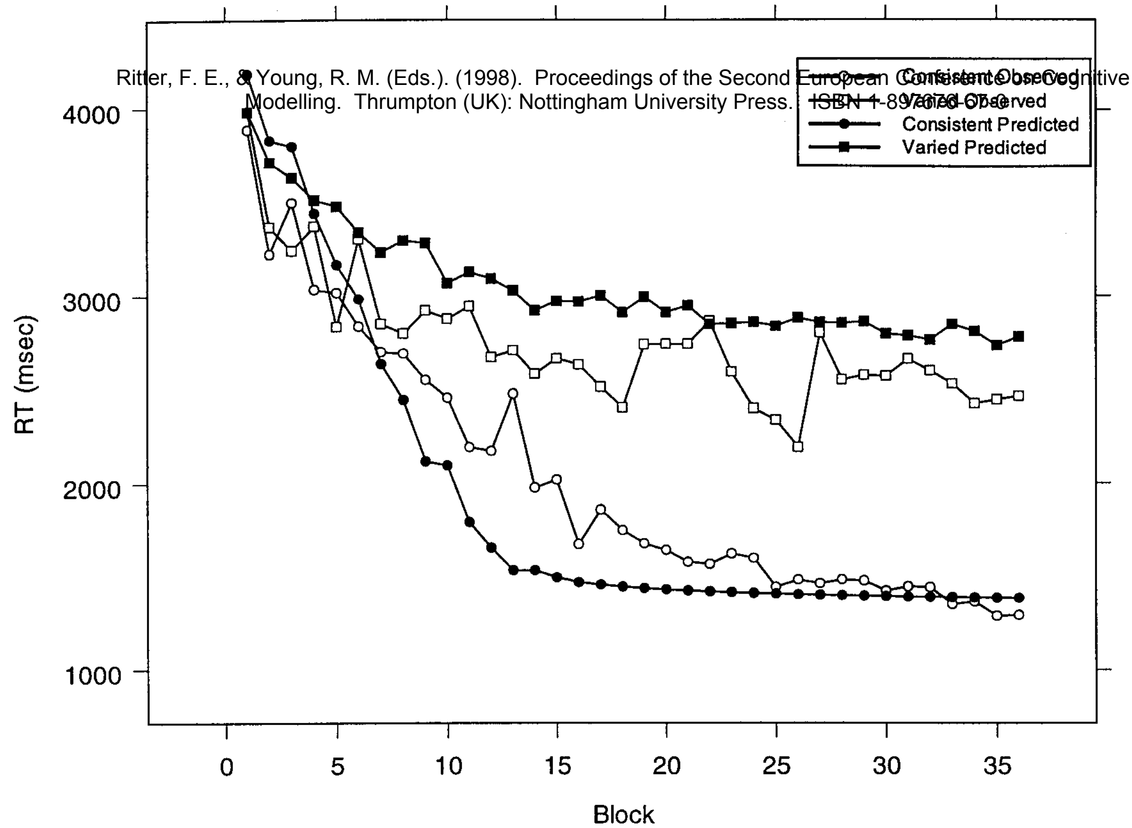


Figure 4: Observed and predicted mean response times during alphabet arithmetic training as a function of training group and practice block. Observed data replotted from Rabinowitz and Goldberg (1995).

curves for the two conditions.

The best fit was obtained with base-level learning decay set to .7, strength decay set to .5, retrieval threshold set to .55, and permanent activation noise variance set to .15. In addition, the total time to read the problem and type a letter was estimated at a constant 1.25 sec. This defines the lower bound of the model's response times. To reflect familiarity with the alphabet, all alphabet DMEs were given initial base-level activations of .974, reflecting 100 uses in the last 1000 seconds. Production rule strengths were initially set to .486, reflecting 25 uses in the past 1000 seconds. All other parameters used the default Act-R 4.0 values.

The model's predictions for the training phase in Experiment 1 are shown in Figure 4 along with the observed data. The model predictions were produced by simulating 15 subjects in each condition. The same model and parameter values were used for both conditions. The R^2 for the consistent condition was .89 and for the varied condition .78. This is pretty good considering that two different groups of subjects were modeled using the same parameters. In addition, the model captures the qualitative trends in the data—consistent simulations get much faster than varied simulations.

The transfer results are shown in Figures 5 and 6. The model closely fits the quantitative and qualitative results for alphabet addition transfer: consistent training leads to a large slow down in the transfer phase, whereas varied training results in perfect transfer. The subtraction transfer simulation matches the qualitative results, but not the quantitative ones: consistent training leads to better performance on subtraction than does varied training, but the model underestimates the latency of subtraction problems. Overall though, the fit is quite impressive, considering that four groups of subjects in four different conditions are fit using the same model and parameter values.

The modeling results raise several issues that will be addressed in the next section. The poor fit of the model to the quantitative subtraction data for the varied condition is easy to fix. It is possible to increase the time to compute a subtraction problem answer by either decreasing the strength of the subtraction counting rules or by switching to a different technique to solve the problems. A decrease in the rules' strengths is justifiable because most people rarely need to recite the alphabet backwards. However, it is also possible that people use a different strategy, such as guessing an answer and then counting forward to see if it is the right one.

The poor match to the subtraction latency in the consistent condition is much more puzzling. Specifically, why do the participants need over 4 seconds to solve each problem? If they are really recalling an alphabet addition problem and inverting it, then they should be closer to the predicted times, but instead their times are more than double the predictions. One possibility is that only a subset of varied participants actually switched to retrieval, whereas the remainder used computation.

The model's good fit to the data shows that active declarative knowledge is not needed to account for the results. Thus, the two experiments do not discriminate between declarative knowledge being inert or active.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

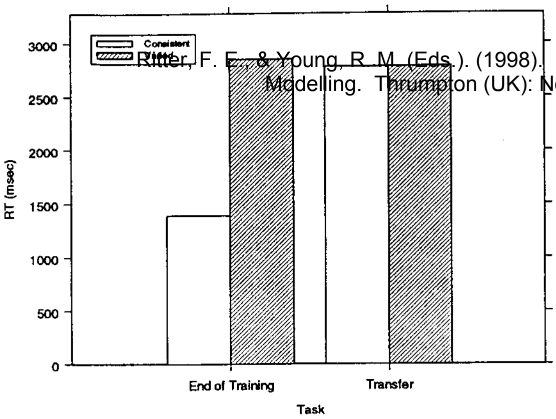


Figure 5: Mean predicted response times for Experiment 1 as a function of task and group.

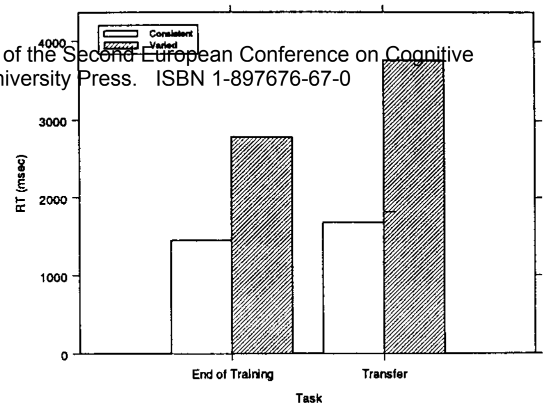


Figure 6: Mean predicted response times for Experiment 2 as a function of task and group.

However, it is possible that protocol data might provide evidence concerning this issue.

PROTOCOL ANALYSIS

To better understand the strategies that people use for alphabet arithmetic, particularly with respect to subtraction, a variant of Experiment 2 was run at The Ohio State University. Participants were 42 undergraduate students at The Ohio State University who received course credit for their effort. This experiment was similar to Rabinowitz and Goldberg's except that participants answered a questionnaire halfway through training and immediately after the transfer phase. Part 1 of the questionnaire contained the question: "Please describe all strategies that you used to solve the alphabet addition problems. If you used multiple strategies (or changed strategies), be as specific as possible about where and when you used them." Part 2 (completed at the end of the experiment) contained two questions: 1) "Please describe all strategies that you used to solve the alphabet **ADDITION** problems since the break. If you used multiple strategies (or changed strategies), be as specific as possible about where and when you used them." and 2) Please describe all strategies that you used to solve the alphabet **SUBTRACTION** problems. If you used multiple strategies (or changed strategies), be as specific as possible about where and when you used them."

Three main strategies were mentioned during the training phase: counting only, counting plus recall, and computing (in an unspecified way) plus recall. Many more strategies were mentioned in the transfer phase: counting backwards, recall plus inversion only, computing initially then switching to recall and inversion, and generate and test. Table 2 shows the results in terms of the percentage of participants in each category. For this analysis, responses to both training questions were coded together. The results clearly support the assumption that varied training leads to faster counting, whereas consistent training leads to direct retrieval. 95% of the participants in the consistent group reported using recall during training, versus only 32% of those in the varied condition. Most participants in the varied group (68%) reported that they used only counting throughout the entire training phase, in contrast to only 5% of participants in the consistent group.

The transfer protocol results are consistent with the hypothesis that varied training leads to strengthened asymmetrically accessible procedural knowledge for counting up, whereas consistent training leads to symmetrically accessible declarative knowledge. 70% of the consistent group reported recalling and inverting the addition problems, versus only 5% of the varied group. Likewise, only 15% of the consistent group reported counting back only, versus 36% of the varied group. Another 18% of the varied group used the generate and test strategy.

These results help clarify the model's problems of underestimating the difficulty of subtraction. First, they show that at least 15% of the consistent group used computation instead of recall, offering a possible explanation for the higher than predicted response times for this group on the transfer task. Second, the results indicate that the model's strategy of counting backward is consistent with the majority of participants in the varied group, but that the model is simply underestimating the time required to count back. In fact, two participants who used generate and test, mentioned that they switched to this method because counting back was too difficult. In contrast, counting back in the model within an alphabet chunk is just as fast as counting forward. The model's slower subtraction times are due only to the increased time needed to retrieve the previous chunk, thus subtraction problems that do not cross a chunk boundary are just as fast as addition problems. Resolving this problem should bring the model's predictions closer to the observed data.

The protocol data provides little evidence of whether declarative knowledge is inert or active. Only 10% of the consistent group mentioned computing the answers to a few subtraction problems before recognizing them as inverted addition problems.

CONCLUSION

This paper has three main results. The first is that the successful fit of the model to the alphabet arithmetic results shows that the two experiments fail to discriminate between active or inert declarative memory. Declarative memory in Act-R is inert—it can only be retrieved in the service of a production rule. Although the protocol data provided little insight into this issue, it does

Table 2: Reported strategy use based on training group and task.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

	Condition	
	Consistent (n = 20)	**Varied** (n = 22)
Training		
Counting only	5 % (1)	68% (15)
Count + Recall	80% (16)	32% (7)
Compute + Recall	15% (3)	0%
Transfer		
Counting back only	15% (3)	36% (8)
Recall and Invert	60% (12)	5% (1)
Count back then recall and invert	5% (1)	0%
Compute then Recall and Invert	5% (1)	0%
Generate and Test	5% (1)	18% (4)
Count back + Generate and Test	0%	9% (2)
Other	5% (1)	5% (1)
Not codable	5% (1)	27% (6)

suggest that some kind of recognition process is needed before a participant can switch to recall and inversion. Recent work on feeling-of-knowing (i.e., the feeling that you know an answer to a problem) provides some support for this claim. Schunn, et al. (1997) have shown that feeling-of-knowing is based on similarity of the problem to previously seen problems, not on the availability of an answer to the problem. Since subtraction problems are so different from the inverted addition problems, it seems likely that solving one or two subtraction problems might lead to a feeling of knowing based on similarity between the solved subtraction problem and previously seen addition problems. This feeling-of-knowing might then prompt a person to consciously explore the similarities.

Second, the model's successful fit to the data and the protocol results provide additional support for separate declarative and procedural long-term memory stores. In addition, the model also shows that the separate strengthening of procedural and declarative knowledge can produce the observed results.

Finally, the paper shows that Act-R is sufficient to capture both the qualitative and quantitative details of the acquisition and transfer of procedural and declarative memory. Even more importantly, the model shows that several Act-R mechanisms working together can predict whether training will lead to procedural strengthening or the recall of declarative knowledge.

ACKNOWLEDGMENTS

Special thanks to Mitchell Rabinowitz for graciously supplying the experimental data presented in this paper, and to Hongbin Wang for his help during data collection. This research was supported in part by grants N00014-95-1-0241 and N00014-96-1-0472 from the Office of Naval Research, Cognitive and Neural Science and Technology Division.

REFERENCES

Anderson, J. R. (1993). *Rules of the Mind*. Hillsdale, NJ: Lawrence Erlbaum.

Anderson, J. R., & Lebiere, C. (in press). *The Atomic Components of Thought.* Hillsdale, NJ: Lawrence Erlbaum.

Larkin, J. H. (1981). Enriching formal knowledge: A model for learning to solve textbook physics problems. In J. R. Anderson (Ed.), *Cognitive Skills and Their Acquisition* (pp. 311-334). Hillsdale, New Jersey: Lawrence Erlbaum Assoc.

Miller, C. S. (1993). *Modeling Concept Acquisition in the Context of a Unified Theory of Cognition.* Unpublished PhD, Univ. of Michigan.

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

Newell, A., & Rosenbloom, P. S. (1981). Mechanisms of skill acquisition and the law of practice. In J. R. Anderson (Ed.), *Cognitive Skills and Their Acquisition* (pp. 1-55). Hillsdale, New Jersey: Lawrence Erlbaum.

Rabinowitz, M., & Goldberg, N. (1995). Evaluating the structure-process hypothesis. In F. E. Weinert & W. Schneider (Eds.), *Memory Performance and Competencies: Issues in Growth and Development* (pp. 225-242). Hillsdale, NJ: Lawrence Erlbaum.

Rosenbloom, P. S., & Aasman, J. (1990). Knowledge Level and Inductive Uses of Chunking (EBL). Paper presented at the *Proceedings of the Eighth National Conference on Artificial Intelligence.*

Schunn, C. D., Reder, L. M., Nhouyvanisvong, A., Richards, D. R., & Stroffolino, P. J. (1997). To calculate or not to calculate: A Source activation confusion model of problem familiarity's role in strategy selection. *Journal of Experimental Psychology: Learning, Memory, and Cognition,* 23(1), 3-29.

Steier, D. M., Laird, J. E., Newell, A., Rosenbloom, P. S., Flynn, R. A., Golding, A., Polk, T. A., Shivers, O. G., Unruh, A., & Yost, G. R. (1987). Varieties of learning in Soar: 1987. In P. S. Rosenbloom, J. E. Laird, & A. Newell (Eds.), *The Soar Papers: Research on Integrated Intelligence* (Vol. 1, pp. 537-548). Cambridge, Mass: MIT Press.

Stillings, N. A., Weisler, S. E., Chase, C. H., Feinstein, M. H., Garfield, J. L., & Rissland, E. L. (1995). *Cognitive Science: An Introduction.* (Second ed.). Cambridge, MA: MIT Press.

Trafton, J. G. (1996). Alphabet arithmetic and Act-R: A reply to Rabinowitz and Goldberg, *Proceedings of the Eighteenth Annual Conference of the Cognitive Science Society* . Mahwah, NJ: Lawrence Erlbaum Associates.

VanLehn, K. (1989). Problem solving and cognitive skill acquisition. In M. I. Posner (Ed.), *Foundations of Cognitive Science* (pp. 527-579). Cambridge, MA: MIT Press.

VanLehn, K. (1996). Cognitive skill acquisition. *Annual Review of Psychology*, 47, 513-539.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Skill Learning Using A Bottom-Up Hybrid Model

Ron Sun, Edward Merrill, Todd Peterson
The University of Alabama
Tuscaloosa, AL 35487, USA
rsun@cs.ua.edu

Abstract

This paper presents a skill learning model CLARION. Different from existing models of mostly high-level skill learning that use a top-down approach (that is, turning declarative knowledge into procedural knowledge), we adopt a bottom-up approach toward low-level skill learning, where procedural knowledge develops first and declarative knowledge develops from it. CLARION which follows this approach is formed by integrating connectionist, reinforcement, and symbolic learning methods to perform on-line learning. We compare the model with human data in a minefield navigation task. A match between the model and hûman data is observed in several comparisons.

1 Introduction

Skills vary in complexity and the degree of cognitive involvement. They range from simple motor movements and other routine tasks in everyday activities to high-level intellectual skills. We want to study "lower-level" cognitive skills, which have not received sufficient research attention. One type of task that exemplifies what we call low-level cognitive skill is reactive sequential decision making (Sun and Peterson 1995). It involves an agent selecting and performing a sequence of actions to accomplish an objective on the basis of moment-to-moment information (hence the term "reactive"). An example of this kind of task is the minefield navigation task developed at The Naval Research Lab (see Gordon et al. 1994). This kind of task setting appears to tap into real-world skills associated with decision making under conditions of time pressure and limited information. Thus, the results we obtain from human experiments will likely be transferable to real-world skill learning situations. Yet this kind of task is suitable for computational modeling given the recent development of machine learning techniques (Sun et al 1996, Watkins 1989).

The distinction between procedural knowledge and declarative knowledge has been made in many theories of learning and cognition (for example, Anderson 1982, 1993, Keil 1989, Damasio et al. 1994, and Sun 1995). It is believed that both procedural and declarative knowledge are essential to cognitive agents in complex environments. Anderson (1982) originally proposed the distinction based on data from a variety of skill learning studies, ranging from arithmetic to geometric theorem proving, to account for changes resulting from extensive practice. Similar distinctions have been made by other researchers based on different sets of data, in the areas of skill learning, concept formation, and verbal informal reasoning (e.g., Fitts and Posner, 1967; Keil, 1989; Sun, 1995).

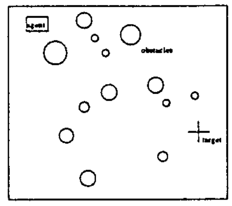


Figure 1: Navigating Through Mines

Most of the work in skill learning that makes the declarative/procedural distinction assumes a top-down approach; that is, learners first acquire a great deal of explicit declarative knowledge in a domain and then through practice, turn this knowledge into a procedural form ("proceduralization"), which leads to skilled performance. However, these models were not developed to account for skill learning in the absence of, or independent from, prexisting explicit domain knowledge. Several lines of research demonstrate that individuals can learn to perform complex skills without first obtaining a large amount of explicit declarative knowledge (e.g., Berry and Broadbent 1988, Stanley et al 1989, Lewicki et al 1992, Willingham et al 1992, Reber 1989, Karmiloff-Smith 1986, Schacter 1987, and Schraagen 1993). In research on *implicit learning*, Berry and Broadbent (1988), Willingham et al (1992), and Reber (1989) expressly demonstrate a *dissociation* between explicit knowledge and skilled performance in a variety of tasks including dynamic decision tasks (Berry and Broadbent 1988), artificial grammar learning tasks (Reber 1989), and serial reaction tasks (Willingham et al 1992). Berry and Broadbent (1988) argue that the psychological data in dynamic decision tasks are not consistent with exclusively top-down learning

Ritter, F.E. & Young, R.M. (Eds.) (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

models, because subjects can learn to perform the task without being provided a priori declarative knowledge and without being able to verbalize the rules they used to perform the task. This indicates that procedural skills are not necessarily accompanied by explicit declarative knowledge, which would not be the case if top-down learning is the only way to acquire skill. Willingham et al (1989) similarly demonstrate that procedural knowledge is not *always* preceded by declarative knowledge in human learning, and show that declarative and procedural learning are not necessarily correlated. There are even indications that explicit knowledge may arise from procedural skills in some circumstances (see Stanley et al 1989). Using a dynamic decision task, Stanley et al. (1989) found that the development of declarative knowledge paralleled but lagged behind the development of procedural knowledge.

Similar claims concerning the development of procedural knowledge prior to the development of declarative knowledge have surfaced in a number of research areas outside the skill learning literature and provided additional support for the bottom-up approach. *Implicit memory* research (e.g., Schacter 1987) demonstrates a dissociation between explicit and implicit knowledge/memories in that an individual's performance can improve by virtue of implicit "retrieval" from memory and the individual can be unaware of the process. This is not amenable to the exclusively top-down approach. *Instrumental conditioning* also reflects a learning process that differs from the top-down approach, because the process is typically non-verbal and involves the formation of action sequences without requiring a priori explicit knowledge. It may be applied to simple organisms as well as humans (Gluck and Bower 1988). In *developmental psychology*, Karmiloff-Smith (1986) proposed the idea of "representational redescription". During development, low-level implicit representations are transformed into more abstract and explicit representations and thereby made more accessible. This process is not top-down either, but in the opposite direction.

2 The Model

The difference between declarative and procedural knowledge leads naturally to "two-level" architectures (Sun 1995). We thus developed the model CLARION, which stands for *Connectionist Learning with Adaptive Rule Induction ON-line* (Sun et al 1996). It embodies the distinction of declarative and procedural knowledge (or, conceptual and subconceptual knowledge), and it performs learning in a bottom-up direction. It consists of two main components: the top level encodes explicit declarative knowledge in the form of propositional rules, and the bottom level encodes implicit procedural knowledge in neural networks. In addition, there is an episodic memory, which stores recent experiences in the form of "input, output, result" (i.e., stimulus, response, and consequence).

A high-level pseudo-code algorithm that describes CLARION is as follows:

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

In the bottom level, 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. We can choose an action based on Q-values. To acquire the Q-values, supervised and/or reinforcement learning methods may be applied. A widely applicable option is the *Q-learning* algorithm (Watkins 1989), a reinforcement learning algorithm. In the algorithm, $Q(x, a)$ estimates the maximum discounted cumulative reinforcement that the agent will receive from the current state x on. The updating of $Q(x, a)$ is based on minimizing $r + \gamma e(y) - Q(x, a)$, where γ is a discount factor and $e(y) = \max_a Q(y, a)$. Thus, the updating is based on the *temporal difference* in evaluating the current state and the action chosen: In the above formula, $Q(x, a)$ estimates, before action a is performed, the (discounted) cumulative reinforcement to be received if action a is performed, and $r + \gamma e(y)$ estimates the (discounted) cumulative reinforcement that the agent will receive, after action a is performed; so their difference (the temporal difference in evaluating an action) enables the learning of Q-values that approximate the (discounted) cumulative reinforcement. Using Q-learning allows sequential behavior to emerge in an agent. Through successive updates of the Q function, the agent can learn to take into account future steps in longer and longer sequences.

To implement Q functions, we chose to use a four-layered network (see Figure 2), in which the first three layers form a (either recurrent or feedforward) backpropagation network for computing Q-values and the fourth layer (with only one node) performs stochastic decision making. The output of the third layer (i.e., the output layer of the backpropagation network) indicates the Q-value of each action (represented by an individual node), and the node in the fourth layer determines probabilistically the action to be performed based on a Boltzmann distribution (i.e., Luce's choice axiom; Watkins 1989). This learning process performs both structural credit assignment (with backpropaga-

Ritter, F. E., & Young, R. M. (Eds.) (1998) Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

tion), so that the agent knows which element in a state should be assigned credit/blame, as well as temporal credit assignment, so that the agent knows which action leads to success or failure. This learning process enables the development of procedural skills potentially solely based on the agent independently exploring a particular world on a continuous and on-going basis.

In the top level, declarative knowledge is captured in a simple propositional rule form. To facilitate correspondence with the bottom level and to encourage uniformity and integration (Clark and Karmiloff-Smith 1993), we chose to use a localist connectionist model for implementing these rules (e.g., Sun 1992, Towell and Shavlik 1993). Basically, we translate the structure of a set of rules into that of a network. For each rule, a set of links are established, each of which connects a node representing a concept in the condition of a rule to the node representing the conclusion of the rule. For more complex rule forms including predicate rules and variable binding, see Sun (1992).

To fully capture bottom-up learning processes, we devised an algorithm for learning declarative knowledge (rules) using information in the bottom level (the *Rule-Extraction-Refinement* algorithm). The basic idea is as follows: if an action decided by the bottom level is successful then the agent extracts a rule (with its action corresponding to that selected by the bottom level and with its conditions corresponding to the current sensory state), and adds the rule to the top-level rule network. Then, in subsequent interactions with the world, the agent refines the extracted rule by considering the outcome of applying the rule: if the outcome is successful, the agent may try to generalize the conditions of the rule to make it more universal; if the outcome is not successful, then the conditions of the rule should be made more specific and exclusive of the current case.

We perform rule extraction at each step, based on the following information: (x, y, r, a), where x is the state before action a is performed, y is the new state entered after an action a is performed, and r is the reinforcement received after action a. Rules are in the following form: *conditions* $\longrightarrow$ *action*, where the left-hand side is a conjunction of individual conditions each of which refers to the value of an element in the (sensory) input state. Three different criteria can be used for rule learning at each step: (1) direct reinforcement received at a step, (2) temporal difference (as used in updating Q-values), and (3) maximum Q-values in a state. We adopt a three-phase approach, with each phase lasting for a certain number of episodes. Phase transition can be automatically determined based on the current performance level of the model. At each step, we apply the current-phase criterion to determine whether we should construct a rule. If so, a rule is wired up in the rule network. After rules are extracted, at each step, the algorithm reexamines the rules matching the current step to decide if each of them should be kept, revised, or discarded. See Sun et al. 1996 for the full details of rule learning.

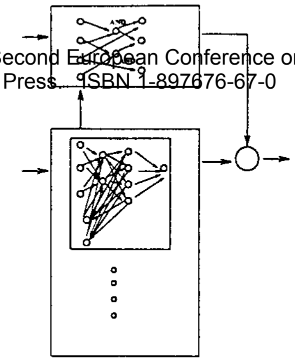


Figure 2: The implementation of CLARION.

Step 4 is for making the final decision on which action to take by incorporating outcomes from both levels. We combine the corresponding values for an action from the two levels by a weighted sum; that is, if the top level indicates that action a has an activation value v (which should be 0 or 1 as rules are binary) and the bottom level indicates that a has an activation value q (the Q-value), then the final outcome is $w_1 * v + w_2 * q$. Stochastic decision making with Boltzmann distribution (based on the weighted sums) is then performed. Figure 2 shows the two levels of the model.

3 Experiments

In all of the human experiments, subjects were seated in front of a computer monitor that displayed an instrument panel containing several gauges that provided current information (see Figure 3). The following instruction was given to explain the setting:

> I. Imagine yourself navigating an underwater submarine that has to go through a minefield to reach a target location. The readings from the following instruments are available:
>
> (1) Sonar gauges show you how close the mines are to the submarine. This information is presented in 8 equal areas that range from 45 degrees to your left, to directly in front of you and then to 45 degrees to your right. Mines are detected by the sonars and the sonar readings in each of these directions are shown as circles in these boxes. A circle becomes larger as you approach mines in that direction.
>
> (2) A fuel gauge shows you how much time you have left before you run out fuels. Obviously, you must reach the target before you run out of fuel to successfully complete the task.
>
> (3) A bearing gauge shows you the direction of the target from your present direction; that is, the angle from your current direction of motion to the direction of the target.
>
> (4) A range gauge shows you how far your current location is from the target.
>
> II. At the beginning of each episode you are located on one side of the minefield and the target is on the other side of the minefield. You task is to navigate through the minefield to get to the target before you run out of fuel. An episode ends when: (a) you get to the goal (success); (b)

you hit a mine (failure); (c) you run out of fuel (failure).

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

A random mine layout was generated for each episode. This setting is *stochastic* and *non-Markovian.* Five training conditions were used:

- The standard training condition. Subjects received five blocks of 20 episodes on each of five consecutive days (100 episodes per day). In each episode the minefield contained 60 mines. The subjects were allowed 200 steps.
- The verbalization training condition. This condition was identical to the standard training condition except that subjects were asked to step through slow replays of selected episodes and to verbalize what they were thinking during the episode. Subjects received replays on the first, third, and fifth days of training. The subjects were replayed five episodes after the first block of 20 episodes and five episodes after the fifth block of 20 episodes on these days.
- The over-verbalization training condition. In this condition subjects were presented replays of 15 of their first 25 episodes, and asked to verbalize during the slow playback. Replay of an episode occurred immediately after the subject finished the episode.
- The 30-to-60 transfer condition. This condition was also identical to the standard training condition except that subjects performed the task with 30 mines on the first two days of training and switched to 60 mines starting the third day.
- The mixed training condition. "Mixed" refers to the fact that mine density was manipulated during training. Subjects performed the task with 30, 50, 70, or 90 mines. Subjects received eight blocks of 10 episodes per day over five days, two at each mine density. Order of presentation was randomized.

In CLARION each gauge was represented by a set of nodes that corresponded to what human subjects would see on screen. This input setup yielded a total of 43 primary perceptual inputs. Thus, there were more than 10^{12} possible input states. Thus the model had to deal with the problem of high dimensionality. As a result, a lookup table implementation for Q-learning at the bottom level was not possible (Tesauro 1992, Lin 1992). To deal with the situation. a functional approximator such as backpropagation networks must be used. Also in correspondence to the human experimental setting, the action outputs consisted of two clusters of nodes representing turn and speed.

The model started out with no more a priori knowledge about the task than a typical human subject, so that bottom-up learning can be captured. The bottom level contained randomly initialized weights (with a pre-chosen, fixed topology). The top level started empty and contained no a priori knowledge about the task, either in the form of instructions or instances. The episodic memory was empty at the beginning. There was no supervised learning (i.e., no teacher input). The reinforcement signals embodied some a priori notions regarding getting close to target and avoiding explosion that were also provided to human subjects through instructions. The learning algorithm with all the requisite parameters was pre-set, presumably reflecting the learning mechanisms in humans.

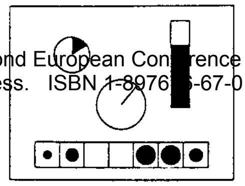


Figure 3: The Navigation Input

The display at the upper left corner is the fuel gauge; the vertical one at the upper right corner is the range gauge; the round one in the middle is the bearing gauge; the 7 sonar gauges are at the bottom.

The results of the experiments are analyzed as follows.

The standard training condition. We obtained performance data over 500 episodes per subject. We averaged the data over 10 human subjects. We did the same with the model: Each model run was initialized with different random number sequences and thus produced different results; we averaged 10 such runs in exact correspondence with human experiments (i.e., we did not tune the random number sequences to generate a match, but randomly set seeds for random number generators, analogous to random selection of human subjects in this experiment). We compared *average* success rates because in this way we can eliminate the uninteresting impact of individual differences and instead focus on essential features of learning in this task. These data are presented in Figure 4. Both sets of data were best fit by power functions (for failure rate). The degree of similarity is evident. A Pearson product moment correlation coefficient was calculated (treating blocks as individuals and human versus model as the X and Y variables). The analysis yielded a high positive correlation (r = .82), indicating a high degree of similarity between human subjects and model runs.

The verbalization training condition. Obviously, we could not require verbalization from the model. However, we posited that much of the effect of verbalization on learning was associated with rehearsing previous steps and episodes (although there may be additional factors involved). Thus for the model, we used episode memory playback (Lin 1992) in a first attempt to capture this effect. Episode memory playback involves training the model with previously performed episodes between blocks of actual trial episodes in exactly the same manner as in human experiments. In this case, the data from 5 human subjects was com-

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

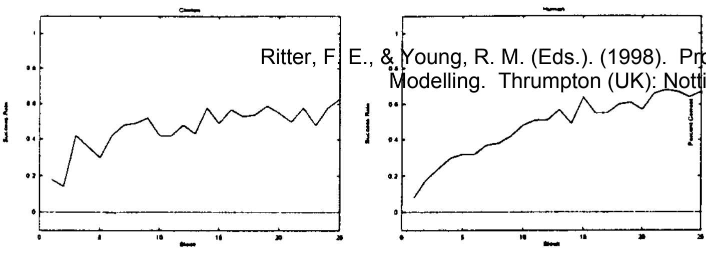


Figure 4: The learning curves in terms of success rates in the standard condition. The right side is the human data and the left side is the model data.

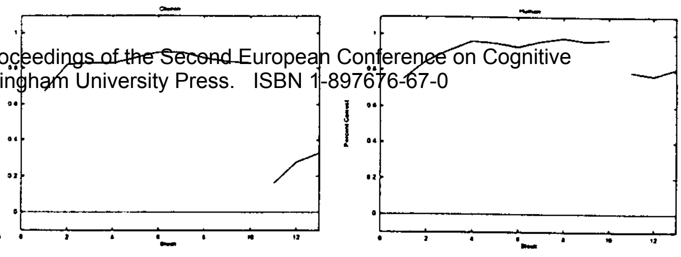


Figure 6: The 30-to-60 transfer data in terms of success rates.

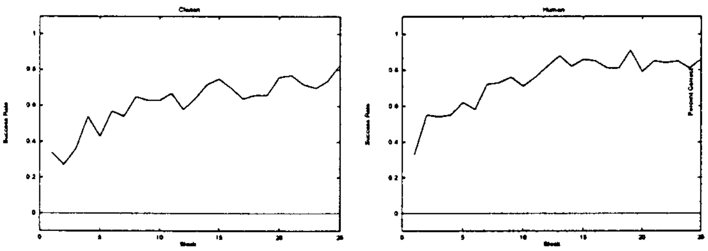


Figure 5: The learning curves in terms of success rates in the verbalization condition.

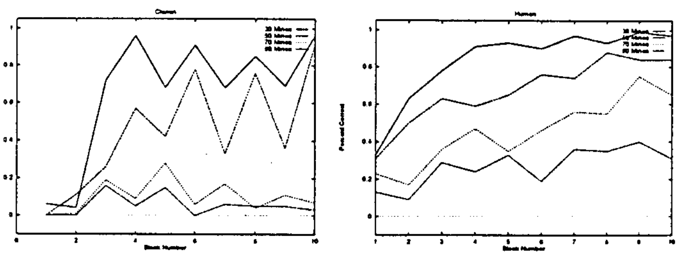


Figure 7: Average success rates for each mine densities in the mixed condition.

pared to that of 5 model runs. Data was averaged for each of 25 blocks (see Figure 5). Again, both sets of data were highly similar and both were best fit by power functions. We also calculated a Pearson product moment correlation coefficient, which yielded a high positive correlation (r = .84).

We subsequently compared the changes in performance due to verbalization for the human subjects and the model runs. This was done by averaging failure rates across blocks separately for each human subject and for each model run and subjecting that data to a 2 × 2 ANOVA. The analysis of these data indicated the both groups exhibited a significant increase in performance due to verbalization ($p < .01$), and that the changes due to verbalization for the two groups were not significantly different (52 to 25 percent failure rate for the human subjects versus 53 to 38 percent failure rate for the model runs). The effect of explication of implicit knowledge which likely results from verbalization was captured through the usual rule learning process, which was also at work during episode replay.

The 30-to-60 transfer condition. Subjects were first trained on 30-mine minefields, and then transferred to 60-mine minefields. The model was tested under the same condition. Both human and model data were averaged over 10 subjects. Comparing the human and model data (see Figure 6), we noticed that both learned well at 30 mines, although human data was slightly better. When transferred to 60 mines, both exhibited a significant drop in performance, although the model exhibited a deeper drop. Specifically, we compared performance of the last block before the change in mine density and the first block after the change. Success rates were 98% and 79% for the human subjects and 83% and 26% for the model runs respectively. The drops were both statistically significant. At first look, it might appear that the drop in performance for the model runs was much greater than that for the human subjects. However, this might not be a fair assessment in that we did not allow the model runs to reach the same performance as the human subjects before changing the mine density. Indeed, the 5 highest performing of the model runs before the change performed 8 times better after the change than did the 5 lowest performing ones.

The mixed training condition. We plotted learning curves in terms of success rates for each mine density separately. The data were averaged over 8 human subjects and 8 model runs, respectively. The average curves are shown in Figure 7. We calculated overall success rates for each of the mine densities. Both the human subjects and model runs performed best with the lowest mine density and performance decreased with each increase in the number of mines. Thus, we observed a similar pattern. The drop in performance was roughly the same for human subjects and model runs between the 30 and 50 mine densities (16% versus 13%, respectively). We do not know for sure what accounts for the failure of the model at the 70 and 90 mine densities. However, questionnaires completed by the human subjects indicated that they treated the higher density conditions as different from the lower density conditions. Because the model runs did not "start over" at each density, they were applying what was learned to conditions in which it did not work. In contrast, human subjects could sense the change in conditions and discard their old strategies.

The over-verbalization condition. Human subjects under the over-verbalization condition failed to learned. During the 25 episodes of training, their success rates were well below 10%, compared with the

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

33% performance for the subjects under the (sparse) verbalization condition. If we eliminate one subject who performed at 60%, the remaining subjects only achieved approximately 3% success rate. CLARION accounts for this phenomenon by positing that too much verbalization (e.g., verbalizing for more than half of the training episodes) caused the learner to switch to a completely explicit mode of learning; they tended to rely completely on the top-level learning mechanism and shut down the bottom level. This is consistent with the similar hypothesis by Stanley et al (1989), for explaining their findings regarding the difficulty their subjects had in learning a dynamic decision task after being given instructions that encouraged them to be explicit. Schooler et al (1993) also reported that requiring verbalization impaired subjects' ability to solve problems that require "insight", by forcing them to be overly explicit. CLARION explains the findings readily with the shut-down mechanism. The top-level learning mechanism when disconnected from the bottom level, clearly has trouble learning this kind of sequential task, because of its lack of a temporal credit assignment process (comparable in power to Q-learning) and its all-or-nothing learning process. On the other hand, in the bottom level, the distributed network representation and learning process that incorporates gradedness and temporal information handle complex sequences well.

Verbalization segments indicating bottom-up learning. The verbalization data we collected from the subjects (under the verbalization training condition) were consistent, in an informal sense, with our assumption of bottom-up learning being prominent in this task setting, as exemplified by the following segments.

> S: I thought about it after I started doing it. I said, look at me .... look what I'm doing. I didn't start thinking about it until I started doing it. I figured out that it started helping me and that's when I started doing it myself. (subj.38)

> S: When I started off ...... I didn't understand at all .... I couldn't grasp the whole sonar concept at all. (subj.38)

> S: So, basically what I do – not thinking about driving a submarine or mine. (subj.38)

> S: When you get in a situation like this, where there are gaps, it's purely instinctual. (subj.37)

> S: That's pretty much I've done the whole game [being instinctual], with the exception of a couple of patterns I've started to recognize. (subj.37)

In sum, the verbalization by the subjects suggested that some degree of bottom-level (implicit) learning/decision making and gradual bottom-up learning existed. This is the kind of learning CLARION was meant to capture.

We also compared the verbalizations of good performers (subjects) vs. poor performers. Our analysis indicated a lack of difference: we failed to notice any significant difference across a variety of measures (such as length of verbalization, detailedness, and types of statements uttered). We suggest that this is one more piece of evidence that indicates the importance/prominence of bottom-level (implicit) learning: The performance is mostly determined by implicit procedural learning, which cannot be easily verbalized, while verbalized explicit knowledge is nonspecific and has relatively minor impact during learning.

4 Conclusions

In sum, we discussed a hybrid connectionist model CLARION as a demonstration of the approach of bottom-up skill learning, which consists of two levels for capturing both procedural and declarative knowledge and performing bottom-up learning. Some degree of match with human data was found across a number of different experimental conditions.

Acknowledgements

This work is supported in part by Office of Naval Research grant N00014-95-1-0440. The simulator was provided by NRL.

References

J. R. Anderson, (1982). Acquisition of cognitive skill. *Psychological Review.* Vol.89, pp.369-406.

D. Berry and D. Broadbent, (1988). Interactive tasks and the implicit-explicit distinction. *British Journal of Psychology.* 79, 251-272.

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. Damasio, (1994). *Decartes' Error.* Grosset/Putnam, NY.

D. Gordon, et al. (1994). *User's Guide to the Navigation and Collision Avoidance Task.* Naval Research Lab. DC.

H. Dreyfus and S. Dreyfus, (1987). *Mind Over Machine: The Power of Human Intuition*, The Free Press, New York, NY.

P. Fitts and M. Posner, (1967). *Human Performance.* Brooks/Cole, Monterey, CA.

M. Gluck and G. Bower, (1988). From conditioning to category learning. *Journal of Experimental Psychology: General.* 117 (3), 227-247.

W. James, (1890). *The Principles of Psychology.* Dover, NY.

A. Karmiloff-Smith, (1986). From meta-processes to conscious access. *Cognition.* 23. 95-147.

F. Keil, (1989). *Concepts, Kinds, and Cognitive Development.* MIT Press. Cambridge, MA.

P. Lewicki, et al. (1992). Nonconscious acquisition of information. *American Psychologist.* 47, 796-801.

L. Lin, (1992). Self-improving reactive agents based on reinforcement learning, planning, and teaching. *Machine Learning.* Vol.8, pp.293-321.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Nottingham, UK: Nottingham University Press. ISBN 1-897676-67-0

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

J. Schooler, S. Ohlsson, and K. Brooks, (1993). Thoughts beyond words: when language overshadows insight. *Journal of Experimental Psychology: General.* 122 (2). 166-183.

D. Schacter, (1987). Implicit memory: History and current status. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 13, 501-518.

R. Shiffrin and W. Schneider, (1977). Controlled and automatic human information processing II. *Psychological Review.* 84. 127-190.

P. Smolensky, (1988). On the proper treatment of connectionism. *Behavioral and Brain Sciences*, 11(1):1–74.

W. Stanley, et al (1989). Insight without awareness *Quarterly Journal of Experimental Psychology.* 41A (3), 553-577.

R. Sun, (1992). On Variable Binding in Connectionist Networks, *Connection Science*, Vol.4, No.2, pp.93-124.

R. Sun, (1995). Robust reasoning: integrating rule-based and similarity-based reasoning. *Artificial Intelligence.* 75, 2. 241-296.

R. Sun, T. Peterson, and E. Merrill, (1996). Bottom-up skill learning. Technical Report TR-CS-96-0021, University of Alabama. shortened version in *Proc.of 18th Cognitive Science Society Conference*,

T. Tesauro, (1992). Practical issues in temporal difference learning. *Machine Learning.* Vol.8, 257-277.

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

D. Willingham, et al (1989). On the development of procedural knowledge. *Journal of Experimental Psychology: Learning, Memory, and Cognition.* 15, 1047-1060.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Modelling Memory-Updating Characteristics of 3- and 4-Year Olds

C. Philip Beaman
John Morton
MRC Cognitive Development Unit
4 Taviton Street
London WC1H 0BT, UK
+44 171 387 4692
P.Beaman@cdu.ucl.ac.uk
J.Morton@cdu.ucl.ac.uk

ABSTRACT

In this paper a memory perspective on young children's performance at a particular false belief task, the Smarties task, is described. The theoretical analysis focuses on the computational conditions that are required to resolve the Smarties task, on the possible limitation in the developing memory system that may lead to a computational breakdown resulting in a failure to resolve, and on ways of bypassing such limitations to ensure correct resolution. A symbolic model of this analysis implemented using the COGENT modelling environment is described, and its fit to the data considered.

Keywords

Developmental modelling, false belief, memory updating, COGENT

INTRODUCTION

One of the many constraints identified by Newell (1990) on any form of cognitive architecture which attempts to model human cognition is that it should be capable of arising from earlier forms by a process of developmental maturation. Developmental constraints, and discrete developmental stages, have received surprisingly little attention from symbolic modellers, although questions of how a mature system might develop from a relatively simple template are now being considered within the connectionist research program (e.g., Elman et al., 1996). The present study considers a developmental stage believed to be crucial to the maturation of memory processes, and aims to demonstrate how the failure of 3- and 4-year olds at a task which adults find trivially easy (the Smarties task; Perner, Leekam & Wimmer, 1987) can be modelled using a destructive-updating process. A subtle alteration of the memory encoding characteristics of this task enables 3- and 4-year olds to perform the task correctly. The patterns of children's performances are modelled as discrete developmental stages using the COGENT (Cognitive Objects in a Graphical EnvironmeNT) modelling environment of Cooper and Fox (in press).

The Smarties Task

The basic procedure for the Smarties task is as follows. The subjects are shown a tube of Smarties (a popular brand of sweet) and asked what the tube contains. Children of around the age of four are usually both able and willing to provide an answer to this question. The top is then taken off the tube, and its contents are shown to the child. The contents of the tube are pencils rather than the anticipated Smarties. The top is then replaced on the tube, and the child is asked two questions, the reality question (what is in the tube?) and the belief question (when you first saw the tube, what did you think was in the tube?). Typically, 70% of 3-year-old children who are able to answer the first question correctly (pencils) now also give the same answer to the second question.

A Memory-Updating Explanation

The original form of the Smarties task implies some peculiar memory characteristics. Children who fail this task are incorrectly reporting a belief which they had held, and told to the experimenter, only seconds previously. Although a conceptual deficit, an inability to comprehend false belief, can be put forward to explain these results, it seems strange to suppose that this deficit manifests itself in the child's inability to correctly recall the contents of this belief, even though they were able to report to the experimenter what the contents of this belief were immediately before it was shown to be false. Instead, it is argued (Barreau, 1997; Morton, 1997) that the child's inability is centred around a memory updating system, such that the false belief (that the tube contains Smarties) is never encoded as a stable, long-term representation, and so is immediately supplanted by the incoming information that the tube contains pencils. Thus, when such children are asked the belief question, the only source of information available to them is the representation of the current state of reality: *in(tube, pencils)*.

The Bag Experiment.

A variation on this experimental procedure designed to maximise the possibility that the contents of the tube are translated into a long-term format is described by Barreau (1997). Immediately after showing the tube to the child, and asking the child what they believed the tube to contain, the contents of the tube were emptied into a bag. Although the child witnessed this operation, at no time were they able to see the contents of the tube either at first or during the transfer. The tube was then shown to the child to demonstrate that it was empty, and then ostentatiously

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

hidden from view. The child is then asked what they believe to be in the bag. All children replied "Smarties". The contents of the bag were then shown to the child. In this case, the bag contained marbles, rather than Smarties. The child was then asked five questions concerning the contents of the bag and the tube:

1. Before I opened the bag, what did you think was in the bag? (BAG:BELIEF: PAST)
2. What is really in the bag? (BAG:REALITY: PRESENT)
3. When I first showed you the tube, what did you think was in the tube? (TUBE: BELIEF: PAST)
4. What is inside the tube now? (TUBE: REALITY: PRESENT)
5. What was really inside the tube? (TUBE: REALITY: PAST)

In Barreau's (1997) experiment, twenty-four children were questioned in this manner, the results of this experiment are shown in the table below:

TABLE 1: Table of answers to the tube and bag questions.

Questions	Correct	Reversed	Double
BAG	8	8	8
TUBE	15	3	6

In order to be scored correct, both the bag questions, (belief and reality) had to be correctly answered. To be scored correct in the tube condition, the belief questions and at least one of the reality questions had to be correctly answered. A "double" score refers to a repeat answer, i.e. a reality response to a belief question. This category also includes one child who gave belief answers to reality questions. The reversed response indicates a reversal between the belief and reality answers in the bag questions, and the belief and one of the reality answers in the tube questions.

The assumptions underlying this experiment were that when the tube was removed from view, the tube→ bag transferral episode would be coded as ended, and details of the whole episode would be translated into long-term memory. Thus, when the current representation of the bag's contents is updated, the representation of the tube's contents will be invulnerable.

The data has also been analysed as suggesting that three qualitatively different developmental processes are occurring amongst the children tested (Barreau, 1997).The children were divided into three groups on the basis of the scores they were given for the bag questions. Of the 8 children who were scored as correct for the bag questions, 7 were also correct for the tube question, and 1 gave a "double" response. Of the 8 children who gave reversed responses for the bag question, 6 were scored as correct on the tube question, there was 1 reversed response, and 1 double response, and for the 8 children who scored "double" responses for the bag questions, 2 were correct on the tube questions, 2 gave reversed responses, and 4 gave double responses. This pattern of data was considered to be a little too complex to be easily handled by a traditional verbal theory.

A COGENT IMPLEMENTATION

To properly test the theory against the data, a family of models were produced using the COGENT modelling environment. The basic architecture used in this approach is reproduced below: In this figure, hexagons represent processes, rounded rectangles represent buffers, and diamonds represent data boxes. Square boxes represent compounds, which may contain buffers and processes. Arrows with standard heads indicate message sending. Arrows with black triangular tails indicate buffer reading. Compound arrows (which are denoted by triangular and standard heads) allow both functions.

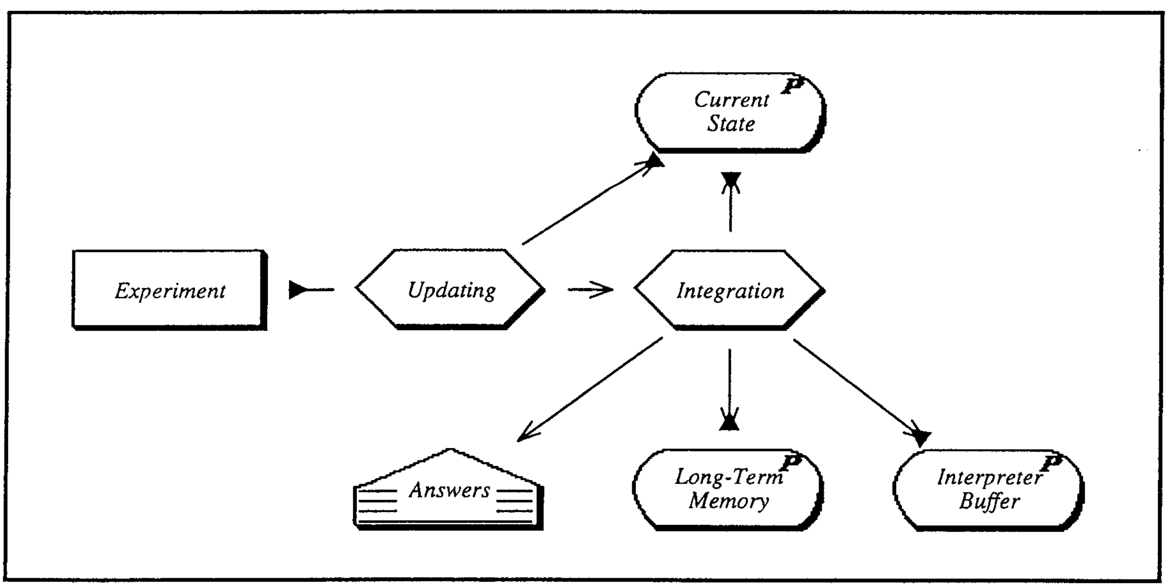


figure 1 - the COGENT object-level representation of the simulation

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

For the purposes of this paper, the "Experiment" compound is used only as a means of feeding information to the system simulating the child's mental processes, and will not be discussed in any great detail. Note that for the bag experiment, the simulation must include the correct answering of three "belief establishing" questions prior to the five questions of main interest within the experiment. The belief establishing questions were included within the experiment to ensure that the child had formed the correct representations of the state of the world prior to being tested on their memory for the sequence of events. These questions include the initial question of the Smarties task (What do you think is in the tube?), a repeat of the question to ascertain that the child believes the tube is empty (What is in the tube now?) once the transfer operation has taken place, and a question to ensure that the child has tracked the transferral of the supposed Smarties (What do you think is in the bag?). In the experiment, after asking one of the belief establishing questions, the experimenter waited until the child had answered before continuing with the procedure. Accordingly, in the simulation, no further input was fed to the system until the cycle after the system had output the answer to the previous question. This protocol was observed throughout all the simulations.

The Smarties Simulation.

We assume that the 30% of 3- and 4-year olds who pass the Smarties test do so by accessing a long-term memory (LTM) representation of the likely contents of a Smarties tube, so we do not attempt to deal with this question in any detail here. This is consistent with the developmental literature, which has focused only upon those children who fail. The initial simulation then, must be one that gives a "reality" answer to a "belief" question under the circumstances of the Smarties experiment. The experimental procedure is modelled by adding propositions about the current state of the environment a cycle at a time to an "environment" buffer, within the Experiment compound, which is read by the updating process. The Current State Buffer is a representation of current environmental contingencies. This is kept up-to-date by **destructive updating** which occurs by the operation of the following rules:

RULE 1.
IF: A is in Experiment: Environment
not A is in Current State
THEN: add A to Current State

RULE 2.
IF: in(X,Y) is in Experiment: Environment
in(X,Z) is in Current State
THEN: delete in(X,Z) from Current State

Thus, if *in(tube,smarties)* is in the Current State and *in(tube,pencils)* appears in the Environment, *in(tube,smarties)* is deleted from the Current State by the second of the above rules and is replaced by *in(tube,pencils)*.

The basic workings of the model of the Smarties task are as follows:

In LTM there is a generic representation of past experience of Smarties tubes,

g(in(tube,smarties)),

and a further rule in the integration process that states the contents can be matched to their containers on the basis of such past experience:

RULE 3.
IF: g(in(X,Y)) is in Long-Term Memory
object(X) is in Current State
not in(X,Z) is in Current State
THEN: add in(X,Y) to Current State

This rule is refracted, so that it only fires the first time its conditions are satisfied within a COGENT run. When a tube representation is added to the Current State Buffer, this rule fires and the inference is made that the tube contains Smarties. This information is overwritten, however, when the further information is added from the environment that the tube contains pencils. Thus, when the question regarding the contents of the tube is presented to the system

question(present(in(tube, What))),

the present representation of the current contents of the tube in the Current State Buffer instantiates the unknown variable in the question, and provides the only possible answer: *in(tube, pencils)*.

Questions are dealt with by being passed immediately over from the Current State Buffer to the Interpreter Buffer. Once a question is received in the Interpreter Buffer, it activates the relevant search processes according to the following rules:

RULE 4.
IF: question(present(X)) is in Interpreter Buffer
X is in Current State
THEN: clear Interpreter Buffer
add answer(X) to Interpreter Buffer

RULE 5.
IF: question(past(X)) is in Interpreter Buffer
record(Y) is in Long-Term Memory
X is a member of Y
not X is in Current State
THEN: clear Interpreter Buffer
add record (Y) to Interpreter Buffer
add answer (X) to Interpreter Buffer

Thus, the unknown variables within the question are instantiated either in the Current State Buffer or in LTM, and translated into an answer format. All answers within the Interpreter Buffer are immediately sent to the output processes represented in the diagram by the triangular "Answers" block.

The Bag Simulation.

In the case of the bag experiment, the simulation is a little more complex. In particular, we have to tackle the creation

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1897676-67-0

of event records. To do this, a rule must fire when an event is perceived to end. This rule translates all information currently being processed (the contents of the Interpreter Buffer), together with the current representation of the environment (the contents of the Current State Buffer) into an LTM format. In the hypothesis underlying the experimental procedure, the event was signalled to be at an end by a contextual change, the removal of the tube. In the simulation, a record is closed if there are more objects represented in the Current State Buffer than are present in the environment. This is captured formally by the updating rule:

RULE 6.
IF: Objects is the list of all object(X) such that
object(X) is in Experiment: Environment
Representations is the list of all object(X)
such that object(X) is in Current State
A is the length of Objects
B is the length of Representations
B > A
THEN: send close_record to Integration

Upon receiving the close_record trigger, a further rule fires within the integration process which transforms the information within the Current State Buffer and the Interpreter Buffer into a list structure in LTM. The Interpreter Buffer is then cleared.

Simulation Results.

The basic simulation can easily handle the results of the first group of children, those who were scored correct on the bag question (group A). When asked the bag questions, the simulation of this group of children has a record available containing the previous belief concerning the bag's contents,

in(tube,smarties)

which it can use to answer the first question (BAG: BELIEF: PAST), in accordance with rule 5. When asked the second bag question (BAG: REALITY: PRESENT), a Current State representation of the bag's current contents is employed to answer this question in accordance with rule 4.

Seven out of eight of this group of children were also scored as correct for the tube question. In the model, the tube question is handled by the existence of a record available in LTM which can be retrieved to answer the question. The creation of this record was triggered by the removal of the tube. Note that the record does *not* contain a verbatim representation that the tube contained marbles. Instead, the record contains the representation that the contents of the tube were emptied into the bag:

action(empty(tube,bag)),

that the tube is now empty:

in(tube,[]) (where [] denotes the empty set),

and that the bag contained marbles. To correctly answer questions regarding the initial contents of the tube (questions 3 and 5, TUBE: BELIEF: PAST and TUBE: REALITY: PAST) a further rule is necessary to allow the inference that the tube's contents can be ascertained by backwards reasoning from the bag's contents, and the fact that the contents of the tube were entered into the bag. Formally, this rule is:

RULE 7.
IF record(Y) is in Interpreter Buffer
question(past(in(A,B))) is in Interpreter Buffer
action(empty(A,C)) is a member of record(Y)
in(C,D) is a member of record(Y)
THEN: clear Interpreter Buffer
add answer(in(A,D)) to Interpreter Buffer

This rule is triggered if the current representation of the tube's contents is identical to the retrieved LTM representation. Since the child is presumably not expecting to answer a "present" question at this point, the rule allows the search, via inference, for an alternative "past" answer. Note that the simulation demonstrates that Morton's (1997, p. 938) comment that "the conditions are the same" for the tube questions of the bag experiment and for the same questions in the Smarties experiment is not strictly necessary when analysed in terms of the underlying theory. In this simulation, when the inference rule regarding the transferral operation is manually prevented from firing the default answer from the system to the tube questions is that the tube was empty. Since the child was shown the empty tube during the bag episode this forms part of the same record. The full contents of this record are displayed below:

record([[in(bag,smarties), in(tube,[]), object(bag),
action(empty(tube,bag)), object(tube)]
action(remove(tube))]).

With this set of rules, the simulation therefore produces the same answers in the bag experiment as seven out of eight of the children in group A.

The initial results of those children who were scored as giving "reversed" answers (group B) to the bag question need to be explained differently. Recall that these children gave reality answers to belief questions and vice versa. The simulation of this situation uses the same basic structure as the simulation of group A (the "corrects"). However, it is assumed that the group B children attempt to answer all questions initially from their current state representation of the world. Arguably this is less effortful than retrieving information from LTM (see Morton, Hammersley & Bekerian, 1985 for a discussion of the complexities of retrieval from LTM). In effect, we assume that the tagging of questions as referring to past and present is not as well established in this group as in group A. The group B children, then, are not forced to search LTM in response to a PAST question. Rather, they only look in LTM when the Current State search has failed. Since the Current State Buffer representation is one of reality rather than belief, these children's default strategy results in a reversal of belief and reality answers.

Briefly, the simulation of this state of affairs works as follows. The "past" and "present" modifiers in the input are ignored in the integration process by rules 4 and 5, and, instead, all questions are followed by an initial search in CS. This leads to the initial mistake. The reversal of the situation with the next question is simply implemented by making that the look-up rule for information in Current State into a refracted rule so that it cannot be used as a default when the next question is asked. This is the "present" reality question, and the only way the child can answer the question is by searching for a long-term memory representation with information about the contents of the bag. This is found in the record which specifies

in(bag,smarties)

resulting in a reversed pattern of results.

This simulation works well when only the bag question is considered, but runs into problems when the tube questions are also added to the simulation's input, since it produces a further "reversed" pattern of results for these questions. In fact only one child in this group was scored as giving "reversed" responses to the tube question, and six were scored as correct. This failing will be considered in more detail later.

The final group of children to be considered (group C) gave the "reality" answers to "belief" questions. Working on the logic employed in the simulation of group B's results it is assumed that these children also ignore the past/present modifiers and attempt to answer the question in the simplest way possible, by retrieving an answer from the Current State Buffer representation. However, for these children the assumption is that the search rule for the Current State Buffer is not refracted. Accordingly, the simulation produces repeated answers from the Current State Buffer, which are identical to the "double" responses given by this group. Of the eight children who were scored as "doubles" on the bag questions, this simulation matches the repeated "double" scores of four of these children on the tube questions.

GENERAL DISCUSSION

Successes and Failings

The memory-updating explanation of the Smarties task is outlined by Morton (1997), and the 3-buffer architecture used here to simulate this theory was derived from Barreau (1997), (see Barreau, 1997 for an account of why a 3-buffer system is necessary). The resulting simulation, however, differs in significant ways from either of these accounts. It is intended to be a forerunner of a number of such simulations, building up a set of mutual constraints on later models of on-line processing by this age group (c.f. Barnard, 1985). As such, it has a number of distinct successes and flaws. Not least amongst its successes is that it is - to our knowledge - the only fully specified computational theory of 3- and 4- year olds failings at "false belief" tasks. Other accounts of these phenomena rely upon the assumption that children of this age suffer from a conceptual deficit in representing the beliefs of others, and their own earlier beliefs if inconsistent with current reality (e.g., Hogrefe, Wimmer & Perner, 1986; Perner, Leekam & Wimmer 1987), or else are in other ways not as completely specified as the account given here (Halford, Wilson & Phillips, in press).

Viewed as a modelling project in its own right, a number of flaws become evident with the current account. Firstly, if it is considered to be a straightforward account of the current data independent of theoretical statements put forward elsewhere (Barreau, 1997; Morton, 1997), then it suffers from a rather poor fit to the data in the case of group B, the "reversed" response children. The mechanism which allows for a reversed response to the bag questions should also produce reversed responses for the tube questions. However, the majority of children in this group (six out of eight) were scored as correct in this case.

Elsewhere, the fit to the data is better. The account given by the basic bag simulation is also able to account for the failure of children at the Smarties task with no change to the model, merely altering the input to simulate the change in task. This simulation correctly produces the same results as the "correct" group (A) on all the questions. The modified simulations for groups B and C also give the identical patterns of results to the children they were intended to model for the bag questions, and in the case of group C (the "double" responses) this success is repeated with the simulation giving the same results as the largest subset of these children.

The conclusion to be drawn from this pattern of success and failure is that although there is a large degree of agreement between the performance of the children and that of the underlying model, there is a flaw in the manner in which the model operates. In particular, it should not function in the same way in response to the tube questions as it did to the bag questions. There are two broad ways of accomplishing this. The first is to add other rules which would interpret the material in the record in response to questions concerning the tube. A backwards inference using rule 7 concerning belief could take

action(empty(tube,bag))
in(bag,smarties)

and come up with

in(tube,smarties)

to go along with the *in(tube,[])* already available in the record. The ordering of these two contradictory options in the buffers could give rise to the differences in responding to the tube questions among the children in group B.

The second general approach to the mismatch is to change the way in which the Group A children solve the questions. One approach is to create records of questions and answers. This would make the answer to the initial belief question available, even though the primary representation *in(tube,smarties)* has been deleted. Use of the record

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

```
record([[in(bag,smarties), in(tube,[]), object(bag),
    action(empty(tube,bag)), object(tube)]
    action(remove(tube))]).
```

would then be restricted to questions about the tube. This resembles the account given by Barreau (1997). To achieve all this, we will have to characterise the differences among the three groups of children somewhat differently. Both these options will be explored in the next phase of simulation.

ACKNOWLEDGEMENTS

The experimental data which provided the impetus for this simulation were provided by Sofka Barreau. The resulting simulation has greatly benefitted in various ways by discussions with Paul Abeles, Sofka Barreau, Rick Cooper, Cathy Dillon, John Fox and Kerry Sims.

REFERENCES

Anderson, J. R. (1996). *Rules of the mind.* Erlbaum, Hillsdale N. J.

Barnard, P.J. (1985) Interacting Cognitive Subsystems: A psycholinguistic approach to short term memory. In A. Ellis (Ed.) Progress in the Psychology of Language, (Vol. 2), Chapter 6, London: Lawrence Erlbaum Associates, 197-258

Barreau, S. (1997) Developmental constraints on theories of memory. PhD thesis, Department of Psychology, University College London.

Cooper, R., & Fox, J. (in press). COGENT: A visual design environment for cognitive modelling. *Behavior Research Methods, Instruments and Computers.*

Elman, J. L., Bates, E. A., Johnson, M. H., Karmiloff-Smith, A., Parisi, D., & Plunkett, K. (1996). *Rethinking innateness: A connectionist perspective on development.* MIT Press, Cambridge, Ma.

Halford, G., Wilson, W. H., & Phillips, S. (in press). Processing capacity defined by relational complexity: Implications for comparative, developmental, and cognitive psychology. *Behavioral and Brain Sciences.*

Hogrefe, G.-J., Wimmer, H., & Perner, J. (1986). Ignorance versus false belief: A developmental lag in attribution of epistemic states. *Child Development*, 57, 567-582.

Morton, J. (1997). Free associations with EPS and memory. *Quarterly Journal of Experimental Psychology,* 50A, 924-941.

Morton, J., Hammersley, R. H., & Bekerian, D. A. (1985). Headed records: A model for memory and its failures. *Cognition*, 20, 1-23.

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

Perner, J., Leekam, S. R., & Wimmer, H. (1987). Three-year-olds' difficulty with false belief: The case for a conceptual deficit. *British Journal of Developmental Psychology*, 5, 125-137.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Modelling Common-Sense Psychology and the False Belief Test

Stuart Watt
Knowledge Media Institute and Department of Psychology
Open University
Walton Hall, Milton Keynes, MK7 6AA, UK
+44 1908 653169
S.N.K.Watt@open.ac.uk

ABSTRACT

In this paper, we describe a cognitive modelling framework for common-sense psychology. We'll show a number of comparable cognitive models for different theories of common-sense psychology, and show that these models can help to illuminate some of similarities and differences between the differing theories.

Keywords

Common-sense psychology, theory of mind, false belief test, cognitive model

INTRODUCTION

Common-sense psychology — or people's common sense ability to think about our own and other people's minds — is currently being researched actively in several different disciplines. While this interdisciplinary collaboration can be very productive, it can lead to its own problems. This is exacerbated by complexity, both methodological and theoretical, of common-sense psychology itself.

Much of the problem is that nobody is really sure what common-sense psychology is, theoretically. Astington and Gopnik (1991), for example, distinguish between six different possible interpretations, all of which are subtly different. There are many different theories of common-sense psychology. Unfortunately, there is no common ground which allows these different theories to be compared and contrasted. In this paper, we'll introduce a cognitive model that can begin to play that role.

To compare the different theories, we'll use a standard tool from common-sense psychology, Baron-Cohen *et al.*'s (1985) false belief test. We'll begin by introducing and describing this test, and one of the theories of common-sense psychology, Leslie's (1987) 'decoupler' model. Although common-sense psychology is hugely complex, and can only be modelled in the most sketchy form, we'll show how Leslie's theory can be implemented as a cognitive model. Finally, we'll show how alternative theories of common-sense psychology can be represented as small variations on this model, and that we can draw some conclusions about the similarities and differences between the theories with this modelling framework.

MODELS OF COMMON-SENSE PSYCHOLOGY

While common-sense psychology has been a focus for recent research, most work in this either has either been experimental or purely theoretical; there are few cognitive models in this area, even though it is precisely the kind of area that modelling has proved so helpful for in the past (Samet, 1993). The exception is the work of Shultz (1988, 1991). All the models which have been developed, though, focus on small parts of the problem; for example, studying how people assess whether or not planned actions were intentional (Shultz, 1988).

We propose a different strategy. Instead of a narrow but deep model, we propose using a broad but shallow one; one which can be used to compare theories on a grand scale. With this level of modelling, we believe that even in the limited false belief test, we can help to clarify the similarities and differences between some of the grand-scale theories in the field.

THE FALSE BELIEF TEST

The false belief test has its origins in Premack and Woodruff's (1978) experiment to determine whether or not chimpanzees could reason about one another's mental states — whether or not they had a "theory of mind", another term for common-sense psychology. Unfortunately, there was a methodological problem with this experiment; their chimpanzee subject, Sarah, could use her own beliefs rather than reasoning about another's, because the two were identical. To prove that Sarah was really able to reason about another's beliefs, they had to show that Sarah could still predict another's behaviour when her beliefs were different from that other's — that is, when the other had beliefs which Sarah believed to be false.

Following these problems with Premack and Woodruff's experiment, Wimmer and Perner (1983) devised a false belief test, which evaluated a (human) subject's to ascribe definite but false beliefs to another. Baron-Cohen *et al.* (1985) later simplified Wimmer and Perner's test so they could compare autistic, Down's syndrome, and normal children at different ages. Baron-Cohen *et al.*'s simplified false belief test is shown in figure 1.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Baron-Cohen *et al.*'s false belief test is presented as a simple story. There are two puppets, Sally and Anne. Sally has a marble, which she keeps in a basket. Then Sally leaves the room, and while she is away Anne takes the marble out of the basket and hides it in the box. Sally comes back into the room.. The child subject is then asked the question: "where will Sally look for her marble?" Older children say that she will look in the basket, because although they know the marble is in the box, they know that Sally doesn't know it has been moved from the basket, and they can distinguish Sally's (false) belief from their own (true) belief. Younger children, on the other hand, and autistic children, do not distinguish between the two They simply say that Sally will look in the box. The false belief test, therefore, explores the change that happens as common-sense psychology develops.

Baron-Cohen *et al.*'s theory was that a failure in the development of common-sense psychology might be responsible for autism, and the results from their experiment (and others which followed) certainly seemed to bear that out. As a result, there has been a focus of interdisciplinary research which has led to a number of different hypotheses about the nature and development processes involved in common-sense psychology.

Figure 2 shows a model for one possible theory of common-sense psychology, Leslie's 'decoupler' model. At the heart of Leslie's model is a manipulator that is capable of pretence — of decoupling beliefs from one context and applying them in another. It is this that makes reasoning about false beliefs possible, because a child can use this decoupling mechanism to separate someone else's beliefs into a different context from their own.

Given this simple theory of common-sense psychology, we will now turn to the cognitive model, and show how Leslie's 'decoupler' model can be represented in a model. But first, a few words on the modelling environment that we'll be using.

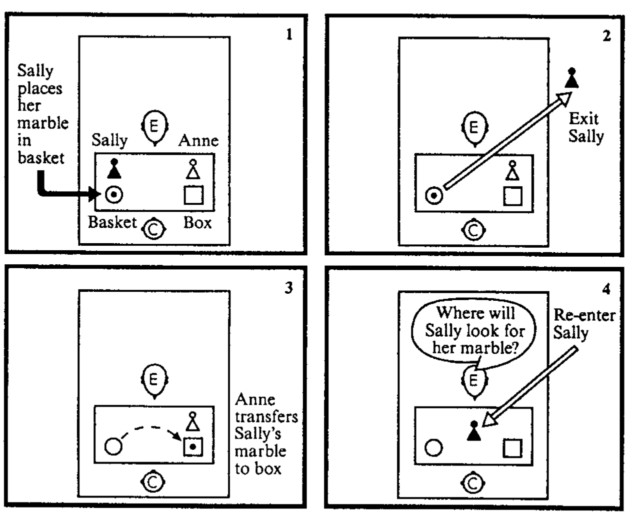


Figure 1. Baron-Cohen *et al.*'s (1985) false belief test

THE MODELLING ENVIRONMENT

Before we can build the models adequately, we need a representation language that is strong enough to do the physical and psychological reasoning required. In practice, the psychological parts of the model require the ability to reason about different contexts, distinguishing one agent's false beliefs from another agent's true beliefs. Something like a modal logic, therefore, is going to be required (Leslie, 1988, makes a direct comparison between the requirements for common-sense psychology and the properties of modal logics).

The model we present borrows this from McCarthy's (McCarthy & Hayes, 1969) 'situation calculus', where the effects of an event are described as a consequence relation between one state and another. At the core of McCarthy's calculus is a special function *result*, which represents the effects of an action on a situation by returning a new, modified, situation. The function $result(p, \sigma, s)$, where p is a person, σ is an action, and s is a situation, has a value which is a new situation representing the effects of p doing σ in s. For example:

$$inside(marble, X, s) \wedge \neg\, inside(marble, box, s) \Rightarrow inside(marble, box, t) \wedge \neg\, inside(marble, X, t)$$

where $t = result(alison, putin(marble, box), s)$

This says that if *marble* is inside something that isn't *box* in situation s, the effect of *alison* putting *marble* in *box* is a new situation t such that *marble* is no longer where it was (in X), but is now inside *box*.

The full situation calculus is more powerful and more complicated than this implies, but this subset of it is sufficient for the purposes of this model, and further, it doesn't need the heavy inference machinery that a complete modal logic would. The situation calculus, then, is strong enough for the model, fairly easy to use computationally, yet it retains the referential properties of modal logics (McCarthy & Hayes, 1969).

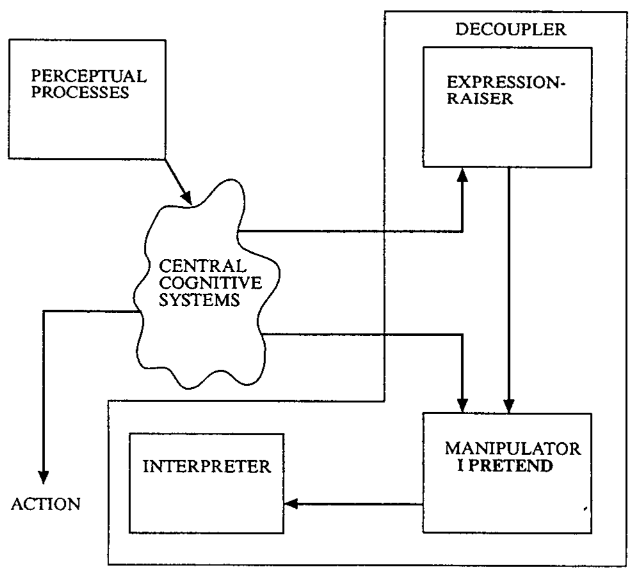


Figure 2. Leslie's (1987) 'decoupler' model

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

The model implements a modified subset of the situation calculus in a Prolog-like language embedded in Common Lisp. Apart from the Lisp-like syntax, there is only one significant difference from standard Prolog — variables are normally prefixed with a *?* question mark, but output variables in a clause head are prefixed with a ^ caret. *?value* and *^value* refer to the same variable.

MODELLING LESLIE'S 'DECOUPLER'

The base model for the false belief test comprises a number of separate modules. There include;

- a physical environment model,
- a basic physical reasoning module,
- a basic psychological reasoning module, and
- a script for the false belief test.

The Physical Environment Model

The first part of the modelling environment is a physical environment model which implements an event-driven simulation environment. As objects are physically moved from one place to another events are generated and passed to all objects equipped with sufficient perceptual apparatus to be aware of them.

The Physical Reasoning Module

Even in the false belief test, physical reasoning is needed. The basic physical reasoning module is shown in figure 3. This implements the rules that Alison (as we'll call the subject in the false belief test) uses to make predictions about what happens as a result of physical actions and events.

As far as physical reasoning is concerned, only three *result* actions are of interest. First, people can see an object being put into a container. Second, people can see an object being taken out of a container. And third, if a person enters a room, they can see all the objects (but not contained, or hidden, objects) within that room. All three of these actions serve to keep a person's model of the physical

```
;;; If we see ?object in a place ?container, then we find out
;;; where it was in the situation, and return a new situation
;;; so that it is now in ?container.

((result yes ?stance-to (place ?object ?container)
   ?situation ^new-situation) :-
 (member (inside ?object ?outer) ?situation)
 (difference ?situation
   ((inside ?object ?outer)) ?situationl)
 (append ?situationl
   ((inside ?object ?container)) ?new-situation))

;;; If we see an object being put into a new place, ?container,
;;; then again we find out where it was before in the situation,
;;; and return a new situation so that it is now in ?container.

((result yes ?stance-to (put-in ?object ?container)
   ?situation ^new-situation) :-
 (member (inside ?object ?outer) ?situation)
 (difference ?situation
   ((inside ?object ?outer)) ?situationl)
 (append ?situationl
   ((inside ?object ?container)) ?new-situation))

;;; If we see an object being taken out of a place ?container,
;;; we return a new situation so that it is no longer in
;;; ?container, but is now outside it, in ?outer-container.

((result yes ?stance-to
   (take-out ?object ?container)
   ?situation ^new-situation) :-
 (member (inside ?container ?outer) ?situation)
 (difference ?situation
   ((inside ?object ?container)) ?situationl)
 (append ?situationl
   ((inside ?object ?outer)) ?new-situation))
```

Figure 3. The basic physical reasoning module

The Psychological Reasoning Module

At the core of the model is a representation of one person's ability to reason about other people's mental states. This basic psychological reasoning module, corresponding to Leslie's theory of mind mechanism, is shown in figure 4. There are three *result* rules. The first rule is associated with *perceived* events; this is where the essence of psychological reasoning happens. The other two rules are associated with *believes* events, and are used for modelling the answering of questions; for this reason they print out an answer.

The first *result* rule uses the *ascribe* rule to keep all the notional worlds up to date with the *perceived* event. The *ascribe* rule implements the decoupler model in figure 2. It works like this. First, the *those* procedure is used to get all of *?self*'s beliefs out of the situation; this corresponds to *?self*'s notional world. Next, the *requote* procedure is used to raise all the expressions in the notional world, to create a new situation, *?situation2*. Then, the rule passes this new situation to the interpreter, through the manipulator. The manipulator is played by the *in-stance* procedure, which 'pretends' to be in the right context to handle the given event. The interpreter is called by the nested call to the *result* procedure. Finally, the nested call to result returns a new situation, *?situation3*, which is passed to *requote* again to restore its expression status in *?new-self-notional-world*. This is then used to replace the old notional world in the situation, and the modified situation is returned.

Perhaps this will be clearer with a more concrete example. Imagine that we ask *(result ?response sally (perceived sally (put-in marble box) ?S, ?NewS)*, in a situation *?S*. Because this is a *perceived* event, the first *result* rule will be applied, calling *ascribe*. The *those* and *require* procedures are used to go through the situation *?S*, decoupling all the relations *(believes sally ?X)* and generating a new situation *?S'*. Then the model applies the physical reasoning rules in this new situation *?S'*, to generate an updated physical situation *?R'*. The second requote call goes through *?R'* to restore its quotation status to normal, and returns *?R*. Finally, *?R* is used to replace all Sally's beliefs in *?S*, and the final situation returned in *?NewS*.

The Script for the False Belief Test

The final component of the model is a script for the

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

false belief test. This is shown in figure 5. There are two parts to this script. First, there are a serious actions which corresponds more or less to the movements of the characters in Baron-Cohen *et al.*'s story, shown in figure 1. Second, there are a number of questions; these are the kind of questions that an experimenter might ask a subject after acting out the scenario. It is the answers to these questions which reveal whether or not, or how, the child passes the false belief test.

So far, we have described a basic version of the theory of mind mechanism, a version which successfully models the passing of the false belief test. With this in place, we can now begin to compare this with some of the alternatives. In this paper, we will only look at three alternative theories of common-sense psychology, the simulation theory, the copy theory, and the situation theory.

```
;;; The rules for handling perceived events. When you
;;; perceive something and see that ?someone, sees the
;;; same thing, get ?someone's notional world into ?self-
;;; notional-world, and then, in that world, predict its
;;; physical effects. Then map these physical effects into
;;; changes to ?someone's notional world.

;;; Rule perceive
((result ^response ?someone
   (perceived ?object (?action ?other-object ?event))
   ?situation ^new-situation) :-
 (ascribe ?someone ^response ?someone
   (perceived ?object (?action ?other-object ?event))
   ?situation ^new-situation))

;;; Rule ascribe
((ascribe ?someone ^response ?other
   (perceived ?object (?action ?other-object ?event))
   ?situation ^new-situation) :-
 (those (believes ?someone ?something) ?situation
   ?notional-world)
 (requote (believes ?someone ?something)
   ?notional-world ?something ?situation2)
 (in-stance ?other-object ?action
   (result ?response ?other-object
     (?action ?other-object ?event)
     ?situation2 ?situation3))
 (requote ?something ?situation3
   (believes ?someone ?something) ?new-notional-world)
 (difference ?situation ?notional-world ?situation1)
 (append ?new-notional-world ?situation1
   ?new-situation))

;;; These are the rules for answering questions about
;;; people's beliefs. In effect, all that happens is that we
;;; look for the truth of the question in ?object's notional
;;; world.

;;; Rule answer-yes
((result yes ?someone (believes ?object ?something)
   ?situation ^situation) :-
 (member (believes ?object ?something) ?situation)
 (write-list (yes ?object believes ?something)))

;;; Rule answer-no
((result no ?someone (believes ?object ?something)
   ?situation ^situation) :-
 (not (member (believes ?object ?something)
             ?situation))
 (write-list (no ?object does not believe ?something)))
```

Figure 4. The basic psychological reasoning module

COMPARING MODELS 1: THE SIMULATION THEORY

The first alternative theory to be compared against Leslie's is the 'simulation theory', which is typified by a 'role taking' or 'perspective taking' approach. Gordon illustrates this by saying that "*Smith believes that* Dewey won the election" should be read as "let's do a Smith simulation. Ready? *Dewey won the election*" (Gordon, 1986, original emphasis).

According to the simulation theory, young children are simply unable to take other people's points of view. This can be modelled by dividing the main *perceive* rule into two — one for self, and one for others. In young children, the *perceive* rule for self functions as before, but the *perceive* rule for others does nothing. This is shown in figure 6.

When run, this seems to fail the false belief test correctly in that Alison doesn't give answers at all for either Sally or Anne; before Alison can pass the test she needs to acquire the ability to simulate, or take the role of, other people. This corresponds to the development of a simulation ability: "before internalising this system, the child would simply be unable to predict or explain human action [but] after internalising the system the child could deal indifferently with ac-

```
;;; Start by introducing the characters. The order doesn't
;;; matter much. Alison will become aware of all the other
;;; objects as soon as she enters the room.

(tell-model (put-in basket room))
(tell-model (put-in box room))
(tell-model (put-in marble room))

(tell-model (put-in sally room))
(tell-model (put-in anne room))

(tell-model (put-in alison room))

;;; Put the marble in the basket
(tell-model (put-in marble basket))

;;; Sally leaves the room
(tell-model (take-out sally room))

;;; Move the marble from the basket into the box
(tell-model (take-out marble basket))
(tell-model (put-in marble box))

;;; Sally comes back into the room
(tell-model (put-in sally room))

;;; Where does Alison think that the marble is?
(ask-object-if alison
  (believes alison (inside marble ?where)))

;;; Where does Alison think that Sally thinks the marble is?
(ask-object-if alison
  (believes sally (inside marble ?where)))

;;; Where does Alison think that Anne thinks the marble is?
(ask-object-if alison
  (believes anne (inside marble ?where)))
```

Figure 5. Actions and questions for the false belief test

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

tions caused by true beliefs and actions caused by false beliefs" (Gordon, 1986). This is why the kind of failure in the simulation theory is interesting; Alison simply fails to give answers for either Sally or Anne, because she failed to take their roles properly.

The second stage in the model, then, is the complete simulation rule, which implements a role taking strategy through the *in-self* primitive. This primitive has the effect of temporarily pretending to be a different self, and then handling the whole event in that context instead. It is this replacement second rule that allows Alison to pass the false belief test. The replacement rule which models this strategy is shown in figure 7.

There are a number of important conclusions to be drawn from this idea. First, in the simulation theory the behaviour involved in ascribing mentality to oneself is different from that involved in ascribing mentality to others. This contrasts with the theory of mind mechanism described earlier, where there is no difference between first person and third person ascription. This is shown by the rules' sensitivity to the *self* relation, which shows that there is an egocentricity involved in the simulation theory. The second point to note is that, in practice, the behaviour of this system is the same as that of the basic psychological reasoning module shown in figure 4, because the replacement second rule combines with the first to behave just as if there was a single rule using the *ascribe* action, a rule identical to the first *result* rule in figure 4. This is in accord with Perner's (1994) suggestion that, in practice, the difference between a theory and a simulation may be at worst one of emphasis.

```
;;; Here are the rules for the simulation theory. Initially, if
;;; we are seeing something ourselves, then we do the right
;;; ascription, otherwise we leave the situation alone. These
;;; two rules, together, replace the perceive rule in figure 4.

;;; Rule perceive-self, compare to perceive in figure 4
((result ^response ?someone
   (perceived ?object (?action ?other-object ?event))
   ?situation ^new-situation) :-
 (self ?someone)
 (ascribe ?someone ?response ?someone
   (perceived ?object (?action ?other-object ?event))
   ?situation ?new-situation))

;;; Rule perceive-other, compare to perceive in figure 4
((result ^response ?someone
   (perceived ?object (?action ?other-object ?event))
   ?situation ^situation) :-
 (not (self ?someone)))
```

Figure 6. Rules for the simulation theory (first version)

```
;;; The replacement second rule for the simulation theory. If
;;; we are not seeing something for ourselves, then we
;;; "pretend" to be someone else through the in-self primitive,
;;; and process the event as if we were that person. This rule
;;; replaces the perceive-other rule in figure 6.

;;; Rule perceive-other, compare to perceive-other in
;;; figure 6.
((result ^response ?someone
   (perceived ?object (?action ?other-object ?event))
   ?situation ^new-situation) :-
 (not (self ?someone))
 (in-self ?someone
   (result ?response ?someone
     (perceived ?object (?action ?other-object ?event))
     ?situation ?new-situation)))
```

Figure 7. Replacement rule for the simulation theory

COMPARING MODELS 2: THE COPY THEORY

The second model I'll compare against Leslie's theory of mind mechanism is Chandler's 'copy theory'. Chandler and Boyes describe younger children as behaving "as though they believe objects to transmit, in a direct-line-of-sight fashion, faint copies of themselves which actively assault and impress themselves upon anyone who happens in the path of such 'objective' knowledge" (Chandler and Boyes, 1982). They argue that this is the precursor to a complete theory of mind such as Leslie's, and therefore I'll only show the version which fails the false belief test — a version which passed the test would be identical to the complete model in figure 4.

From the complete model of the theory of mind mechanism corresponding to an adult theory of mind,

```
;;; Here are the ascription rules for the copy theory. Initially,
;;; if we are seeing something ourselves, then we do the right
;;; ascription, otherwise we leave the situation alone. These
;;; two rules, together, replace the perceive rule in figure 4.
;;; Note that these replacement rules are identical to those
;;; in figure 6.

;;; Rule perceive-self, compare to perceive in figure 4
((result ^response ?someone
   (perceived ?object (?action ?other-object ?event))
   ?situation ^new-situation) :-
 (self ?someone)
 (ascribe ?someone ?response ?someone
   (perceived ?object (?action ?other-object ?event))
   ?situation ?new-situation))

;;; Rule perceive-other, compare to perceive in figure 4
((result ^response ?someone
   (perceived ?object (?action ?other-object ?event))
   ?situation ^situation) :-
 (not (self ?someone)))

;;; Here are the answering rules for the copy theory. They
;;; have the effect of considering the target's notional world
;;; to be a 'copy' of the ascriber's. These rules replace the
;;; rules answer-yes and answer-no in figure 4.

;;; Rule answer-yes-self, compare to answer-yes in
;;; figure 4
((result yes ?someone (believes ?object ?something)
   ?situation ^situation) :-
 (self ?self)
 (member (believes ?self ?something) ?situation)
 (write-list (yes ?object believes ?something)))

;;; Rule answer-no-self, compare to answer-no in figure 4
((result no ?someone (believes ?object ?something)
   ?situation ^situation) :-
 (self ?self)
 (not (member (believes ?self ?something) ?situation))
 (write-list (no ?object does not believe ?something)))
```

Figure 8. Rules for the copy theory

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

we can modify the psychological reasoning module slightly to represent a child with a copy theory of belief. The main point of the copy theory is, in effect, that instead of ascribing beliefs to others, a 'copy' of one's own beliefs is used instead. Instead of building different notional worlds for Sally and Anne, both use the same, a copy of Alison's.

According to the copy theory, children simply do not ascribe real beliefs to others. This is shown by the modified *result* rules in figure 8, which replace the *result* rule in figure 4 so that beliefs are only ascribed to oneself. Note that these *result* rules are identical to the first (before full theory of mind) version of the simulation theory in figure 6. This is to be expected — Chandler's theory is an account of how children escape the kind of egocentricity that marks a simulation theory. But this is not the whole story in the copy theory; when children are asked about other people's beliefs, they answer by drawing on their own. For this, we also need to change the result rules for the believes relation; these are the rules which model how the child answers the kind of questions used in the false belief test. These changes are also shown in figure 8. Both the question rules are changed from figure 3 by using the *self* relation to find and use one's own beliefs, rather than anybody else's, to answer the given question. Because of this dependence on the self relation, this model shows that the copy theory, like the simulation theory, has an implicit (if rather better hidden) egocentricity.

There are more complex variations on the copy theory; for instance, Wellman (1990) argues that younger children have a copy theory of belief, but not of desires. This is outside the scope of this model because desire psychology isn't yet part of the modelling environment — this is an area for future work. But while the copy theory works to the extent that, when run, it correctly fails the false belief test, the model is quite radically different from an adult theory of mind, and it does seem to require a developmental jump of significant magnitude. All the egocentricity of the rules in figure 8 must be lost, and the child needs to learn to extend notional worlds to other people. This matches all the empirical evidence that is against a copy theory; Perner (1991) has argued convincingly that experiments involving inference from parts to wholes show that the evidence is against children having a copy theory at any age. Even so, this is something which could, in principle, be investigated further quite easily with this modelling approach.

COMPARING MODELS 3: THE SITUATION THEORY

The third reference comparison I'll make against the theory of mind mechanism is Perner's (1991) 'situation theory'. Perner's theory is substantially different from those presented so far because he draws a hard distinction between real and non-real situations, or contexts. The notional world an agent has of itself has a unique status. This is not mirrored in the basic psychological reasoning module in figure 3.

Perner argues that the reason younger children don't pass the false belief test is because the child subject applies the verbal form of questions incorrectly to the situation corresponding to reality, not to the non-real situation which has been played out by the puppets. According to the situation theory, unlike the copy theory, young children do have notional worlds, but they are not so good at understanding that a real question can apply to a non-real situation. Perner uses this distinction to explain why children who fail the false belief test are still capable of sophisticated notional world reasoning, such as that required by Zaitchik's (1990) 'false photograph' test.

Figure 9 shows the rules for the first version of the situation theory model — the version which models a child who cannot yet pass the false belief test. Note

```
;;; The key to Perner's model is a clear distinction between
;;; the status of one's own notional world, and those of others.
;;; This is represented in these models by adding a status flag
;;; to the rules which ascribe those notional worlds. This
;;; status value is knows for one's own notional world, and
;;; believes for other people's. These two rules, together,
;;; replace the perceive rule in figure 4.

;;; Rule perceive-self, compare to perceive in figure 4
((result ^response ?someone
   (perceived ?someone (?action ?other-object ?event))
   ?situation ^new-situation) :-
 (self ?someone)
 (ascribe ?someone knows ?response ?someone
   (perceived ?someone (?action ?other-object ?event))
   ?situation ?new-situation))

;;; Rule perceive-other, compare to perceive in figure 4
((result ^response ?someone
   (perceived ?object (?action ?other-object ?event))
   ?situation ^new-situation) :-
 (not (self ?someone))
 (ascribe ?someone believes ?response ?someone
   (perceived ?someone (?action ?other-object ?event))
   ?situation ?new-situation))

;;; The ascription rule is extended to take the additional
;;; status value. This value is used, instead of the fixed status
;;; value believes, to distinguish between one's own notional
;;; worlds and other people's. This rule replaces the ascribe
;;; rule in figure 3.

;;; Rule ascribe, compare to ascribe in figure 4
((ascribe ?someone ?status ^response ?other
   (perceived ?object (?action ?other-object ?event))
   ?situation ^new-situation) :-
 (those (?status ?someone ?something)
   ?situation ?notional-world)
 (requote (?status ?someone ?something)
   ?notional-world ?something ?situation2)
 (in-stance ?other-object ?action
   (result ?response ?other-object
     (?action ?other-object ?event)
     ?situation2 ?situation3))
 (requote ?something ?situation3
   (?status ?someone ?something) ?new-notional-world)
 (difference ?situation ?notional-world ?situation1)
 (append ?new-notional-world ?situation1
   ?new-situation))
```

Figure 9. Ascription rules for the situation theory

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

that the main *result* rule has been split into two: one for self and one for others. Superficially, this might look like egocentricity again, but this time the only difference between them is in the status they assign to different notional worlds, *knows* for self, and *believes* for others. Initially, as shown by the modified answer rules in figure 10, children can only link verbal questions to the world for self beliefs — the notional world with the status *knows*. Other notional worlds can and do exist, though; it is just that they cannot be accessed through verbal questions.

Perner claims that the principal change in children between the ages of two and a half and four is the acquisition of a representation theory, which allows them to recognise that questions can refer not to reality, but to worlds or situations that are represented — that is, worlds or theories with the believes predicate. This corresponds to the child's development from a situation theorist into a representation theorist, shown in the modified rules in figure 11.

Perner argues that this change isn't a radical overturning of the existing theory — the kind of radical change that makes the copy theory implausible. Instead, he suggests that the change that happens is a "theory extension" (Perner, 1991), a relatively minor change to the existing theory. This character if theory extension is important to any developmental account of common-sense psychology, because the empirical evidence is that common-sense psychology develops gradually, not in big jumps (Carey, 1985).

```
;;; These are the rules for answering questions about one's
;;; own beliefs. In this group, the "believes" question is
;;; coupled to the knows predicate of a notional world. These
;;; implement the 'self' half of the answer rules in figure 4.

;;; Rule answer-yes-self, compare to answer-yes in figure 4.
((result yes ?someone (believes ?self ?something)
   ?situation ^situation) :-
 (self ?self)
 (member (knows ?self ?something) ?situation)
 (write-list (yes ?self believes ?something)))

;;; Rule answer-no-self, compare to answer-no in figure 4.
((result no ?someone (believes ?self ?something)
   ?situation ^situation) :-
 (self ?self)
 (not (member (knows ?self ?something) ?situation))
 (write-list (no ?self does not believe ?something)))

;;; These are the rules for answering questions about other
;;; people's beliefs. This is a model of what happens before
;;; the representation theory is acquired, where the effect is
;;; to link into the knows predicate instead of the believes
;;; predicate. These implement the 'other' half of the answer
;;; rules in figure 4.

;;; Rule answer-yes-other, compare to answer-yes in
;;; figure 4.
((result yes ?someone (believes ?object ?something)
   ?situation ^situation) :-
 (not (self ?object))
 (member (knows ?self ?something) ?situation)
 (write-list (yes ?object believes ?something)))

;;; Rule answer-no-other, compare to answer-no in
;;; figure 4.
((result no ?someone (believes ?object ?something)
   ?situation ^situation) :-
 (not (self ?object))
 (not (member (knows ?self ?something) ?situation))
 (write-list (no ?object does not believe ?something)))
```

Figure 10. Answer rules for the situation theory

DISCUSSION

These models highlight several of the most important features of the common-sense psychology that underlies the false belief test, and show that these features can be emphasised by models that represent the different and competing theories in this field. Of the models presented, the one that seems to work best in this modelling framework is Perner's 'situation theory' model. The principal reason for this is that the apparent distance between passing and failing the false belief test is much smaller. For both the simulation theory and for Chandler's 'copy theory' there must be a radical development to the ascription of notional worlds. Perner's model clearly shows the character of theory extension which he suggests should be expected of a theory which matches the empirical psychological data on the development of these theories (Carey, 1985).

The simulation theory is quite similar to the version of Leslie's theory of mind mechanism that we have used as a base model — but both it and Chandler's copy theory show an apparent egocentricity. In practice, as I've argued, there are good reasons for supposing that in any real common-sense psychology, both theory and simulation aspects will be required and, therefore, a simulation theory will actually be complementary to, rather than alternative to, the models presented here (Perner, 1994). However, most of the people who have argued for a simulation theory have argued for it as an alternative to something

```
;;; These are the rules for answering questions about other
;;; people's beliefs. In this group, the "believes" question is
;;; correctly coupled to the believes predicate of a notional
;;; world. These rules override the default which gives the
;;; wrong answer in the first version of the situation theory.

;;; Rule answer-yes-other, compare to answer-yes-other
;;; in figure 10.
((result yes ?someone (believes ?object ?something)
   ?situation ^situation) :-
 (not (self ?object))
 (member (believes ?object ?something) ?situation)
 (write-list (yes ?object believes ?something)))

;;; Rule answer-no-other, compare to answer-no-other
;;; in figure 10.
((result no ?someone (believes ?object ?something)
   ?situation ^situation) :-
 (not (self ?object))
 (not (member (believes ?object ?something)
              ?situation))
 (write-list (no ?object does not believe ?something)))
```

Figure 11. Changes from the situation theory to the representation theory

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

like Leslie's 'decoupler' theory of mind mechanism, and therefore don't give much thought to how a simulation theory and a theory of mind mechanism might be combined in practice. But there is a twist to the simulation model; although it shows an apparent egocentricity, it can actually be functionally identical to Leslie's 'decoupler' model. This further backs up the arguments that the distinction between a theory and a simulation is one of interpretation rather than a real difference in behaviour (Perner, 1994).

It is, of course, possible to pursue this strategy still further developing models of some of the other models of common-sense psychology. Unfortunately, for an accurate model many of these require more complex models of perceptual apparatus (e.g. Baron-Cohen's, 1995, shared attention mechanism), or more complete models of common-sense psychology (e.g. Wellman's, 1990, simple-desire psychology) than have yet been developed within this framework. Even so, as a first attempt at the problem, the technique does seem to back up the existing points and arguments remarkably well, and to clarify the distinctions between the models which have been developed so far. And apart from anything else, at least within this limited scenario, it seems to work!

The usefulness of the modelling approach as a tool for studying common-sense psychology is a topic which deserves fuller discussion than is possible here. Even so, we believe that these models show cognitive modelling can help in this area.

REFERENCES

Astington, J. W., & Gopnik, A. (1991). Theoretical Explanations of Children's Understanding of the Mind. *British Journal of Developmental Psychology, 9*, 7-31.

Baron-Cohen, S., Leslie, A. M., & Frith, U. (1985). Does the Autistic Child have a 'Theory of Mind'? *Cognition, 21*(1), 37-46.

Baron-Cohen, S. (1995). *Mindblindness*. MIT Press.

Carey, S. (1985). *Conceptual Change in Childhood*. Cambridge, Massachusetts: MIT Press.

Chandler, M. J., & Boyes, M. (1982). Social-Cognitive Development. In B. B. Wolman (Ed.), *Handbook of Developmental Psychology* (pp. 387-402). Prentice-Hall.

Gordon, R. M. (1986). Folk Psychology as Simulation. *Mind and Language, 1*(2), 158-171.

Leslie, A. M. (1987). Pretence and Representation: the Origins of 'Theory of Mind'. *Psychological Review, 94*(412-426).

Leslie, A. M. (1988). Some Implications of Pretense for Mechanisms Underlying the Child's Theory of Mind. In J. W. Astington, P. L. Harris, & D. R. Olson (Eds.), *Developing Theories of Mind*. Cambridge University Press.

McCarthy, J., & Hayes, P. J. (1969). Some Philosophical Problems From the Standpoint of Artificial Intelligence. In B. Meltzer & D. Michie (Eds.), *Machine Intelligence 4* (pp. 463-502). Edinburgh: Edinburgh University Press.

Perner, J. (1991). *Understanding the Representational Mind*. Cambridge, Massachusetts: MIT Press.

Perner, J. (1994). The Necessity and Impossibility of Simulation. In C. Peacocke (Ed.), *Objectivity, Simulation and the Unity of Consciousness* (pp. 145-154). Oxford University Press for the British Academy.

Premack, D., & Woodruff, G. (1978). Does the Chimpanzee Have a 'Theory of Mind'? *Behavioural and Brain Sciences, 4*, 515-526.

Samet, J. (1993). Autism and Theory of Mind: Some Philosophical Perspectives. In S. Baron-Cohen, H. Tager-Flusberg, & D. J. Cohen (Eds.), *Understanding Other Minds: Perspectives from Autism* (pp. 427-449). Oxford: Oxford University Press.

Shultz, T. R. (1988). Assessing Intention: A Computational Model. In J. W. Astington, P. L. Harris, & D. R. Olson (Eds.), *Developing Theories of Mind*. Cambridge: Cambridge University Press.

Shultz, T. R. (1991). From Agency to Intention: A Rule-Based Computational Approach. In A. Whiten (Ed.), *Natural Theories of Mind: Evolution, Development and Simulation of Everyday Mindreading* (pp. 79-95). Oxford: Basil Blackwell.

Wellman, H. M. (1990). *The Child's Theory of Mind*. Cambridge, Massachusetts: MIT Press.

Wimmer, H., & Perner, J. (1983). Beliefs about Beliefs: Representation and Constraining Function of Wrong Beliefs in Young Children's Understanding of Deception. *Cognition, 13*, 103-128.

Zaitchik, D. (1990). When Representations Conflict With Reality: The Preschooler's Problem With False Beliefs and 'False' Photographs. *Cognition, 35*(41-68).


```
;;; Trace output for Leslie's 'decoupler' model, simulation
;;; theory (final version), and situation theory (final
;;; version). Compare to the results of Baron-Cohen et al.'s
;;; (1985) false belief test.

yes alison believes (inside marble box)
yes sally believes (inside marble basket)
yes anne believes (inside marble box)

;;; Trace output for simulation theory (first version).

yes alison believes (inside marble box)
no sally does not believe (inside marble ?where)
no anne does not believe (inside marble ?where)

;;; Trace output for copy theory and situation theory (first
;;; version).

yes alison believes (inside marble box)
yes sally believes (inside marble box)
yes anne believes (inside marble box)
```

Figure 12. Trace output from the different models

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Initial explorations of modifying architectures to simulate cognitive and perceptual development

Gary Jones
Department of Psychology
University of Nottingham
Nottingham NG7 2RD, UK
+44 (0) 115 951 5361
gaj@psychology.nottingham.ac.uk

Frank E. Ritter
Department of Psychology
University of Nottingham
Nottingham NG7 2RD, UK
+44 (0) 115 951 5292
frank.ritter@nottingham.ac.uk

ABSTRACT

We modified a cognitive architecture (ACT-R) and an attached interaction architecture (the Nottingham interaction architecture) to simulate developmental changes in problem solving. We started with an existing model that fits adult data on a blocks world task used to study the development of problem solving in children. We modified the model and architectures in three, independent ways to simulate a younger problem solver: (a) reduced the working memory, (b) deleted a piece of knowledge, and (c) reduced the accuracy of vision. We found that our modifications allowed the model to fit 7 year old's data better but not perfectly. These results suggest that cognitive models and their architectures can help answer the question of "What develops?"

Keywords

Cognitive architectures, development, problem solving, working memory, vision, ACT-R, interaction.

INTRODUCTION

As children grow older, they tend to be more able to learn new strategies and tasks, and be more efficient at, those strategies and tasks that they knew previously (e.g. Siegler, 1986). What changes are occurring in order for this to happen? It would be useful to be able to specify in information processing terms how the behaviour seen at each age is achieved, and therefore what the differences are between ages (Simon, 1962).

The solving of physical puzzles is a good area in which to examine differences in behaviour. A detailed analysis of the task behaviour is possible via videotape. Many strategies will be readily visible, reducing the need for the experimenter to infer what mental structures and strategies are being used. For this reason, a physical problem solving puzzle, the "Tower of Nottingham", is used to study differences in children's behaviour and the factors influencing them.

The Tower of Nottingham

The Tower of Nottingham task involves building a pyramid from 21 wooden blocks (see Figure 1). There are six layers to the pyramid, the lower five consisting of four blocks each, with a single block as the top layer. The blocks in the lower five layers all share the same characteristics, differing only in size. Each layer is normally formed via two sets of paired blocks. For example, placing the peg of block A into the hole of block B brings the two half holes together to form a pair having a hole (a hole-pair). Similarly, placing

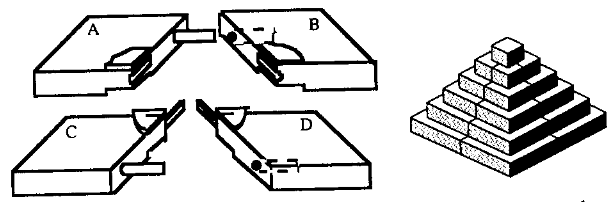


Figure 1. The blocks, on the left, that make up each layer, which are then stacked to create a tower, shown on the right.

block C and block D together forms a pair with a peg (a peg-pair).

Other strategies for creating a layer also exist, however, such as forming a pair having two pegs (blocks A and C) and a pair having two holes (blocks B and D).

There are two other features that may give rise to additional construction strategies. Each block has a quarter circle indent on top and a quarter circle depression underneath. When a layer is created, the quarter circles form circles in the centre such that layers can be stacked on top of each other by placing the circular depression of the upper layer onto the circular indentation of the lower layer. Constructions can be created by aligning the quarter circles so that they form a semi-circle.

Behaviour on the Task Varies with Age

Children of three are able to complete the Tower of Nottingham, yet performance improves with age all the way up to adulthood. For example, older children on the task accomplish more correct operations, produce less errors and take less time than their younger counterparts (Murphy & Wood, 1981; Wood & Middleton, 1975). Studying performance across ages on this task allows us to examine problem solving behaviour at each age and the differences in problem solving between ages.

The Use of Cognitive Models and Cognitive Architectures

Computational modelling across ages requires defining the behaviours that occur at each age (or performance level), because the model will require the knowledge and procedures that children may be using at each age. Where the behaviour cannot be defined in these terms, the model makes predictions about the missing elements. Therefore modelling task behaviour can help provide a means of defining how the different behaviours are generated.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

This enables a method for examining to what extent changes in task performance can be attributed to differences in knowledge and to what extent changes in task performance can be attributed to developmental processes. Existing models of development have only really considered differences in knowledge as the reason for changes in task performance, and have largely ignored the developmental processes that various developmental theories put forward (e.g. changes in working memory).

Early production system models of development, such as that of Young (1973), model differences in task performance by altering the rule set (i.e. the knowledge) within the production system. Klahr and Wallace (1976) implement possible developmental factors in their production system model of development (such as visual memory), but do not explore their effects.

Modelling techniques which have not used the production system style view development as being experience with the task, which can be seen as implicit knowledge. In the connectionist model of McClelland and Jenkins (1991), improved performance is attained by further training of the network on the task. In Siegler and Shipley's (1995) Adaptive Strategy Choice Model, improved performance is achieved by the model learning through experience of the task which strategies to employ for which sums.

All of these models have had success when they have been compared to subject data. However, developmental theory suggests that there are further changes occurring that also influence development. To what extent are these changes able to influence performance?

Two approaches stand out for creating a model of our task. One method is to model a lower performance level and see if that model can then progress to the higher performance levels that we see on the task. The other method is to begin at the highest performance level (that of adults), and then see if reduced versions of this model show behaviour that looks like lower performance levels. We have chosen to start with the simpler (adult) behaviour and work towards the more chaotic (child-like) behaviour.

We wish to examine how changes in both knowledge *and* development can influence task performance. To do this, we will begin with an adult model of our task and then impair it in theoretically motivated ways. By examining performance of the model after these changes, we hope to see to what extent the impairment can account for lower performance levels (those of children).

Cognitive architectures are important here as well, for they should also guide us (together with developmental theory) as to what are the sensible changes to make to the architecture. However, the role of change in architectures, with particular reference to development, has been rarely studied. The first definitions and implementations of cognitive architectures stressed that architectures do not change across tasks (Newell, 1990, p. 81). Newell (1990) argues that within Soar, development is just learning, and the architecture remains the same. Development is not mentioned with respect to ACT-R (Anderson, 1993). For these reasons we will look towards developmental theory as to what changes to make to the architecture.

Overview of the Paper

In the remainder of this paper, we will first describe the adult model upon which we base the other models. We describe its structure and the set of blocks that it interacts with. The model has been improved since it was last reported (in Jones & Ritter, 1997), and although the fit to the data is not improved substantially, it does enable the model to be broken in more theoretically motivated ways. We therefore describe the model in detail here. The stage is thus set for describing the three changes we make to the architecture. Each of the changes is described in terms of why they are suggested by developmental data, how they have been implemented, either in ACT-R or the Nottingham interaction architecture, and the effect they have on the model's behaviour. We conclude with a summary of these changes and the implications they have for the disentangling of what changes in cognitive development.

THE ADULT MODEL

The adult model is based on the ACT-R cognitive architecture (Anderson, 1993). In the development of the adult model the architecture has in part been used as a vehicle for the development of our own theories of performance on the task, although the model is consistent with most of the principles of ACT-R such as being goal driven, giving activation to memory elements, subjecting activation to both decay and noise, being rule based, and so on.

A simulation of the task also exists (see Figure 2), which is written in Garnet (Myers, et al., 1990). The simulation contains a full graphical representation of the task (all blocks and features), which is 2 1/2 dimensional—blocks cannot be turned on their side or held in mid-air, but can be face-up or face-down.

The simulation also represents an eye and two hands. The eye and hands are designed to meet a set of requirements identified for creating a psychologically plausible architecture for interacting with an external task (Baxter & Ritter, 1996).

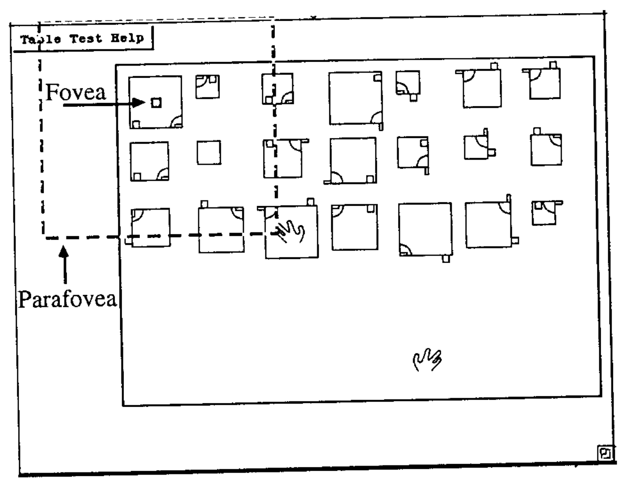


Figure 2. The Tower of Nottingham interaction interface.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

The eye is able to saccade and fixate, and passes to the model what it sees with regard to blocks and constructions (e.g. a peg-pair will be represented as a construction having two blocks that are flush on their outer edges and have their quarter circles and halfpegs aligned).

The visual information passed to the model is based upon where blocks are positioned in relation to the fovea. Three areas are defined: fovea, parafovea and periphery. Full information is passed for blocks or features in the fovea and parafovea, though the parafovea subjects features and block sizes to noise. For items in the periphery, the eye only returns to cognition a block ID. The hands are able to pick up, drop, rotate, turn over, fit, and disassemble blocks.

The model contains 226 rules which allow it to complete the task. The rules also interact with the simulation of the task, directing the eye and the hands. Within the model, all blocks and block features have an associated activation level. When several rules are instantiated, the one with the highest activation is selected. Therefore, in general, rules fire whose conditions have the most active blocks and block features in them. The activation levels are subject to *decay* each cycle, such that when they fall below a specified level (the *retrieval threshold*) they can no longer be matched in conditions of rules. Activation is raised based on what the goals of the model currently are, and by what blocks the fovea is looking at.

The learning mechanism that we included in the architecture is a simple method of increasing the chances of fitting blocks by specific features if a previous fit using the same features was deemed a success. Success is determined by the blocks in the construction being flush on their outer edges and having their quarter circles aligned (this is consistent with adult data on the task). Therefore, on some occasions the model may believe a successful construction has been made when in fact it has not (e.g. aligning the quarter circles of blocks A and B such that the blocks are not connected via a peg/hole). This learning mechanism approximates adult learning on the task (Jones & Ritter, 1997).

The model contains working memory and visual memory. Working memory contains all blocks and block features that are active enough to be matched in the conditions of rules (i.e. their activation is above retrieval threshold). Therefore, working memory is variable based on how active blocks and block features are in the model. Visual memory means we can remember some of the blocks that have been looked at previously even though they are now in the periphery. Visual memory is static (it is set at seven items), and compliments working memory since blocks in visual memory that are not in working memory can also be matched in conditions of rules.

Comparing the models with the data

It would be useful to compare subject performance on the Tower of Nottingham with the performance of models of the task using a metric that cannot be set as a parameter of the architecture. One such metric is the proportion of productions fired in the construction of each layer compared to the proportion of time subjects take in the construction of each layer. However, the task involves interaction with an external world, so timings for subjects include their perceptual and motor actions whereas the model production firings do not. This means timing estimates for interaction must be used in part of our model/subject comparisons.

We use the ACT-R default timing of 50 ms per production firing, which increases to 250 ms (Baxter & Ritter, 1996) for productions involving perceptual actions (eye movements and fixations), and 550 ms (Jones & Ritter, 1997) for productions involving motor actions (fitting and disassembling blocks). This enables a more complete comparison between model and subject timings. Production firing latencies in ACT-R also take into account activation of memory elements. In order for the influence of memory elements on production firing latencies to be negligible, the base level activation of memory elements was set to 10.0. Where other ACT-R parameters were used (decay, retrieval threshold), we adhered to the suggested default settings. The models begin with the initial knowledge of the task that subjects had, such as blocks of the same size go together, pegs go in holes, etc.

For every run of the model, the activation noise parameter within ACT-R was set to 0.005. This causes the activation of constructions and features in the model to differ, making the model's behaviour variable.

For comparisons between the model and subjects, measurements are given on an overall and layer-by-layer basis. The reason for reporting times and errors per layer is that subjects learn throughout the task. Since the model includes a learning mechanism, we want to see not only the effect that impairment to the model has upon overall behaviour, but also the impact it has upon the learning of the task.

We provide r-squared estimates for correlations between the model and subjects on a layer-by-layer basis, and t-test comparisons for summary data. These should only be taken as initial guides to the quality of the fit between the model and the subject data.

Comparison of the model with adult subjects

The adult subjects (N=5; taken from Jones & Ritter, 1997) had completed the task once. We compare 5 runs of the model to the 5 adult subjects.

The comparison of the adult model to the adult subject data is favourable. On the measures we will be using when we break the adult model, it fits the adult subjects reasonably well (see Table 1), although the model makes more incorrect constructions than subjects.

If we compare the times to complete each layer for the adult model and the adult subjects (see Figure 3), the trend of the model is the same as subjects—the time to complete each layer decreases until the final layer where the time increases slightly ($r^2 = 0.92$). The model takes more time to complete the task because it makes slightly more errors (see Figure 4; $r^2 = 0.67$).

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

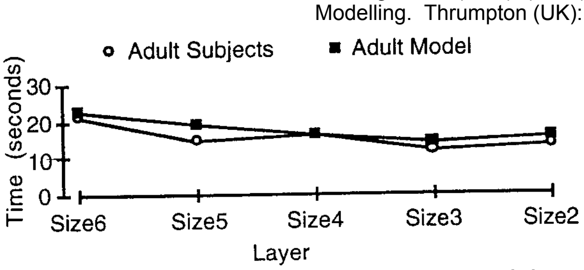


Figure 3: Time taken (seconds) to complete each layer for adult subjects and the adult model.

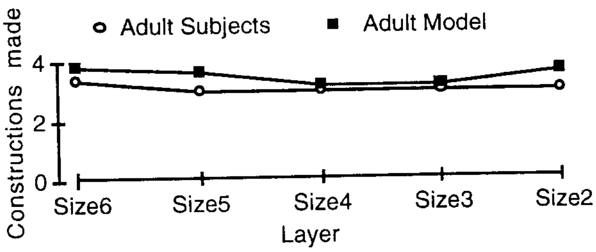


Figure 4: Construction attempts to complete each layer for adult subjects and the adult model.

The model provides a reasonable fit to the adult subject data in most of the behaviours that we are presently interested in. An exact fit on every measure is not essential because we will be examining the relative increases and decreases of these measures that result from the alterations that we carry out. The model fits the data well enough that it is fruitful to start exploring how problem solving changes when the architecture is changed to reflect that of younger problem solvers.

CHANGES TO THE ADULT MODEL

In order to examine how problem solving could change with development, we created three changed versions of the adult model. These changes are the most plausible based on the developmental literature and our knowledge of children's performance on the task. (a) We reduced the working memory capacity. (b) We removed a piece of knowledge. (c) We altered the accuracy of the parafovea. There are further changes that should be explored as well, such as basic processing speed, fovea size, and further changes to knowledge.

In this initial exploration we made each of these changes independently in order to keep the first order effects clear. For each change we explain its implementation, its rational, and its effect on problem solving.

The seven year olds we use to compare the altered models against were assisted on their first attempt at completing the Tower (contingently tutored, Wood & Middleton, 1975), and so we compared the model with their second attempt where they received no help in completing the Tower.

Reduced Working memory capacity model

Why

Several developmental theories suggest working memory capacity may influence task performance (e.g. Case, 1985; Halford, 1993). On the Tower of Nottingham, children have been noted to search with replacement (D.Wood, personal correspondence), a characteristic which may well be linked to working memory in that the children forget which blocks they have tried fitting together. On the Tower of Nottingham, seven year old children fit the same blocks together an average of 3.68 times, whereas this behaviour never occurs for adults completing the task.

How

Our model provides an easy way to manipulate working memory capacity to see what effect it has upon performance. In order to get a large, initial effect, we implemented this change to the model in three ways (the first two are parameters in ACT-R and the third is a parameter in the Nottingham interaction architecture). First, raising the retrieval threshold (from 0.0 to 2.5) means that constructions need to be higher in activation than in the adult model in order to be matched in rules. Second, raising decay (from 0.05 to 0.15) means constructions are forgotten more quickly than in the adult model. Third, reducing the number of items in visual memory (from 7 to 3) means that visual memory provides less support to working memory. The ACT-R parameters and mechanisms that we manipulate have also been used by Lovett, Reder and Lebiere (1997) in their ACT-R model of working memory differences, although they kept the parameter values constant and manipulated a third parameter. In this way they were able to model individual differences in working memory.

Measure	Adult Subjects	Adult Model	t-score
Total time taken to complete the Tower	80.6 s (13.3)	92.2 s (9.47)	$t(8)=1.59$ $p>0.05$
Total number of errors (incorrect constructions) made	0.2 (0.45)	2.4 (1.14)	$t(8)=4.017$ $p<0.05$
Errors where the blocks involved are of the same size	0.2 (0.45)	2.4 (1.14)	$t(8)=4.017$ $p<0.05$
Errors where the blocks involved are of different sizes	0	0	N/A
Number of times a construction attempt is made using the same blocks	0	0	N/A

Table 1: Mean (standard deviation) and t-scores for adult model and adult subject comparisons.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Measure	7yo Subjects	Reduced WM Model	t-score
Total time taken to complete the Tower	214.4 s (95.81)	134.0 s (24.1)	$t(8)=1.82$ $p>0.05$
Total number of errors made	7.6 (2.41)	5.4 (2.88)	$t(8)=1.31$ $p>0.05$
Number of times the same blocks are fitted together	1.75 (0.96)	2.0 (1.41)	$t(4)=0.27$ $p>0.05$

Table 2: Comparison between seven year old subjects and the reduced working memory model. Standard deviations, where appropriate, are given in parentheses.

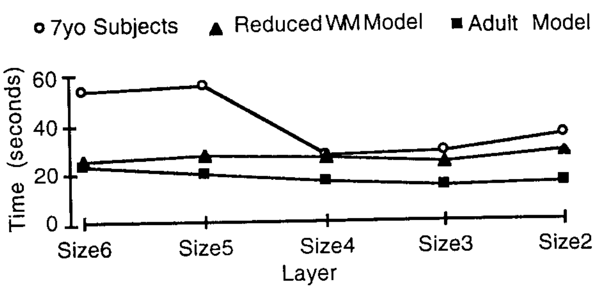


Figure 5: Time taken (seconds) to complete each layer.

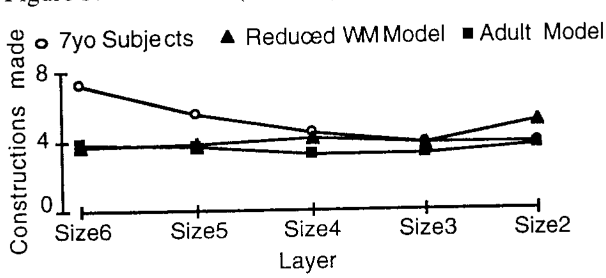


Figure 6: Construction attempts to complete each layer.

Predicted effect

Less working memory should lead to more search with replacement—the same pairs of blocks should be fitted together more often. A side-effect of searching with replacement is that the task should take longer and involve more errors.

Effect

Table 2 shows the summary statistics for the seven year old subjects and the reduced WM model. Figures 5 and 6 show comparisons on a layer by layer basis.

As predicted, reducing the working memory capacity in the adult model leads to fitting the same blocks together more often (from 0 in the adult model to 2.0 in the reduced WM Model). Increases are seen in both the time to complete the task (from 92.2 s in the adult model to 134.0 s in the reduced WM Model) and the number of errors (from 2.4 in the adult model to 5.4 in the reduced WM Model). This increase is not enough for the reduced WM Model to appear like a seven year old on the task. Although there are no reliable differences between the reduced WM Model and seven year olds in the total time taken and total number of errors, there are clear differences in the magnitude of these totals.

On a layer by layer basis, the reduced WM Model can be seen to not differ greatly from the adult model in terms of time and construction attempts made. However, the learning mechanism seems to be affected by the reduction in working memory capacity, because the original adult model provides a better fit to the seven year old subject data (times $r^2 = 0.85$; constructions $r^2 = 0.74$) than the reduced WM Model does (times $r^2 = 0.24$; constructions $r^2 = 0.63$). The original adult model and the reduced WM Model do not correlate at all (times $r^2 = 0.07$; constructions $r^2 = 0.05$).

Reducing the working memory capacity has allowed the model to fit the seven year old data a lot better than the adult model for overall times and errors, but at the cost of impeding the learning mechanism. This is probably because of the type of learning mechanism we use: there are less block features to be raised in activation upon success because working memory capacity is smaller. This suggests that further learning mechanisms must be used in order to fit the seven year old subject data better.

Less Knowledgeable model

Why

Children have a much smaller knowledge base to draw upon than do adults (e.g. Siegler, 1986). It is quite possible that children's knowledge of the Tower of Nottingham is less than that of adults. Examination of how seven year olds produce correct constructions compared to how adults produce correct constructions reveals that the children fit pegs into holes to produce a pair on 37 occasions yet only fit a halfpeg into a halfhole on 6 occasions. Adults fit via a peg and hole on 26 occasions as compared to fitting by halfpeg and halfhole 14 times. It is a possibility that children only learn about halfpegs and halfholes fitting together whilst they are completing the task.

How

Previously the model knew that halfpegs could fit into halfholes. This knowledge was deleted from the model.

Predicted Effect

The effect this will have upon performance is unclear. The number of constructions made via a peg and hole will rise sharply; however, the current learning mechanism offers no opportunity for learning that halfpegs and halfholes can fit together, and therefore it is expected that fitting by halfpegs and halfholes will be dramatically reduced. It will not be eradicated because there are other ways in which constructions can indirectly be made via a halfpeg/halfhole (e.g. quarter

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Measure	7yo Subjects	Less Knowledge-able Model	t-score
Total time taken to complete the Tower	214.4 s (95.81)	164.8 s (40.4)	t(8)=1.07 p>0.05
Total number of errors made	7.6 (2.41)	5.6 (3.36)	t(8)=1.08 p>0.05
Ratio of correct constructions fitted via peg/hole:halfpeg/halfhole	37:6	31:6	N/A

Table 3: Comparison between seven year old subjects and the less knowledgeable model. Standard deviations, where appropriate, are given in parentheses.

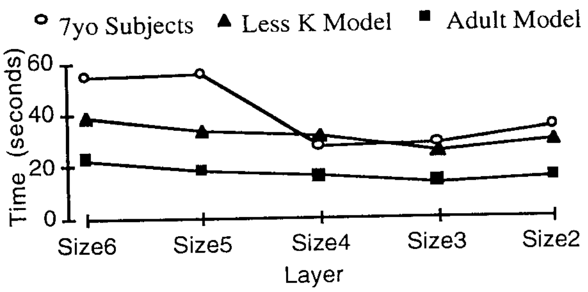


Figure 7: Time taken (seconds) to complete each layer.

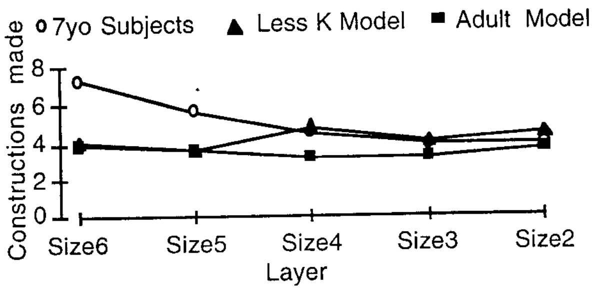


Figure 8: Construction attempts to complete each layer.

circles can be aligned in such a way that the halfpeg and halfhole fit together). We predict that the number of errors will remain the same. This is because fitting random blocks of the same size by a peg/hole arrangement and by a halfpeg/halfhole arrangement offer the same chances of success. The time to complete the task should not change, because no more errors are expected.

Effect

Table 3 shows the summary statistics for the seven year old subjects and the less knowledgeable (Less K) model. Figures 7 and 8 show comparisons on a layer by layer basis.

As predicted, deleting the knowledge that halfpegs fit into halfholes meant that fitting by pegs and holes rose sharply (from 14 in the original adult model to 31 in the less K Model), and fitting by halfpegs and halfholes dropped but was not eradicated (from 15 in the original adult model to 6 in the less K Model). The ratio of 31:6 compares favourably with the 37:6 ratio of seven year olds.

There were increases in both the total time taken to complete the task (from 92.2s in the original adult model to 164.8s in the less K Model), and the number of errors produced in completing the task (from 2.4 in the original adult model to 5.6 in the less K Model). This helps the less K model to fit the seven year old data (there are no reliable differences between the summary measures for the less K model and seven year old subjects, although there are clear differences on the layer-by-layer plots). Part of the increase in time can be attributed to more search being required (as we now have a reduced feature set because we no longer know that halfpegs fit into halfholes). However, most of the increase in time is because more errors are made. We do not yet have a valid reason for why this occurs.

As with the reduced WM model, we again see that the original adult model correlates better with the seven year old data on a layer by layer basis (original model and seven year olds: $r^2 = 0.85$ for times and $r^2 = 0.73$ for constructions; less K model: $r^2 = 0.73$ and $r^2 = 0.44$ respectively). This again suggests that the learning mechanism is impeded by the removal of knowledge. The type of knowledge removed means that learning must now occur over a reduced feature set. However, the reduced feature set still has the same chance of success as the old set, and it is therefore difficult to explain why the less K model does not learn as well as the original adult model.

Reduced Parafovea accuracy model

Why

Children find it more difficult to select blocks by size in the Tower of Nottingham task (Murphy & Wood, 1981). Although this is more pronounced for children of five years of age and below, seven year olds still average 1.8 constructions involving different sized blocks; the adults do not make any constructions involving blocks of different sizes.

How

We set the parafovea noise parameter for size to be 30 percent, representing a 30 percent chance that a block in the parafovea will be perceived as being a different size than it actually is (there are other possible mechanisms to implement this).

Predicted Effect

The increased size noise should mean that more incorrect constructions are produced involving blocks of different sizes. This increase in error should also lead to an increase in the time taken to construct each layer.

Effect

Table 4 shows the summary statistics for the seven year old subjects and the parafovea accuracy model. Figures 9 and 10 show comparisons on a layer by layer basis.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Measure	7yo Subjects	Reduced Parafovea Accuracy Model	t-score
Total time taken to complete the Tower	214.4 s (95.81)	126.2 s (24.6)	t(8)=1.99 p>0.05
Number of errors involving blocks of the same size	5.8 (2.59)	3.4 (1.34)	t(8)=1.84 p>0.05
Number of errors involving blocks of a different size	1.8 (2.68)	0 (0)	N/A

Table 4: Comparison between seven year old subjects and the reduced parafovea accuracy model. Standard deviations, where appropriate, are given in parentheses.

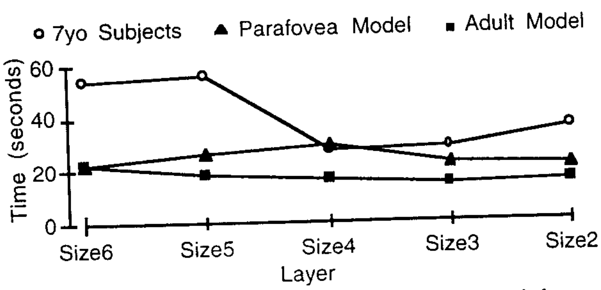


Figure 9: Time taken (seconds) to complete each layer.

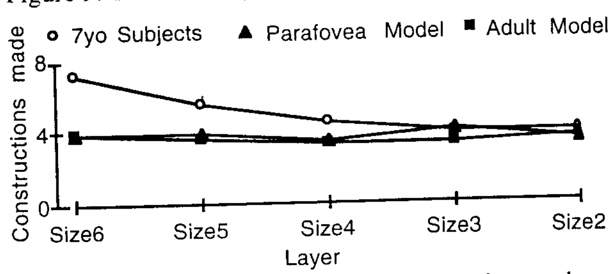


Figure 10: Construction attempts to complete each layer.

The results found go against our main prediction that there will be a greater number of constructions made which involve blocks of different sizes (neither the original adult model or the reduced parafovea accuracy model produce any). In hindsight, the reason for this is that when picking up a block, the model fixates upon it. Since at this point the block is in the fovea, the correct size is returned, and therefore if the block is the wrong size it is replaced. This provides an interesting result because it indicates that seven year olds either do not examine the block again once they have decided to pick it up, or their fovea vision is not as accurate as adults.

As predicted, there is an increase in the overall time taken (from 92.2 s for the original adult model to 126.2 s for the reduced parafovea accuracy model) and the number of errors produced (from 2.4 for the original adult model to 3.4 for the reduced parafovea accuracy model). This increase is not sufficient enough to make the reduced parafovea accuracy model appear to be like seven year old subjects on the task, although there are no reliable differences for either measure.

The reduced parafovea accuracy model does not correlate well with either the original adult model ($r^2 = 0.05$ for times; $r^2 = 0.03$ for constructions) or the seven year old subjects ($r^2 = 0.13$ and $r^2 = 0.29$ respectively). The increase in overall timings is probably due to the increase in visual search that is required due to the parafovea being less accurate. There should be no reason other than chance that there is an increase in construction attempts over the original adult model.

SUMMARY

We took an initial adult model and broke it in three ways to simulate a younger problem solver: cognitively (reducing working memory capacity), via knowledge (removing knowledge), and perceptually (reducing parafovea accuracy). All of these impaired the performance of the model to differing degrees and in different ways. None of the alterations was sufficient to produce behaviour similar to seven year old subjects, and all of the alterations indicated that more than one learning mechanism is required to fit the seven year old data properly. However, in breaking the adult model, we were able to show that changes that have been hypothesised to exist in younger problem solvers (i.e. developmental factors) do lead to different problem solving behaviour.

Further work must modify the model and its architecture in additional ways, motivated by developmental theory. There are several other ways to degrade the model's performance that we have not yet explored, such as changes in processing speed. These explorations will allow us to see how much each factor influences performance. The extent to which each factor contributes toward the observed behaviour indicates where our attention must lie in creating a complete model of seven year olds that is comparable and related to adult behaviour on the Tower.

However, we cannot simply consider each influencing factor independently because we have shown that this is not sufficient to produce the behaviour of seven year old subjects. The adult model will need several interacting changes to its architecture before its behaviour appears realistically to be like a younger problem solver. Therefore, not only will we be breaking the model in additional, independent ways, we will also be looking at combinations of modifications that interact. We expect the interactive effects to reveal more about performance at different ages, but simple changes are still required for our understanding and initial explorations.

This work indicates that the role of change in architectures, which has been little studied since the first definition, can be a fruitful way to use architectures. ACT-R includes many parameters. Before these parameters can be easily used for modelling development and abnormal problem solving, they need to be explored (or explained) to the extent that ranges for normal individual differences are known (e.g. Lovett, Reder, & Lebiere, 1997), and then that the interactions of these parameters are understood. A way to predict the performance of ACT-R models without running them in this area would be useful.

This work will eventually lead to models of five year old's and seven year old's behaviour solving the Tower that are based on modifying the adult model. We hope that these models will be able to explain individual differences within age groups as well as to explain the progression between ages (in terms of differences between the models rather than transition mechanisms). In both cases, we should be able to highlight the knowledge differences or architectural changes that lead to the differences in behaviour. Further learning mechanisms are also required in order that each model can learn from the task in order to perform to the standard of the older models. Explaining how and why problem solving changes with development is difficult, so further work will have to look at more than just this task.

We are now in a position to look at how problem solving changes across development. We have a cognitive model that performs the task. We can add and remove knowledge from the cognitive model and we can modify the architecture to represent developmental changes in cognition (the cognitive model based in ACT-R) and perception (the Nottingham interaction architecture). In the future we may be able to more directly answer "What develops?"

ACKNOWLEDGEMENTS

Josef Nerb wrote the first implementation of the Tower of Nottingham simulation. Heather Wood kindly supplied videos of seven year olds completing the Tower of Nottingham for us to analyse, and provided the tower picture. David Wood provided numerous useful discussions.

REFERENCES

Anderson, J. R. (1993). *Rules of the mind.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Baxter, G. D., & Ritter, F. E. (1996). Designing abstract visual perceptual and motor action capabilities for use by cognitive models (Tech. Report No. 36). ESRC CREDIT, Psychology, U. of Nottingham.

Case, R. (1985). Intellectual development: A systematic reinterpretation. New York: Academic Press.

Halford, G. S. (1993). *Children's understanding: The development of mental models.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Jones, G., & Ritter, F. E. (1997). Modelling transitions in children's development by starting with adults. In *Proceedings of the European Conference on Cognitive Science (ECCS '97).* 62-67. Manchester, UK: AISB.

Klahr, D., & Wallace, J. G. (1976). Cognitive Development: An information processing view. Hillsdale, NJ: Lawrence Erlbaum Associates.

Lovett, M. C., Reder, L. M., & Lebiere, C. (1997). Modeling individual differences in a digit working memory task. In M. G. Shafto & P. Langley (Eds.), *Proceedings of the Nineteenth Annual Conference of the Cognitive Science Society.* 460-465. Mahwah, NJ: Lawrence Erlbaum Associates.

McClelland, J. L., & Jenkins, E. (1991). Nature, nurture and connections: Implications of connectionist models for cognitive development. In K. VanLehn (Ed.) *Architectures for intelligence,* Chapter 3.

Murphy, C. M., & Wood, D. J. (1981). Learning from pictures: The use of pictorial information by young children. *Journal of Experimental Child Psychology, 32,* 279-297.

Myers, B. A., Guise, D. A., Dannenberg, R. B., Vander Zanden, V., Kosbie, D. S., Pervin, E., Mickish, A., & Marchal, P. (1990). Garnet: Comprehensive support for graphical, highly-interactive user interfaces. *IEEE Computer, 23*(11), 71-85.

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

Siegler, R. S. (1981). Developmental sequences within and between concepts. *Monographs of the Society for Research in Child Development, 46 (No. 189, 1-74).*

Siegler, R.S. (1986). Children's thinking. Englewood Cliffs, NJ: Prentice-Hall.

Siegler, R. S. & Shipley, C. (1995). Variation, selection and cognitive change. In T. J. Simon & G. S. Halford (Eds.), *Developing cognitive competence.* Hillsdale, NJ: Lawrence Erlbaum.

Simon, H. A. (1962). An information processing theory of intellectual development. *Monographs of the Society for Research in Child Development, 27 (2, Serial No. 82).*

Wood, D., & Middleton, D. (1975). A study of assisted problem solving. *British Journal of Psychology, 66*(2), 181-191.

Young, R. M. (1973). Children's seriation behaviour: A production-system analysis. Unpublished doctoral dissertation, Carnegie-Mellon University, 1973.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

How to Fatigue ACT-R?

G.M.G. Jongman
Experimental & Work Psychology
University of Groningen
Grote Kruisstraat 2/1, 9712 TS Groningen, The Netherlands.
+31 50 3636289
Linda@tcw3.ppsw.rug.nl

ABSTRACT

In this paper, an ACT-R model of mental fatigue is presented. This model is loosely based on Hockey's state regulation model of compensatory effort (Hockey, 1997). It appears that when spreading of activation is reduced, the ACT-R model can predict the performance changes Hockey describes, and furthermore, show how these may depend on the motivation of the participant. In a model of the Sternberg memory-search task, a reduction of the spreading of activation results in a change in strategy.

Keywords
mental fatigue, strategy use, cognitive control, ACT-R

INTRODUCTION

This paper describes a computational approach towards the investigation of mental fatigue. Mental fatigue is defined as the deterioration of mental performance due to preceding exercise of mental or physical activity (Meijman, 1997). As Meijman explains, it can be conceived of as a problem of keeping attention focused on task goals, or as a deficit in the cognitive-energetic control mechanisms. From his research it appeared that in some task conditions fatigued participants could protect their performance by means of compensatory effort, but in the most unfavourable conditions of the experiment (after 8 hours of work combined with sleep loss) people were no longer able to prevent deterioration of their performance. According to Shiffrin & Schneider (1977) there are two types of information processing: automatic and controlled. It appears that tasks that require more controlled processing are more sensitive to mental fatigue (Meijman, 1997). However, which cognitive processes are responsible for the changes in behaviour which are observed when people have to perform tasks for an extending period of time is a question that has not been answered yet. Bartlett (1943) hypothesised that the processes involved in planning, which is often ascribed to prefrontal functioning, are the ones responsible for these changes in behaviour. West (1996) subdivides the functioning of the prefrontal cortex into three processes. The first one is the inhibition of interfering processes and stimuli. The second process is a working memory process which enables the retrieval of information. The third process involves the preparation of responses. Summarising, there is some evidence that indicates mental fatigue is related to problems with cognitive control.

From many previous studies we already know that people seldom show a total breakdown of performance when they become mentally fatigued. A possible explanation for maintaining adequate task performance is that people change their strategy. More than 20 years ago, Shingledecker and Holding already hypothesised that when people become mentally fatigued they will shift their strategy of task performance towards a strategy that requires less mental effort (Shingledecker & Holding, 1974). In 1997, this hypothesis was brought out again by Hockey (1997). So, some people have hypothesised that mental fatigue involves a change in choice. However, a controlled study that investigates the details of this possible relation between mental fatigue and strategy use, still has to be done.

In order to predict and explain the role of cognitive control and strategy choice on the performance changes associated with mental fatigue, it is necessary to construct a detailed model of how these processes take place, and how they are influenced when people become fatigued. As the models mostly used in this field are mainly descriptive, the main purpose of this paper is to show how the valuable aspects of one of these models can be used to construct a computational model of mental fatigue, from which it will be possible to derive useful predictions of participants' behaviour. To this end, the next paragraphs will describe Hockey's compensatory control model (Hockey, 1997), which is a commonly known descriptive model of mental fatigue, and a cognitive architecture, ACT-R (Anderson, 1983; 1993). Together these components will be the basis for a computational model of mental fatigue.

A DESCRIPTIVE MODEL OF MENTAL FATIGUE

A model currently used for the investigation of mental fatigue is the state regulation model of compensatory control (Hockey, 1997). It is based on the concept of resources, which is described as "the availability of one or more pools of general-purpose processing units, capable of performing elementary operations across a range of tasks, and drawing upon common energy" (Gopher, 1986; Kahneman, 1973; Wickens, 1984). The model makes three assumptions. Firstly, it assumes that

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

behaviour is goal-directed. Further it is assumed that the control process is normally self-regulating. And, thirdly, the model assumes this regulation has costs (expressed in use of mental resources, levels of subjective strain, and physiological changes). An overview of the model is presented in figure 1.

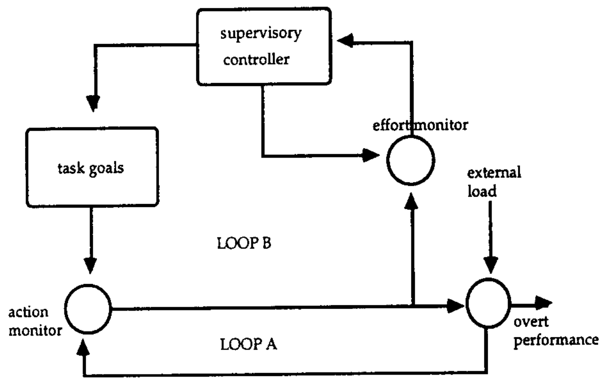


Figure 1. The state regulation model of compensatory control (Hockey, 1997)

The model distinguishes between two levels of control: a lower level, representing routine regulation (loop A), and an upper level, representing effort-based regulation (loop B). The effort-monitor monitors the level of demands in the lower loop. When the demands of the situation change, control will shift to the higher level (here called the supervisory controller) where several options for regulation are available. The model requires two levels for the effort monitor: a lower setpoint and an upper setpoint. This is the part of the model in which resources play an important role, for the upper setpoint represents the maximum level of effort that can be mobilised, which is dependent on motivation. Referring to Holding (1983), Hockey argues that this upper setpoint can be influenced by fatigue. When the perceived demands are too high, the maximum level of effort that can be mobilised should be increased, or the performance will decrease. Hockey describes four kinds of changes that can happen when people protect their performance. The first change he mentions is subsidiary task failure, for example the neglect of subsidiary activities or narrowing of attention. Second, people can make strategic adjustments as less use of working memory and greater use of closed-loop control. Third, maintaining performance could require compensatory costs. People would have to increase mental effort to attain the same performance. Finally, if no changes during task performance are observed, it is possible that people will show after-effects, for example express feelings of fatigue, or show a post-task preference for low-effort strategies.

To summarise, according to this model, task performance normally relies on routine regulation. In situations with high demands (e.g., stressful situations, situations in which the operator is mentally fatigued), task performance requires effort-based regulation (loop B). Thus, the model would predict that when people become mentally fatigued they would need a more effortful manner of control for the same task as before. However, it is not clear how that would lead to the four kinds of change Hockey predicts. It could be the case that when people become fatigued, they invest more effort in the task, change their strategy of performance, neglect subsidiary activities, or show after-effects. The model does not provide predictions about what people will actually do in these situations that require higher level control. A computational model is needed to refine these processes and deliver useful predictions for different situations. To this end, a rather brief explanation will be given of ACT-R (Anderson, 1993), an architecture of cognition, from which it is possible to construct a computational model of fatigue.

ACT-R

The reason for choosing the ACT-R architecture for the construction of a model of mental fatigue is twofold. For the investigation of mental fatigue the measurements of performance that are used most often are the reaction times for completing tasks, the (strategic) choices made during task performance, and the number of errors made by participants. A very attractive aspect of ACT-R is that it can make very detailed predictions about these three kinds of measurements. Furthermore, ACT-R is equipped with global parameters which, when changed, can cause qualitative, task-specific, changes in behaviour. These global parameters make ACT-R suitable for the construction of a model of mental fatigue.

The ACT-R Architecture

The ACT-R architecture distinguishes between two kinds of memory: production memory (memory for procedural knowledge, represented with production rules) and declarative memory (memory for fact knowledge, represented with chunks). Strategies are represented with (a number of) production rules, and additional declarative facts. The conflict resolution process selects production rules according their expected gain, as calculated by equation 1.

$$\text{Expected gain}_i = P_i G - C_i \qquad (1)$$

In this equation P represents the probability of success when using this production rule, G the value of the goal, and C the cost to reach the goal, using this production rule. The preliminary assumption of ACT-R is that cost is the time needed to reach the goal. From the production rules that match the current goal, the production rule that has the highest expected gain is tried first, which means that ACT-R tries to retrieve the declarative memory chunks necessary for the production to fire. Whether ACT-R succeeds in retrieving the chunks depends on the activation level of these chunks. When the activation of a

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

chunk drops below a certain threshold, the *retrieval threshold*, it cannot be retrieved anymore. The activation level of a declarative memory chunk is determined by equation 2.

$$\text{Activation}_i = \text{base level activation}_i + \sum_j \text{source-activation}_j * \text{associative strength}_{ji} \quad (2)$$

In this equation base-level activation represents how recently and frequently the chunk has been used before. The second half of the equation represents spreading activation. Source activation represents the attention given to the elements of the goal and association strength represents the likelihood that fact i is needed if fact j is part of the current goal. If all retrievals succeed, the production will fire, if not, the second-best production is tried. Furthermore it must be mentioned that ACT-R can learn the parameters of the model itself (e.g., the base-level activation, the associative strengths, the probability of success of a production and its cost).

A COMPUTATIONAL MODEL OF MENTAL FATIGUE

In the introduction two aspects of mental fatigue were mentioned: mental fatigue as a cognitive control problem, and mental fatigue as a process involving a shift in choice, a more motivational aspect. How can these aspects be represented in a computational model of mental fatigue? Therefore we have to determine how global parameters can interact with knowledge-specific parameters. In ACT-R two global parameters can be related to these aspects of mental fatigue. In the next two subsections these two parameters will be explained and the third section illustrates the influence of the values of these two parameters on the performance on a Sternberg memory-search task.

Mental Fatigue as a Problem Concerning Cognitive Control

As already mentioned in the introduction, West (1996) distinguishes three cognitive control functions: inhibition of interfering processes and stimuli, and two memory functions. A global parameter in ACT-R related to these functions is the source activation, which was described as a part of equation (2). Source activation spreads from the goal to related chunks, thereby creating more contrast between chunks which are relevant and irrelevant to the current goal. When source activation is low, the contrast between relevant and irrelevant chunks is low. As such, source activation has the same function as inhibition of interfering stimuli, which was described as one of the cognitive control functions possibly harmed by mental fatigue. When source activation is high, the probability of interference is low. When source activation is low, however, interfering stimuli can become problematic. It is also possible that due to low source-activation, the activation level of relevant chunks drops below the retrieval threshold, which means that relevant facts cannot be retrieved at all. Furthermore, there are already some indications that source activation is related to working memory. Lovett, Reder & Lebière (1997), for example, found that individual differences in working memory capacity can be simulated by changing the source activation. Therefore, it can be hypothesised that when people are fatigued, their source activation is lower.

Mental Fatigue as a Motivational Problem

Shingledecker & Holding (1974) and Hockey (1997) hypothesise that mental fatigue may also involve a shift in choice, more specifically, a shift toward strategies requiring less mental effort. This can be related to the motivation of the participants. The parameter closest to the concept motivation is the G parameter described before in equation (1), which represents the value of the goal. Literally, the G parameter represents how much time you are willing to invest in reaching the current goal. When the task does not involve time pressure, the value of the G parameter is partly determined by the motivation of the participant (Taatgen, 1997). So, it can be predicted that a highly motivated participant will favour strategies with a high probability of success, while participants with low motivation will favour strategies with less costs.

An Example: a Model of the Sternberg Memory-Search Task

The model described in this subsection is adapted from Anderson & Lebière (in preparation). The task the model performs is a modified version of the Sternberg memory-search task (Sternberg, 1969). In this task three letters are shown on a computer screen, which the participant has to keep in memory. These three letters are referred to as the memory set. The time the memory set is shown is long enough to read the letters, but not long enough to rehearse them. After that, an attention dot is shown, followed by a set of four letters, called the display set. The participant has to decide whether one of the letters from the display set was part of the memory set. The probability that this is the case is 50 percent. A new memory set is presented on each trial, which immediately starts after the participant has given a response, making the task self-paced.

The two strategies which can be used to perform the task are described in Anderson & Lebière (in preparation). The strategy that generally has the best speed-accuracy properties will here be referred to as *retrieve-and-check*. When the display set is shown, the participant focuses on the first letter in the set. He then retrieves the letter from the memory set with the highest activation. If this retrieved letter equals the attended letter in the display set the participant responds with a yes, else he moves on to the next letter in the display set. If there is a letter in the memory set corresponding with the attended letter, this letter will have the highest activation.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

The main production rules for retrieve-and-check are given below. This strategy will produce fast responses, since the retrieve-trace production will always succeed.

Retrieve-trace
IF the goal is to check if item x is in the memory set
  and there is some item y in the memory set
THEN the target is item y

Retrieve-yes
IF the goal is to check if item x is in the memory set
  and the target is item x
THEN say-yes

Retrieve-no
IF the goal is to check if item x is in the memory set
  and target is not equal to item x
THEN move on to the next item of the display set

The second strategy focuses on accuracy, but is less efficient. It is called *specific-retrieval*, since the participant specifically has to retrieve the memory set item that matches the current display set item. This will result in a higher accuracy, since it is impossible to retrieve a wrong item from the memory set. Another consequence, however, is that the retrieve-trace production will fail most of the time. This results in a longer reaction time, since failing production rules use the time it takes to retrieve items whose activation equals the retrieval threshold. The main production rules for this strategy are given below.

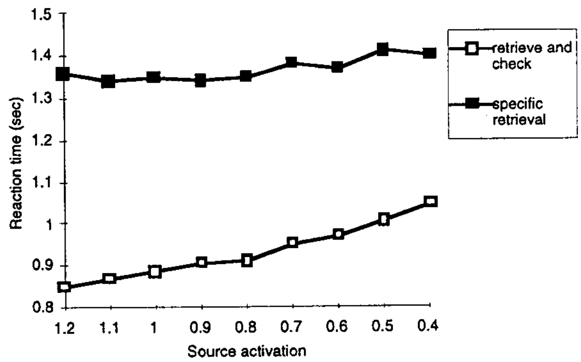
Retrieve-trace
IF the goal is to check if item x is in the memory set
  and item x is in the memory set
THEN the target is item x

Retrieve-yes
IF the goal is to check if item x is in the memory set
  and the target is item x
THEN say-yes

Retrieve-no
IF the goal is to check if item x is in the memory set
THEN move on to the next item of the display set

The retrieve-no rule has a lower expected gain than retrieve-trace, so it will only fire when retrieve-trace fails.

Source activation, which was proposed as a global parameter concerning mental fatigue, effects the retrieve-trace rule, since that rule tries to retrieve an item from the memory set. In the retrieve-and-check strategy the source activation ensures the right item is retrieved. Lowering the source activation will increase the probability of retrieving the wrong item, thereby producing more errors. In the specific-retrieval strategy lowering the source activation hardly influences the number of errors that will be made. This can be seen in figure 3 which presents some simulated data from the model. The figure also shows that for the retrieve-and-check strategy reaction times become slower when source activation is lowered. The reason for this is that the activation of the items in the memory set is lower, because they receive less source activation (see equation 2). In ACT-R it takes more time to retrieve an item when its activation is low.



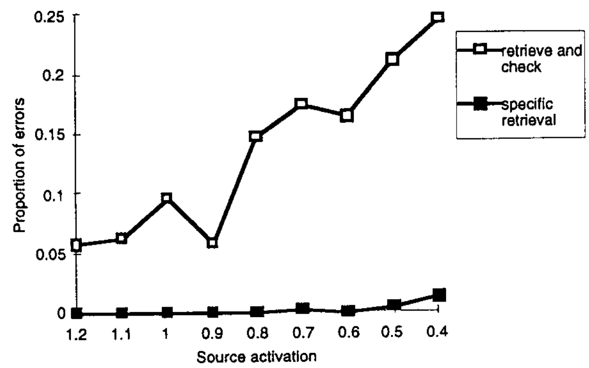


Figure 3. The changes in reaction times and proportion of errors for both strategies, as a result from lowering the source activation.

As already explained before, expected gain determines which strategy will be chosen in a particular situation. When people are fit, and thus have a high source activation, the expected gain of the retrieve-and-check strategy will be highest. However, according to figure 3, when source activation becomes lower, it can be predicted that at some point in time the expected gain of the specific-retrieval strategy will become the highest, and therefore a shift in strategy will be made. The exact timing of this strategy change is dependent on the motivation of the participant. Figure 4 illustrates the effect of motivation and source activation on the expected gain of the two strategies. The expected gain is calculated according to equation 1 using reaction time (from figure 3) as cost, and one minus the proportion of errors as probability of success. ACT-R's conflict resolution mechanism will choose the strategy with the highest expected gain. As can be seen from the figure, when the motivation of the participant is low (represented by a low value of the G parameter) and source activation is lowered, people still maintain the retrieve-and-check strategy, although this results in a great number of errors. However, when the motivation is higher and source activation is lowered, the participant will shift

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

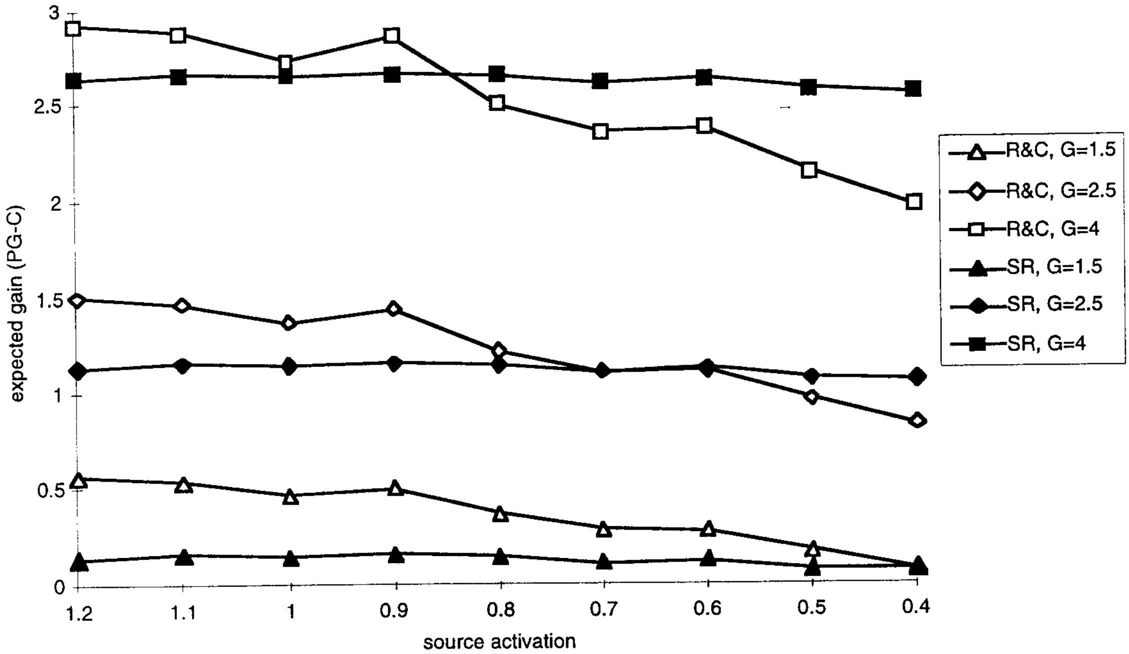


Figure 4. The expected gain of both strategies as a function of the source activation and the motivation (represented by the value of G) of the participant. R&C = retrieve-and-check, SR = specific retrieval.

towards the specific-retrieval strategy. Furthermore, the higher the motivation of the subject, the sooner this strategy shift will take place.

A shift in strategy, or strategic adjustment in Hockey's terms, is one change Hockey describes that can happen when people become mentally fatigued. The ACT-R model, however, can also predict such a change and show how this depends on the participant's motivation. Hockey's model describes that performance normally relies on routine regulation. When people become fatigued two situations can arise: either performance will decrease, or control will be shifted to a higher level (loop B in Hockey's model). What this shift in control involves is not completely clear from the model. The ACT-R model does show what a shift in control involves. When people become fatigued and routine-regulation is not adequate for task performance, the conflict resolution process in ACT-R will select a strategy that is less sensitive to fatigue. So, in this model, the change in cognitive control can be directly derived from the basic processes of the ACT-R theory.

Although an experiment to validate this model has not been done yet, some studies support the outcomes of the model. In two studies (Kerstholt, van Orden & Gaillard, 1994; van Orden, Gaillard & Langefeld, 1996) in which task instructions for the memory-search task focused on accuracy, mental fatigue manifested itself by increasing reaction times, which could indicate the use of the specific-retrieval strategy. In another study (Schellekens, Sijtsma & Vegter, in preparation) in which both accuracy and speed were emphasised, participants only had a fixed time to respond. In this experiment mental fatigue was accompanied by an increase in the number of errors. This decrease of accuracy can be explained by the fact that the time subjects had to respond was too short for the application of the specific-retrieval strategy, so participants had to stick to the retrieve-and-check strategy.

CONCLUSIONS AND RECOMMENDATIONS

As was shown in the previous section, the model provides detailed predictions of performance changes when people become mentally fatigued. Furthermore, the changes it predicts can be directly derived from the ACT-R theory, which allows for generalisation. Given an ACT-R model of a certain task, it is easy to predict the role of mental fatigue in task performance. It will be especially interesting to study the effects of manipulation of source activation on models of more complex tasks that allow participants more strategic freedom, since several authors have argued that these tasks are most influenced when people become fatigued (e.g., Bartlett, 1943; Meijman, 1997). The model also predicts that some tasks will hardly be sensitive to mental fatigue, for example, if the strategy used does not rely on source activation. However, the model has not been validated yet, so future experiments have to be carried out to support it.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

REFERENCES

Anderson, J.R. (1983). *The architecture of cognition.* Cambridge, M.A.: Harvard university press.

Anderson, J.R. (1993). *Rules of the mind.* Hillsdale, NJ: Erlbaum.

Anderson, J.R. & Lebière, C. (in preparation). *The atomic components of thought.* Hillsdale, NJ: Erlbaum.

Bartlett, F.R.S. (1943). Fatigue following highly skilled work. *Proceed. Royal Society, 131*, 247-257.

Gopher, D. (1986). In defence of resources: on structures, energies, pools and the allocation of attention. In G.R. Hockey, A.W.K. Gaillard & M.G.H. Coles (eds.). *Energetics and human information processing.* Dordrecht: Martinus Nijhoff.

Hockey, G.R.J. (1997). Compensatory control in the regulation of human performance under stress and high workload: A cognitive-energetical framework. *Biological psychology, 45*, 73-93.

Holding, D.H. (1983). Fatigue. In G.R.J. Hockey (ed.) *Stress and fatigue in human performance.* Chichester: John Wiley.

Kahneman, D. (1973). *Attention and effort.* Englewood Cliffs, NJ: Prentice-Hall.

Kerstholt, J.H., van Orden, C.Y.D. & Gaillard, A.W.K. (1994). Effecten van vermoeidheid als functie van soort taak en sociale omgeving. TNO-rapport TM 1994 A-9, TNO Technische Menskunde, Postbus 23, 3769 ZG Soesterberg, The Netherlands.

Lovett, M.C., Reder, L.M. & Lebière, C. (1997). *Proceedings of the 19th Annual Conference of the Cognitive Science Society.* Hillsdale, NJ: Erlbaum.

Meijman, T.F. (1997). Mental fatigue and the efficiency of information processing in relation to work times. *International Journal of Industrial Ergonomics, 20*, 31-38.

Orden, C.Y.D., van, Gaillard, A.W.K. & Langefeld, J.J. (1996). Effecten van vermoeidheid en sociale omgeving op prestatie: de rol van feedback. TNO-rapport TM-96-A035, TNO Technische Menskunde, Postbus 23, 3769 ZG Soesterberg, The Netherlands.

Schellekens, J.M.H., Sijtsma, G.J. & Vegter, E. (in preparation). Immediate and delayed aftereffects of mentally demanding work hours. University of Groningen, Groningen, The Netherlands.

Shiffrin, R.M. & Schneider, W. (1977). Controlled and automatic human information processing: II. Perceptual learning, automatic attending, and a general theory. *Psychological Review, 84*, 127-190.

Shingledecker, C.A. & Holding, D.H. (1974). Risk and effort measures of fatique. *Journal of motor behavior, 1*, 17-25.

Sternberg, S. (1969). Memory scanning: Mental processes revealed by reaction time experiments. *American Scientist, 57*, 421-457.

Taatgen, N.A. (1997). A rational analysis of alternating search and reflection strategies in problem solving. *Proceedings of the 19th Annual Conference of the Cognitive Science Society.* Hillsdale, NJ: Erlbaum.

West, R.L. (1996). An application of prefrontal cortex function theory to cognitive aging. *Psychological bulletin, 120*, 272-292.

Wickens, C.D. (1984). Processing resources in attention. In R. Parasuraman & D.R. Davies (eds.) *Varieties of attention.* New York, NY: John Wiley.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Architectures and Tools for Human-Like Agents

Aaron Sloman and **Brian Logan**
School of Computer Science
The University of Birmingham
Birmingham, B15 2TT, UK
+44 121 414 {4775 (Sloman) 3712 (Logan)}
{A.Sloman, B.S.Logan}@cs.bham.ac.uk

ABSTRACT
This paper discusses agent architectures which are describable in terms of the "higher level" mental concepts applicable to human beings, e.g. "believes", "desires", "intends" and "feels". We conjecture that such concepts are grounded in a type of information processing architecture, and not simply in observable behaviour nor in Newell's knowledge-level concepts, nor Dennett's "intentional stance." A strategy for conceptual exploration of architectures in design-space and niche-space is outlined, including an analysis of design trade-offs. The SIM_AGENT toolkit, developed to support such exploration, including hybrid architectures, is described briefly.

Keywords
Architecture, hybrid, mind, emotion, evolution, toolkit.

MENTALISTIC DESCRIPTIONS

The usual motivation for studying architectures is to explain or replicate performance. Another, less common reason, is to account for concepts. This paper discusses "high level" architectures which can provide a systematic non-behavioural conceptual framework for mentality (including emotional states). This provides a new kind of semantics for mentalistic descriptions. We illustrate this using multi-layered architectures based in part on evolutionary considerations. We show briefly how different layers support different sorts of emotion concepts. This complements work by McCarthy(1979, 1995) on descriptive and notational requirements for intelligent robots with self-consciousness.

We provide pointers to an uncommitted software toolkit that supports exploration of hybrid architectures of various sorts, and we illustrate some of the architectural complexity it needs to support.

WHY USE MENTALISTIC LANGUAGE?

We shall need mentalistic descriptions for artificial agents for the same reasons as we need them for biological agents, e.g. (a) because such descriptions will (in some cases) be found irresistible and (b) because no other vocabulary will be as useful for describing, explaining, predicting capabilities and behaviour. ((b) provides part of the explanation for (a).) So, instead of the self-defeating strategy of trying to avoid mentalistic language, we need a disciplined approach to its use, basic mentalistic concepts on information-level architectural concepts.

The "Information level" design stance

Dennett (1978) recommends the "intentional stance" in describing sophisticated robots, as well as human beings. That restricts mentalistic language to descriptions of whole agents, and presupposes that the agents are largely rational. Similarly, Newell (1982) recommends the use of the "knowledge level", which also presupposes rationality. By contrast, we claim that mentality is primarily concerned with an "information level" architecture, close to the requirements specified by software engineers. This extends Dennett's "design stance" by using a level of description between physical levels (including physical design levels) and "holistic" intentional descriptions.

"Information level" design descriptions allow us to refer to various *internal* semantically rich short term and long term information structures and processes. This includes short term sensory buffers, longer term stored associations, generalisations about the environment and the agent, stored information about the local environment, currently active motives, motive generators that can produce motives under various conditions, mechanisms and rules for detecting and resolving conflicts, learnt automatic responses, mechanisms for constructing new plans, previously constructed plans or plan schemata, high level control states which can modulate the behaviour of other mechanisms, and many more.

Some mentalistic concepts refer to the information processing and control functions of the architecture. These functions include having and using information *about* things. E.g. an operating system has and uses information *about* the processes it is running. Here semantic content is present without full-blown intentionality or rationality. Restricting semantic notions to global states of a rational agent, or banning them altogether from explanatory theories, would be as crippling in the study of intelligent agents as it would be in the engineering design of complex control systems. (However, not all semantic states can be fully characterised in terms of *internal* functions, for instance those that refer to *particular* external objects, such as Buckingham Palace, a point beyond the scope of this paper.)

Many of the mechanisms in such an architecture are neither rational nor irrational: even though they acquire information, evaluate it, use it, store it, etc. (Sloman 1994*b*). They are neither rational nor irrational because they are *automatic*. Even a deliberative architecture at some level needs reactive mechanisms to drive the processing. If everything had to be based on prior goals and justifications nothing would ever happen.

ARCHITECTURAL ANALYSIS

Different architectures can correspond to different views

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

of a system, e.g. a physical architecture, composed of the major physical parts, a physiological architecture, corresponding to the major functional roles of physical parts, and an information processing architecture composed of mechanisms involved in acquiring, transforming, storing, transmitting, and using information.

There need not be a one to one correspondence between components in different views. A physical component may be shared between several physiological functions: e.g. the circulatory system is involved in distribution of energy, waste disposal, temperature control, and information transfer.

There is a huge space of possible designs. We make no presumption that information processing mechanisms must all be computational (whatever that means). Nor is there a commitment regarding *forms* used to encode or express information. They may include logical databases, procedures encoding practical know-how, image structures, neural nets or even direct physical representations, as in thermostats and speed governors.

Biological plausibility requires evolvability as well as consistency with experimental data and brain physiology. The capabilities and neural structures of different sorts of animals (e.g. insects, rodents, apes, humans) suggest that different types of architectures evolved at different times, with newer architectures building new sorts of functionality on older ones. We suggest that human mental states and processes depend on interactions between old and new layers in a biologically plausible control architecture producing various kinds of internal and external behaviour, including "internal" processes such as motive generation, attention switching, global redirection in emergencies, problem solving, information storage, skill acquisition, self-evaluation and even modification of the architecture.

Besides the multi-layered central information processing architecture there are sensors and effectors of various kinds. These involve more than just transduction of energy or information into or out of the system. We suggest that both have evolved multiple layers interacting with the different layers in the central system as in Figure 1. Such an architecture can generate a huge variety of concepts relevant to describing its states and processes. It also supports a wide variety of types of learning, yet to be analysed.

Indeterminacy of architecture

Often boundaries between sub-mechanisms and levels of description are unclear, including the boundary between the control architecture and mere physiological infrastructure. In brains, chemical processes provide energy and other resources, along with damage repair and resistance to infections. However, effects of drugs, diseases and genetic defects involving brain chemicals suggest that chemistry forms more than a physiological infrastructure: chemically controlled mood changes may be an important part of an organism's intelligent reaction to changing circumstances, and alcohol can change "no" into "yes"! But we don't know how far chemical reactions play a direct role in information processing or high level control,

In both perception and action the "hardware/software" boundary is blurred. E.g. visual attention can be switched with or without redirection of gaze, and fine-grained manipulation can be shared between software and hardware, e.g. in compliant wrists, which reduce the control problem in pushing a close fitting cylinder into a hole. Simon (1969) pointed out long ago that there can be information sharing between internal and external structures.

It is too early for clear definitions of the boundaries of architectures or their components. However, important ideas are beginning to emerge including contrasts between:
(a) reactive *vs* deliberative functions,
(b) symbolic *vs* neural mechanisms,
(c) logical *vs* other sorts of information manipulation,
(d) continuous *vs* discrete control,
(e) using continuously available environmental information *vs* using information stored in memory,
(f) hierarchical *vs* distributed control,
(g) serial *vs* concurrent processing,
(h) synchronised *vs* asynchronous processing,
(i) genetically determined capabilities, those produced by adaptive mechanisms within individuals, and those absorbed from a culture (e.g. learnt poems and equations).

Instead of viewing these contrasts as specifying *rival* options, we should allow combinations of these alternatives to have roles in multifunctional architectures. Work on hybrid mechanisms (e.g. combinations of neural and symbolic systems) is now commonplace, but in order to explore agents rivalling human or even chimpanzee sophistication we need to understand far more complex combinations of subsystems, including complex sub-architectures *within* perceptual and motor control mechanisms, and a deep integration of cognitive and affective functions and mechanisms (Wright, Sloman & Beaudoin 1996, Sloman 1998(forthcoming)). However, there is no unique "correct" architecture: different designs have different trade-offs, as biological evolution shows. We need to understand the trade-offs and possible trajectories. This includes finding good concepts for describing systems with different designs.

ARCHITECTURES AND EMERGENT CONCEPTS

A deep conceptual framework takes account of the range of possible states and processes supported in an architecture, generating a system of high-level descriptive concepts for describing an organism, software agent, or robot, just as a knowledge of molecular architecture provides a basis for labelling chemical compounds and describing chemical processes.

A control architecture can support a collection of states and processes, often indefinitely large. Concepts derived in this way from the architecture are "deep concepts". "Shallow" concepts, based entirely on observed behavioural patterns bearing no relationship to the architecture, are likely to have reduced predictive and explanatory power, like concepts of physical matter based on visible properties rather than atomic and molecular structure.

Not all states require specific mechanisms in the architecture. A computing system that is "overloaded" does not have an "overloading" mechanism, since overloading results from interaction of many different mechanisms whose functions is not to produce overload. Similarly many mental states, e.g. some debilitating emotions, may *emerge* from interactions within an architecture, rather than from an emotion module.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

If there are several coexisting, interacting sub-architectures (e.g. reactive and deliberative sub-architectures) then higher order concepts are needed to describe the variety of possible relationships between them. For instance, states in one subsystem can modulate processes in others. Such relationships can change over time: sometimes one part is dominant and sometimes the other. Moreover, when training increases fluency in a cognitive skill this may shift responsibility for a task from a general purpose module to a dedicated module.

Familiar prescientific concepts, e.g. "emotion", can be ambiguous if they sometimes refer to processes in a component of the architecture (e.g. being startled, or terrified by a fast approaching menace, may result from a specific module, perhaps part of the limbic system) and sometimes to emergent interactions between subsystems (e.g. guilt and self-reproach).

Unlike emotions which we share with rats, e.g. being startled, which use this old global alarm system, many human emotions involve a partial loss of control of thought processes, (e.g. extreme grief, ecstasy or hysteria). This presupposes the possibility of being in control. That, in turn, depends on the existence of an architecture that supports certain kinds of self monitoring, self evaluation, and self modulation. Being careful or careless requires an architecture able to control which checks are made during planning, deciding and acting.

Which animal architectures can support control of thought processes is not clear. Systems lacking such underpinnings may not be usefully describable as "restrained", "resisting temptation", etc. Can a rat sometimes control and sometimes lose control of its thought processes? Can a rat be careless in its deliberations? Over-simple architectures in software agents will also make such concepts inappropriate to them.

EVOLUTION AND MODULARITY

Our discussion has presupposed that architectures are to some extent intelligible. Will naturally evolved systems be modular and intelligible? In principle, any required finite behaviour could be produced by a genetically determined, unstructured, non-modular architecture, including myriad shallow condition-action rules with very specific conditions and actions providing flexibility. However, as the diversity of contexts grows and the need to cope with unexpected situations, including interactions with other other agents, increases, memory requirements for such a system can grow explosively, and it becomes more difficult find a design which anticipates all the conditions and actions in advance. Thus the time required to evolve all the shallow capabilities is far greater and the required diversity of evolutionary contexts far greater than for a system with planning abilities.

A shallow non-modular system would not only be hard to design, describe and explain: it would be hard to control or modify, whether controlled from outside or controlling itself, whether modified by a designer, or modified by evolution. (Contrast the use of bit-strings in genetic algorithms with the use of trees in genetic programming.)

All this suggests that for complex organisms there would be pressure towards more modular architectures with generic mechanisms that can be combined by a planner to handle new situations, and adaptive architectures that can change themselves to improve performance. Both the normal evolutionary pressures for modularity and reuse, and the need for economy in high level self-control mechanisms could have increased the pressure towards evolution of modular control architectures, in some organisms. So the existence of self-monitoring, self-evaluation and self-control processes could influence the further evolution of control architectures. Apparently insects found a different solution.

It may eventually be possible to investigate this issue in simulated evolution.

THE EMERGENCE OF "QUALIA"

If a system has the ability to monitor its own states and processes, a new variety of descriptions becomes applicable, labelling new forms of self control, including its own discovery of concepts for self-description. The objects of such self-monitoring processes may be virtual machine states as well as internal physical or physiological states.

Many of the spatial, temporal and causal categories used in perceiving the environment have evolved to support biological functions of organisms in those environments, even though precise details can vary widely between species and between individuals in a species. Likewise, it is possible that the basic and most general mentalistic categories that humans use in describing and thinking about themselves and other agents are not reinvented by different individuals (or cultures) but generated by evolutionary processes driving development of self-monitoring capabilities.

Phenomena described by philosophers as "qualia" may be explained in terms of high level control mechanisms with the ability to switch attention from things in the environment to *internal* states and processes, including intermediate sensory datastructures in layered perceptual systems. These introspective mechanisms may explain a child's ability to describe the location and quality of its pain to its mother, or an artist's ability to depict how things look (as opposed to how they are). Software agents able to inform us (or other artificial agents) about their own internal states and processes may need similar architectural underpinnings for qualia.

From this standpoint, the evolution of qualia would not be a single event, but would involve a number of steps as more kinds of internal states and processes became accessible to more and more kinds of self-monitoring processes with different functions, e.g. requesting help from others or discovering useful generalisations about oneself. Such step-wise development may also occur within an individual.

HOW TO MAKE PROGRESS

There are several ways in which we might try to explore the relationship between architecture and mentality. One approach is to push the approach based on "shallow" behaviour-based concepts as far as possible, and analyse where it breaks down, or where patching it is very difficult (e.g. dealing with new unexpected combinations of conditions where applicable rules conflict, or where no rule applies).

Another approach is to attempt a theoretical analysis of the types of situations that will make development increasingly difficult and to produce increasingly general architectures to cope with the difficulties, using any ideas

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

that work, and then conducting experiments to find out where they break down. This approach need not be constrained by theories of how human minds work: there may be alternative architectures capable of producing extremely useful or even "believable" performances. Initially the constraints on this type of theorising will be very ill-defined because of paucity of relevant knowledge and the shallowness of current theories. However, it is likely that as the work progresses more and more constraints can come from advances in other fields, and more and more tests can be generated to help us choose between alternative hypotheses. (Compare the ancient Greek atomic theory with modern atomic theory.)

Yet another approach is to use whatever direct or indirect evidence is available from brain science, experimental psychology, forms of mental disorder, patterns of development in infancy and decay in old age, evolution, folklore, introspection, common observation, or conceptual analysis of everyday mental concepts. Plausible architectures based on such evidence can then be tested by running experimental implementations, or by analysing their consequences and performing empirical research.

Our work is based on the second and third approaches. The architectural ideas in this paper come from a wide range of sources.

ARCHITECTURAL LAYERS

Part of the task is to find increasingly accurate and explicit theories of the types of architecture to be found in various sorts of human minds (and others) to be used as frameworks for generating families of descriptive concepts applicable to different sorts of humans (including infants and people with various kinds of brain damage) and different sorts of animals and artificial agents.

We conjecture that human-like agents with powers of self-control need a type of architecture with at least three distinct classes of mechanisms which evolved at different times (Sloman 1998(forthcoming)):

(1) Very old reactive mechanisms, found in various forms in all animals, including insects — this includes "routine" reactive mechanisms and "global alarm" mechanisms (the limbic system).

(2) More recently evolved deliberative mechanisms, found in varying degrees of sophistication in some other animals (e.g. cats, monkeys);

(3) An even more recent meta-management (reflective) layer providing self-monitoring self-evaluation, and self-control, using in part deliberative mechanisms of type (2), and perhaps found only in humans and other primates (in simpler forms).

Such an architecture is shown schematically (without alarms) in Figure 1 and each of the layers is described in more detail below. Note that the layers occur in perceptual and motor subsystems as well as centrally.

This is one among many possible designs. Some animals or artefacts may have only one or two layers, and different kinds of reactive, deliberative and meta-management mechanisms are possible.

We are not claiming that these mechanisms are alike in all humans. Deliberative capabilities seem very primitive in new born infants, and the third layer may be non-existent at birth. Moreover a culture can influence development of these layers, as can effects of brain damage, disease

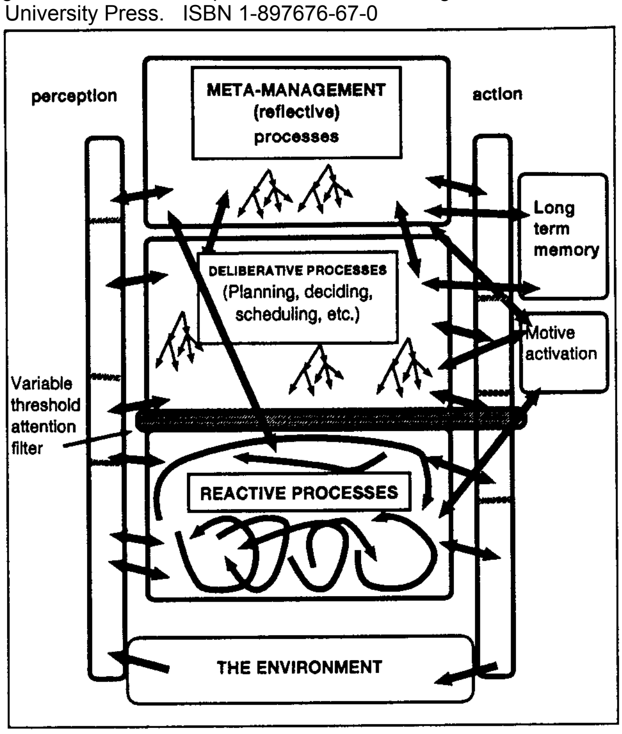


Figure 1: **A three layered agent Architecture**
(Note: global 'alarm' mechanisms not shown.)

or aging. Some architectures may be possible for synthetic agents that are never found in organisms (e.g. solely deliberative architectures, or hybrid systems without global alarms).

Categories and strategies in all layers may be influenced by physical and social environments. A meta-management layer may use both categories and values absorbed from a culture as well as some genetically determined categories and strategies. For instance, certain motives for acting promote negative self-assessment and guilt in some cultures and not in others.

Within an individual, it is also possible for different modes of meta-management to take control in different contexts, e.g. in a family context, in a football game, and in the office. Individual variations might lead, at one extreme to multiple-personality disorder, and at another extreme to excessively rigid personalities.

Concurrent mechanisms

The layers are not assumed to form a rigidly hierarchical control architecture. Rather the three layers operate concurrently, with mutual influences. The reactive mechanisms will perform routine tasks using genetically determined or previously learnt strategies. When they cannot cope, deliberative mechanisms may be invoked, by the explicit generation of goals to be achieved. This can trigger various kinds of deliberative processes including considering whether to adopt the goal, evaluating its importance or urgency, working out how to achieve it, comparing it with other goals, deciding when to achieve it, deciding whether this requires reconsideration of other goals and plans, etc. (See chapter 6 of Sloman (1978).)

At other times the deliberative mechanisms may either attend to long term unfinished business or run in a "free-wheeling" mode, nudged by reactive processes which normally have low priority, including attention-diverting mechanisms in the perceptual subsystems. To allow

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

direct communication with "higher" cognitive functions, perceptual systems may also have layered architectures in which different levels of processing occur in parallel, with a mixture of top-down and bottom-up processing. (Compare seeing a face as a face and as happy.)

If the internal layers operate concurrently, fed in part by sensory mechanisms which are also layered, they may also benefit from a layered architecture in motor systems. For example, reactive mechanisms may directly control some external behaviour, such as running, while the other mechanisms are capable of modulating that behaviour (e.g. changing the speed or style of running, or in extreme cases turning running into dancing). Likewise proprioceptive feedback of different sorts may go to different layers.

Where there is a global alarm system, there may be variations as regards which components provide its inputs and which can be modified by it. In humans connections to and from the limbic system seem to exist everywhere (Goleman 1996).

We now describe in a little more detail the differences between the layers (Figure 1) before discussing their implications for emotions. (The figure is much simplified, to reduce clutter).

Reactive agents

It is possible for an agent to have a purely reactive architecture, where:

- Mechanisms and space are permanently dedicated to specific functions, and can run concurrently, more or less independently, with consequent speed benefits. Some may be digital, some continuous.
- Conflicts may be handled by vector addition, voting, or winner-takes-all nets.
- Some learning is possible: e.g. tunable control loops, change of weights by reinforcement learning. Such learning merely alters links between pre-existing structures and behaviours.
- There is no explicit construction of new plans or structural descriptions or other complex internal objects, and therefore no explicit evaluation of alternative structures.
- Concurrent processing at different abstraction levels can encourage the evolution of different levels of processing in sensory and motor subsystems.
- Some of the reactions to external or internal conditions may be internal, e.g. various kinds of internal feedback control loops.
- If "routine" reactions are too slow a fast "global alarm" system taking control in emergencies may be useful.

As explained above, if all the main possible behaviours need to be built in by evolutionary adaptation or direct programming the space requirements may explode as combinations increase. Likewise the time required to evolve all relevant combinations. A partial solution is to provide "chaining" mechanisms so that simpler behaviours can be re-used in different longer sequences. Simple sub-goaling may achieve this, changing internal conditions that launch behaviours. This may be a precursor to deliberative mechanisms.

It appears that insects have purely reactive architectures, and cannot reflect on possible future actions. Yet the reactive behaviours can produce and maintain amazing construction, e.g. termites' "cathedrals".

There is no form of externally observable behaviour that cannot, in principle, be implemented in a purely reactive system, without any deliberative capabilities, though it seems that in some organisms the evolutionary pressures mentioned above have led towards a different solution — which may coexist with the old one.

Combining reactive and deliberative layers

The ability to construct new complex behaviours as required reduces the amount of genetic information that needs to be transmitted as well as the storage requirements for each individual. It also reduces the number of generations of evolution required to reach a certain range of competence. In a deliberative mechanism:

- Evaluating and comparing options for novel combinations before selecting them requires a new ability to build internal descriptions of internal structures. It also needs a long term associative memory.
- Using re-usable storage space for new plans and other temporary structures, and use of a single associative memory (even if based on neural nets), makes processes inherently serial.
- New behaviours developed by the deliberative system can be transferred to the reactive layer (e.g. learning new fluent skills).
- Sensory and action mechanisms may develop new, more abstract, processing layers, which communicate directly with deliberative mechanisms. This could explain high level sensory experiences (e.g. seeing a face as happy).
- Even if neural nets are used, operation may be resource-limited because learning from consequences becomes explosive if too many things are done in parallel. Limiting concurrent processes may also simplify integrated control.
- Deliberative resource limits may mean that a fast-changing environment can cause too many interrupts and re-directions. Filtering new interrupts via dynamically varying thresholds (see Figure 1) helps but does not solve all problems.
- A global alarm system may include inputs from and outputs to deliberative layers.

The need for self-monitoring (meta-management)

Deliberative mechanisms may be implemented in specialised reactive mechanisms which react to internal structures, and can interpret explicit rules and plans.

However, evolutionarily determined deliberative strategies for planning, problem solving, decision making, evaluating options, can be too rigid. Internal monitoring mechanisms may help to overcome this e.g. by recording deliberative processes and noticing which planning strategies or attention switching strategies work well in which conditions. This could include detecting when one goal is about to interfere with other goals, or noticing that a problem solving process is "stuck", e.g. in a loop, or noticing that a solution to one problem helps with another.

Internal monitoring combined with learning mechanisms may allow discovery of new ways of categorising internal states and processes and better ways of organising deliberation. Meta-management and deliberative mechanisms permit cultural influences via the absorption of new concepts and rules for self-categorisation, evaluation and control.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Attending to intermediate perceptual structures can also allow more effective communication about external objects, e.g. by using viewpoint-centred appearances to help direct attention, or using drawings and paintings to communicate about how things look.

The meta-management layer may share mechanisms with the other two, including the global alarm mechanism (limbic system?) but also needs new mechanisms that can access states and processes in various parts of the whole system, categorise what is going on internally, evaluate it, and in some cases modify it. This can help with proper management of limited deliberative resources.

ARCHITECTURAL LAYERS & EMOTION CONCEPTS

We conjecture that different layers account for different sorts of mental states and processes, including emotional states. Disagreements about the nature of emotions can arise from failure to see how different concepts of emotionality depend on different architectural features, not all shared by all the animals studied.

(1) The old reactive layer, with the global alarm system, produces rapid automatically stimulated emotional states found in many animals (being startled, terrified, sexually excited).

(2) A deliberative layer, in which plans can be created and executed, supports cognitively rich emotional states linked to current desires plans and beliefs (like being anxious, apprehensive, relieved, pleasantly surprised).

(3) Characteristically human emotional states (e.g. humiliation, guilt, infatuation, excited anticipation) can involve reduced ability to focus attention on important tasks because of reactive processes (including alarm processes) interrupting and diverting deliberative mechanisms, sometimes conflicting with meta-management decisions (Wright et al. 1996).

The second class of states depends on abilities possessed by fewer animals than those that have reactive capabilities. The architectural underpinnings for the third class are relatively rare: perhaps only a few primates have them.

Many theories of emotion postulate a system that operates in parallel with normal function and can react to abnormal occurrences by generating some kind of interrupt, like the global alarm mechanism. Consider an insect-like organism with a purely reactive architecture, which processes sensory input and engages in a variety of routine tasks (hunting, feeding, nest building, mating, etc.). It may be useful to detect certain patterns which imply an *urgent* need to react to danger or opportunity by freezing, or fleeing, or attacking, or protecting young, or increasing general alertness. Aspects of the limbic system in vertebrate brains seem to have this sort of function (Goleman 1996).

In architectures combining reactive and deliberative layers, the alarm mechanism can be extended to cause sudden changes also in *internal* behaviour, such as aborting planning or plan execution, switching attention to a new task, generating high priority goals (e.g. to escape, or to check source of a noise). Likewise processing patterns in the deliberative layer may be detected and fed into the alarm system, so that noticing a risk in a planned action can trigger an alarm.

Where a meta-management layer exists, data from it could also feed into the alarm system, and it too could be affected by global alarm signals. One meta-management function could involve learning which alarm signals to ignore or suppress. Another would extend the alarm system to react to new patterns, both internal and external. Another would be development of more effective and more focused (less global) high speed reactions, e.g. replacing a general startle reaction with the reactions of a highly trained tennis player.

This, admittedly still sketchy, architecture, explains how much argumentation about emotions is at cross-purposes, because people unwittingly refer to different sorts of mechanisms which are not mutually exclusive. An architecture-based set of concepts can be made far less ambiguous.

Familiar categories for describing mental states and processes (e.g. believes, desires, perceives, attends, decides, feels, etc.) may not survive unchanged as our knowledge of the underlying architecture deepens, just as our categories of kinds of physical stuff were refined after the development of a new theory of the architecture of matter. Researchers need to be sensitive to the relationships between pre-theoretical and architecture-based concepts as illustrated in (Wright et al. 1996).

THE SIM_AGENT TOOLKIT

We still have much to learn about different agent architectures. The properties of complex systems cannot all be determined by logical and mathematical analysis: there is a need for a great deal more exploration of various types of architectures, both in physical robots and in simulated systems.

Many robot laboratories are doing the former. We work on simulated systems so that we can focus on the issues that are of most interest to us, involving the kind of architecture sketched above including alarm systems, leaving details of sensory devices and motors till later. When simulations are well designed they can sometimes provide cheaper and faster forms of experimentation, though care is always necessary in extrapolating from simulations.

Many toolkits exist to support such exploration, usually based on a particular architecture or class of architectures (e.g. neural net architectures, or SOAR, or PRS). We wished to investigate diverse and increasingly complex architectures, including coexisting reactive and deliberative sub-architectures, along with self-monitoring and self-modifying capabilities, and including layered perceptual and action subsystems. We also wished to explore varying resource-limits imposed on different components of the architecture, so that, for example, we could compare the effects of speeding up or slowing down planning mechanisms relative to the remaining components of an architecture (e.g. in order to investigate various deliberation management strategies, such as "anytime" planning).

To support this exploration we designed and implemented (in the language Pop-11 (Sloman 1996)) the SIM_AGENT toolkit. It is being used at Birmingham for teaching and research, including research on evolutionary experiments, and also at DERA Malvern for designing simulated agents that could be used in training software. An early version of the toolkit developed jointly with Riccardo Poli, was described at ATAL95 (Sloman & Poli 1996). Since then development has continued in response to comments and suggestions from users (Baxter, Hepplewhite, Logan & Sloman. 1998).

The toolkit supports a collection of interacting agents and inanimate objects, where each agent has an internal architecture involving different sorts of coexisting interacting components, including deliberative and reactive components. Not all agents need have the same architecture.

The key idea is that each component within an agent is connected to other components in that agent via a forward-chaining condition-action rulesystem. Each agent's rulesystem is divided into a collection of different rulesets, where each ruleset is concerned with a specific function, e.g. analysing a type of sensory data, interpreting linguistic messages, creating, checking or executing plans, generating motives, etc. Rulesets can be concurrently active, and may be dynamically switched on and off. They may be assigned different resource limits.

Conditions and actions of rules within an agent can refer to databases in that agent. Thus one form of communication between sub-mechanisms is through the databases in the agent. It is possible for an agent to have some global databases accessed by all components of an agent and others which are used only by specific sub-groups. One agent cannot normally inspect another's databases.

An architecture for an agent class is defined by specifying a collection of rulesets and other mechanisms, along with the types of databases, sensor methods, action methods, communication methods and possibly tracing and debugging methods. It is hoped that users will develop re-usable libraries defining different mechanisms and architectures.

The rulesets are implemented in Poprulebase, a flexible and extendable forward-chaining rule-interpreter. Rulesets can be turned on and off dynamically, modelling one aspect of attention shift, and new ones added, modelling some forms of cognitive development. Although the main conditions and actions use patterns matching database components, some conditions and some actions can invoke sub-mechanisms directly implemented in Pop-11, e.g. low level vision or motor-control mechanisms. Other Poplog languages (e.g. Prolog) or external languages (e.g. C, Fortran) can also be invoked in conditions and actions. For example, a rule condition could in principle interrogate physical sensors and a rule action could send signals to motors. Sockets can run sub-systems on other machines, and unix pipes can communicate with processes on the same machine.

To illustrate the power, a Pop-11 rule action can run the rule interpreter recursively on a specialised rule system.

The rule-based formalism is easily extendable, allowing different sorts of condition-action rules to be defined. For example, one of the extensions designed by Riccardo Poli allows a set of conditions matched against a database to provide a set of input values for a neural net, whose output is a boolean vector which can be used to select a subset of actions to be run. A recent extension was a new class of ADD and DELETE actions for automatically maintaining sets of dependency information between database items, so that if an item is deleted then everything recorded as directly or indirectly depending on it, is also deleted. A Pop-11 condition can be used to perform backward chaining if desired.

The interpreter can be run with various control strategies, including the following options for each active ruleset on each cycle: (a) all runnable rules (those with all conditions satisfied) are run, (b) only the first runnable rule found is run, (c) the set of runnable rule instances is sorted and pruned (using a user-defined procedure) before the actions are run.

When the rule interpreter is applied to a ruleset, it can be allowed to run to completion (e.g. until no more rules have all conditions satisfied, or a "STOP" action is executed.) Alternatively it can be run with a cycle limit N, specifying that it should be suspended after N cycles even if there are still rules with satisfied conditions. Another possibility is to set a timer and halt it after a fixed time interval. Either of these mechanisms can be used to impose resource limits on one ruleset relative to others, within an agent.

The design of the toolkit supports multi-agent scenarios, using a time-sliced scheduler which in each time slice allows each agent to run its sensory methods, its internal rulesets, and, in a second pass at the end of the time slice, its *external* action methods.

The object oriented design uses Pop-11's Objectclass system, which supports multiple inheritance and generic functions. This makes it easy for users to extend the ontology by defining new sub-classes, with their own sensing, acting and internal processing methods, without any editing of the core toolkit code. A default class provides a default set of methods, including the sim_run_agent method used to run each the agent's rulesets, along with various tracing methods.

The object oriented approach allows a Pop-11 graphical library to be connected to the toolkit by re-defining tracing and other methods (e.g. move methods) to invoke graphical procedures. The graphical facilities support not only displays of agent actions but also asynchronous user intervention: e.g. using the mouse to move objects in an agent's environment, or turning tracing and profiling mechanisms on or off while the toolkit is running.

Scenarios implemented so far using the toolkit include a simulated robot using a hybrid modular architecture to propel a boat to follow the walls of an irregular room, evolution of a primitive language for cooperation between a blind and an immobile agent, a user controlled sheepdog and sheep to be penned, two purely reactive "teams" of agents able to move past each other and static obstacles to get to their target locations, a simulated nursemaid looking after troublesome infants while performing a construction task, a distributed minder (Davis 1996), one agent tracking another subject to path constraints in 3-D undulating terrain, and, at DERA Malvern, simulated tank commanders and tank drivers engaging in battle scenarios (Baxter 1996). We expect to continue developing the toolkit and building increasingly sophisticated simulations, moving towards the architecture depicted in Figure 1 and subsequently extended in various ways.

In particular we have plans for improving the self modifying and self monitoring capabilities by replacing the rulesystem, currently a list of rulesets and rulefamilies, with database entries. Thus rule actions can then change the processing architecture.

The toolkit is applicable to a wide range of agent development tasks, including simplified software agents which require only a small subset of beliefs, goals, plans, decisions, reactions to unexpected situations, etc. These might be web search agents, or "believable" entertainment agents whose observed behaviour invites

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

mentalistic description whether or not the descriptions are justified by internal mechanisms, states and processes, e.g. the OZ project at CMU (Bates, Loyall & Reilly 1991). The toolkit could also be used to implement teaching and demonstration libraries, e.g. for students in psychology or the helping professions, where students can manipulate the architectures of simplified human-like agents, to gain a deeper understanding of the multiple ways in which things can go wrong.

CONCLUSION

Like software engineers, and unlike Dennett and Newell, we assume semantically competent sub-systems, but not rationality. Using this information-level design stance, we have sketched a framework accommodating multi-disciplinary investigation of many types of architecture of varying degrees of sophistication, with varying mixtures of information-processing capability, based on AI, Alife, Biology, Neuroscience, Psychology, Psychiatry, Anthropology, Linguistics and Philosophy. This framework can extend our understanding of both natural and artificial agents. Above all it generates systems of concepts for characterising various types of mentality. Information-based control architectures provide a new framework for analysing, justifying and extending familiar mentalistic concepts.

There is no uniquely "right" architecture. Types of architectures that are relevant, and dimensions of possible variation, are not yet well understood. More exploration and analysis is required, replacing premature (sometimes confrontational) commitment to particular mechanisms and strategies. We need to understand the structure of design space and niche space, and trajectories that are possible within those spaces (Sloman 1994*a*, Sloman 1994*b*, Sloman 1998(forthcoming)). This requires collaborative philosophical analysis, psychological and neurophysiological research, experiments with diverse working models of agents, and evolutionary investigations. Some of this exploration can be based in part on powerful new software tools.

Such work is likely to throw up types of architectures that we would not otherwise think of, which will force us to invent new concepts for describing synthetic minds which are not like our own, and help us understand our own by contrast.

ACKNOWLEDGEMENTS & NOTES

We acknowledge support from the UK Joint Council Initiative, The Renaissance Trust, and DERA Malvern, and much help from students and staff at Birmingham, especially Luc Beaudoin and Ian Wright.

Pointers to code and documentation for our toolkit are at http://www.cs.bham.ac.uk/~axs/cog_affect/sim_agent.html

Several papers developing these ideas are in the Cognition and Affect Project ftp directory: ftp://ftp.cs.bham.ac.uk/pub/groups/cog_affect

References

Bates, J., Loyall, A. B. & Reilly, W. S. (1991), Broad agents, *in* 'Paper presented at AAAI spring symposium on integrated intelligent architectures'. (Available in SIGART BULLETIN, 2(4), Aug. 1991, pp. 38–40).

Baxter, J., Hepplewhite, R., Logan, B. & Sloman., A. (1998), Sim_agent two years on, Technical Report CSRP-98-2, University of Birmingham, School of Computer Science.

Baxter, J. W. (1996), Executing plans in a land battlefield simulation, *in* 'Proceedings of the AAAI Fall symposium on Plan execution: Problems and issues November 1996', AAAI, pp. 15–18.

Davis, D. N. (1996), Reactive and motivational agents: Towards a collective minder, *in* J. Muller, M. Wooldridge & N. Jennings, eds, 'Intelligent Agents III — Proceedings of the Third International Workshop on Agent Theories, Architectures, and Languages', Springer-Verlag.

Dennett, D. C. (1978), *Brainstorms: Philosophical Essays on Mind and Psychology*, MIT Press, Cambridge, MA.

Goleman, D. (1996), *Emotional Intelligence: Why It Can Matter More than IQ*, Bloomsbury Publishing, London.

McCarthy, J. (1979), Ascribing mental qualities to machines, *in* M. Ringle, ed., 'Philosophical Perspectives in Artificial Intelligence', Humanities Press, Atlantic Highlands, NJ, pp. 161–195. (Also accessible at http://www-formal.stanford.edu/jmc/ascribing/ascribing.html).

McCarthy, J. (1995), Making robots conscious of their mental states, *in* 'AAAI Spring Symposium on Representing Mental States and Mechanisms'. Accessible via http://www-formal.stanford.edu/jmc/consciousness.html.

Newell, A. (1982), 'The knowledge level', *Artificial Intelligence* **18**(1), 87–127.

Simon, H. A. (1969), *The Sciences of the Artificial*, MIT Press, Cambridge, Mass. (Second edition 1981).

Sloman, A. (1978), *The Computer Revolution in Philosophy: Philosophy, Science and Models of Mind*, Harvester Press (and Humanities Press), Hassocks, Sussex.

Sloman, A. (1994*a*), Explorations in design space, *in* 'Proceedings 11th European Conference on AI', Amsterdam.

Sloman, A. (1994*b*), 'Semantics in an intelligent control system', *Philosophical Transactions of the Royal Society: Physical Sciences and Engineering* **349**(1689), 43–58.

Sloman, A. (1996), *Primer of Pop-11*, School of Computer Science, University of Birmingham, School of Computer Science, The University of Birmingham, Birmingham, B15 2TT, UK. (Last revised October 1997. Ftp version available in **ftp://ftp.cs.bham.ac.uk/pub/dist/poplog/**).

Sloman, A. (1998(forthcoming)), What sort of architecture is required for a human-like agent?, *in* M. Wooldridge & A. Rao, eds, 'Foundations of Rational Agency', Kluwer Academic.

Sloman, A. & Poli, R. (1996), Sim_agent: A toolkit for exploring agent designs, *in* M. Wooldridge, J. Mueller & M. Tambe, eds, 'Intelligent Agents Vol II (ATAL-95)', Springer-Verlag, pp. 392–407.

Wright, I., Sloman, A. & Beaudoin, L. (1996), 'Towards a design-based analysis of emotional episodes', *Philosophy Psychiatry and Psychology* 3(2), 101–126.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

PSI: A Theory of the Integration of Cognition, Emotion and Motivation

Christina Bartl & Dietrich Dörner

Lehrstuhl Psychologie II
Otto-Friedrich - Universität
D-96045 Bamberg

Phone: +49/951/863 1861

e-mail: christina.bartl @ppp.uni-bamberg.de
dietrich.doerner@ppp.uni-bamberg.de

ABSTRACT

In this article we describe a theory aiming at the integration of cognitive processes, emotion and motivation. The theory describes the informational structure of an intelligent, motivated, emotional agent which is able to survive in arbitrary domains of reality. This agent is „energized“ by six motives (needs for energy, water, pain-avoidance, affiliation, certainty and competence). The cognitive processes of this agent are modulated by emotional states and processes. By comparing the behaviour of Psi with human behaviour in a complex computer scenario, the model was tested against reality. Subjects were asked to regulate a dynamic system structural identical to the environment of the autonomous agent. First results show striking similarities between artificial and human behaviour as well as differences.

Keywords

Artificial Life, Cognition, Emotion, Motivation, Action Regulation.

INTRODUCTION

In cognitive science there is a focus on cognition when considering action regulation. Emotional and motivational processes, however, play a considerable role in human behaviour triggering cognitive processes. In a state of anger thinking and reasoning differs from processes under „normal“ conditions. Different emotional states even influence perception in a specific manner. — In a long lasting process of action regulation, when humans have to tackle difficult problems, neither emotions nor motives remain constant. Foreseeing that an important problem cannot be solved an individual will feel helpless and this feeling of helplessness will trigger other feelings and can change the current motive. The motive to find a solution for an intellectual task will be replaced by a motive to demonstrate „competence“ as the inability to solve the problem threatens the self-confidence of the individual.

THE PSI THEORY OF ACTION-REGULATION

A single theory of cognitive processes does not succeed in explaining human behaviour. Furthermore it is necessary to include assumptions about the dynamics of emotions and motivations. During the last years we developed a theory – the Psi theory – concerning the interaction of cognitive, emotional and motivational processes. A computer program was constructed to simulate the theoretical assumptions (see Dörner & Hille, 1995; Hille, 1997; Schaub, 1997). The Psi theory is completely formulated in terms of the theory of neuronal networks, but going into details about the inner structure would exeed the aim of this paper

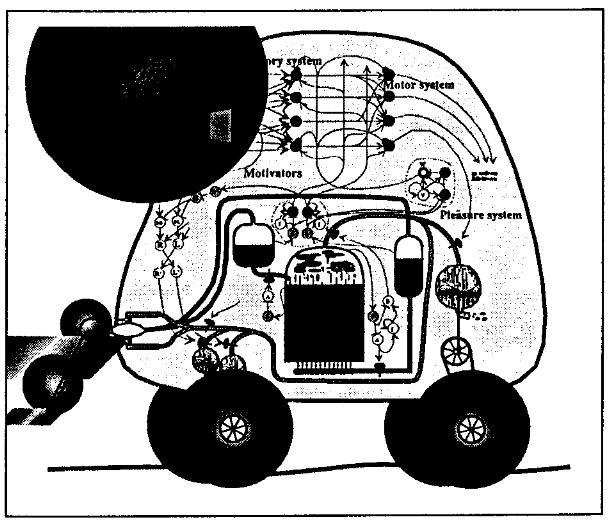


Fig. 1: **Psi as an „autonomous steam engine“.**

The Psi theory includes more than assumptions about single cognitive processes. It aims at a description of

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

the interaction of different cognitive and non-cognitive processes. It is a theory in the tradition of „artificial life“ - research (Steels, 1993). It exists a computer program simulating the theory. The actual version of this computer program is available in internet on page http://141.13.70.49. Fig. 1 shows a possible „materialization“ of Psi as an autonomous steem engine which should care for its existential needs (water and energy). The architecture of the model will be explained below.

Motivation

Fig. 2 shows a rough sketch of Psi's internal structure. At the bottom of fig. 2 the motivational system of Psi is symbolized by a number of „watertanks“. These tanks are mechanical models of „motivators“. „Motivator“ means a system which is sensible for the level of a variable. This should be kept within certain borders (within a setpoint region) by the system. Such variables could be water or energy resources of a system, temperature of a body or any other variable important for life or welfare of a system. When a variable deviates from its set point, a motivator becomes active. In this case there is a need and the motivator will try to launch activities to restore the set point value of the respective variable.

Which motivators are necessary? First of all Psi has to care for its existence. This means that Psi needs (for instance) water and energy. And Psi should preserve its structure; it should avoid pain. Additionally to these „existential“ needs Psi has „informational“ needs, namely a need for certainty, a need for competence and a need for affiliation.

The need for certainty is satisfied by „certainty signals“. An important certainty signal is for example a correct prediction. Acting in a certain domain of reality Psi will learn regularities of its environment. Therefore it will be able to predict the outcomes of its actions and progress of events. If these predictions are correct they will be certainty signals and will fill the „certainty tank“. If the predictions are wrong or if the chain of events does not develop in the predicted way, however this means uncertainty and will decrease the level of the „certainty tank“.

The need for competence is a need for „competence signals“. Each satisfaction of a need, for instance the satisfaction of the need for water, is a signal of competence for Psi. Satisfaction of a need signifies that Psi is able to care for itself. On the other hand a longer lasting period of non-satisfaction signifies inability and therefore is an incompetence signal which empties the competence tank.

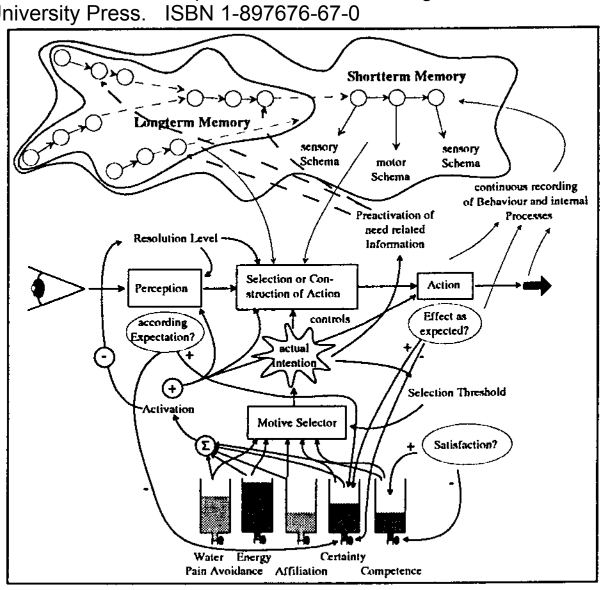


Fig. 2: **The internal structure of Psi. SeeText.**

An empty competence- and an empty certainty-tank launch specific activities. The need for certainty for instance can activate exploration or – depending on the competence (level in the competence tank) – flight. A low level of competence (it shouldn't be too low) will activate „adventure-seeking“, looking for problems the solution of which proves ones own competence.

Group integration is symbolized by the level of the „affiliation tank“. This tank will be filled up by „signals of legitimacy“ (Boulding, 1978) as for instance a smile or a clap on the shoulder. Reports of disapproval serve as signals for nonaffiliation and will empty the „affiliation tank“. – The needs for certainty and for competence are very important for the emotional regulations of Psis behaviour.

Psi's architecture of motivation allows several needs to be active at the same moment. It is therefore vitally important to equip Psi with a selection device, the Motive Selector of fig. 1. This selection device has to select one of the active motives for execution. The motive selected will become the actual intention. An *intention* is a data structure consisting of informations about the goal, about the present state and normally of more or less complete plans for achieving the goal.

The selection device works according to an expectancy - value principle; i.e. it selects the motive with the largest expectancy of success and the largest underlying need. (We call the product of expectancy of effect and amount of the underlying need the **strength** of a motive. So the selection device looks for the motive with the greatest strength.)

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Action regulation, memory and cognitive processes

After an intention has been formed, Psi will „run the intention" to achieve the respective goal. „Running the intention" can mean different processes. When Psi has a lot of experience with the respective domain of reality its memory will often provide a complete course of action as a chain of operations or locomotions leading from the actual situation to the goal. If this however fails an inbuilt planning procedure will try to construct a course of actions by putting together single pieces of knowledge about operators and event chains. (At the moment this planning procedure is a forward-planning, hillclimbing procedure.)

If planning is impossible due to a lack of information or if planning proves to be not successfull, Psi will use trial-and-error procedures to collect information about its respective environment. Generally Psi organizes its activities according to the Rasmussen - system (Rasmussen, 1983). If possible first of all it tries its highly automatized skills, then it changes to „knowledge-based" behaviour and the „ultima ratio" are the trial-and-error procedures.

Psi learns by experience, learns the effects of operators in a specific domain of reality, learns goals and learns chains of events and therefore is able to predict what will happen in the future. But additionally we installed forgetting in the memory of Psi. Forgetting simply is a decay process which continuously diminuishes the strengths of the memory traces. Traces which are rather strong lose less of their strength in time than weak traces which will be destroyed rather quickly. Forgetting has a important function for Psi's cognitive processes. „Punching holes" into sensory and motor schemata of Psi's memory makes them „abstract", „hollow", so that the schemata do not represent concrete images any more, but equivalence classes.

The memory system of Psi is extremely simple and (therefore) powerful. All perceptions and activities are continously recorded. This record is a kind of log of the changing environment, Psis activities and the current intentions. The memory chains representing the immediate past are very dense. Due to forgetting however, memory will consist of single episodes and activities. Memory traces combined with need satisfaction or generation (for instance pain) will be rather strong. Others are weaker and therefore more exposed to decay. Psi has a short term memory which is simply the „head" of the record. This short term memory without any rupture continues into an episodic memory. Remnants of this eventually form the long term memory. If parts of the longterm memory are reused (in planning for instance), the strength of the respective memory trace is enhanced.

Emotions

The information processing of Psi is „modulated". This means that all cognitive processes of Psi are „shaped" according to certain conditions. Such conditions are for instance the strength of the actual intention, the overall amount of all the different needs, the amount of competence and others. These conditions set specific „modulators". One of these modulators is „activation" which depends on the strengths of the needs (roughly spoken the amount of activation mirrors, the sum of the strengths of the needs). Activation triggers some other modulators, for instance „resolution level" and „selection threshold". Resolution level (RL) is the degree of exactness of comparisons between sensory schemata. As most of the cognitive processes of Psi comprise comparisons between schemata this modulator is very important. Comparisons take a long time at a high level level of resolution, but they will be reliable. Under high pressure (when activation is high) the resolution level is low, comparisons don't need a long time, but the risk of „overinclusiveness" is high. A low level of exactness will automatically produce the tendency to consider unequal objects and situations as equal. (This is due to certain mathematical reasons which will not be considered here.) Quick planning processes and a high readiness for action will be the result of a low resolution level, but the plans will be rather risky.

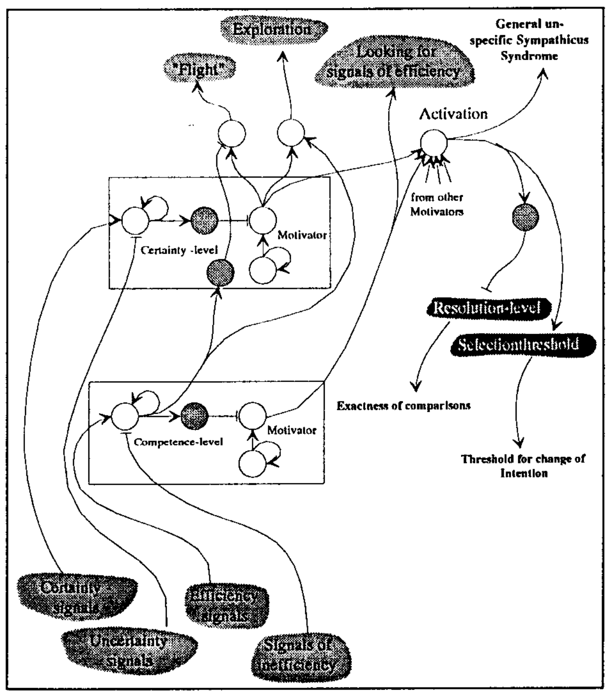


Fig. 3: **Emotional modulations. See text.**

Selection Threshold (ST) could also be called „level of concentration". ST is the strength of the defence of the actual intention against competitors, against other intentions having the tendency to take over the command. The strength of the different motives is not at all

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

constant in the life of Psi, but changes continuously. Because of consumption the needs for energy and for water continuously increase. But a motive can gain strength by external factors too. If for instance Psi notices in a certain situation that it is easily possible to get water, a tendency to shift to the water-intention will result as now the expectancy value for the water – motive increased. Or if an unexpected event will occur the „need for certainty" might increase and Psi will exhibit the tendency to explore the (uncertain) environment or will have the tendency to run away and to hide. Or if for instance planning proves to be unsuccessfull, Psi's „self-confidence" (level of competence) is endangered and Psi will exhibit the tendency to „try its strength", to prove its competence to itself, for instance by looking for a task which is difficult enough that mastery proves competence, but not so difficult that the risk of failure is high.

If ST is high „behavioural oscillations", i.e. a rapid change between different intentions will be hindered to a certain degree (Atkinson & Birch, 1970). A high ST prevents Psi on the other hand from using unexpectedly arising opportunities or from reacting to unexpected dangers. Is ST high, the field of Psi's perception will narrow down.

Fig. 3 gives a general impression of the emotional regulations of Psi. We describe these regulations in terms of neuronal networks (as it is realized in Psi). White circles represent activating neurons, whereas gray circles represent inhibiting neurons. The competence and the certainty - level are now represented as the activation state of neurons. Certainty signals enhance the activity of the „certainty-neuron", whereas uncertainty - signals diminuish this activity. – Satisfaction of a need serves as competence signal and enhances the activity of the „competence-neuron", whereas non-satisfaction decreases this activity. When the uncertainty level is low (high uncertainty) a tendency for flight or aggressive activities will be observable, depending on the competence level. With a high level of competence Psi will exhibit a tendency for aggression in uncertain situations, whereas with a low level of competence it will exhibit flight tendencies.

Activation triggers the „general unspecific sympathicus syndrome"; i.e. high vigilance and a high degree of readiness to react. Additionally it triggers RL and ST, which modulate cognitive processes, perception, planning activities, memory search. It is obvious that Psis emotions are the result of a rather complex interaction of motivational and cognitive processes together with the modulation of RL and ST.

These modulators (RL and ST) together with the need for certainty and the need for competence produce a lot of „emotional" forms of behaviour. Psi exhibits fear (expectation of an uneasy event), anxiety („need for certainty"), anger (when unexpectedly Psi is hindered to reach a goal), surprise (unexpected event). This theory of modulations together with the specific motivational structure of Psi constitute a „subaffective" theory of emotion. A theory, which defines emotions in non-emotional terms. To be able to monitor Psis emotions we gave a human face to Psi which alters according to Psis emotional states. Fig. 4 shows some of the facial expression of Psi in different situations.

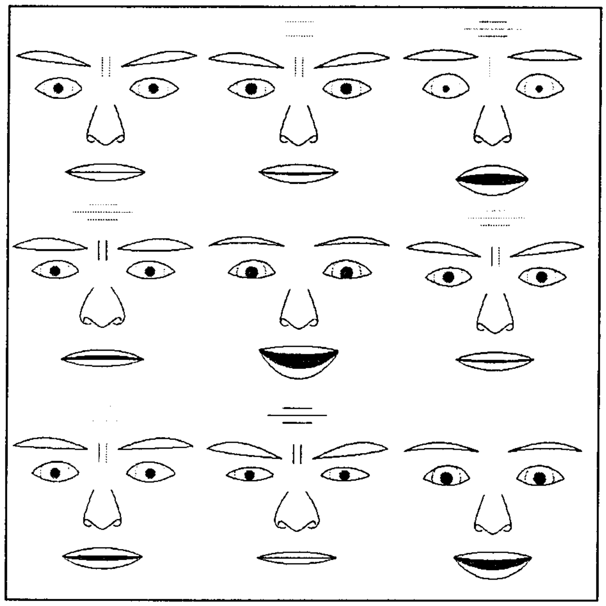


Fig. 4: **Psi's emotions. See text.**

In the upper left corner a resolute Psi can be observed. Psi has a goal and is willing to achieve it against all obstacles. In the upper right corner Psi is seized with horror, helplessly anticipating uneasy events. The middle one face shows Psi in a state of pure joy. The face in the bottom line right shows a joyfull Psi too. You will notice, however, a slight surprise-emotion in this face comparing it with the middle one face. The middle one face in the bottom line shows pain, whereas the face on the right side in the medium line exhibits a state of caution and hesitation. – All these emotions are observable not only in Psis facial expressions, but in its behaviour too[1].

Fig.4 shows what will happen, if you put Psi to a new environment. First the feeling of competence and the feeling of certainty decrease, as Psi is not able to predict what will happen and is not able to care for ist existential needs. But after some learning the respective schemata for appropriate behaviour will be established and Psi is able to cope with its „world".

[1] The procedure for the facial expressions was programmed by Jürgen Gerdes.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

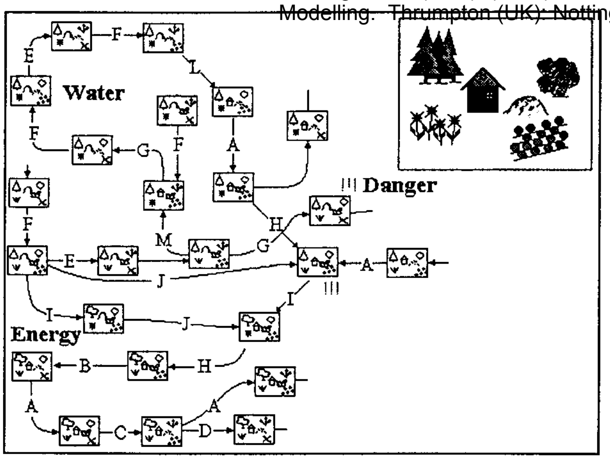


Fig. 4: **An example of the „world“ of Psi and a single „situation“.**

This „world“ is a maze-like environment composed of single „situations“. Fig. 4 shows an example of such a „world“. Psi has to learn how to move from one situation to an other one to arrive at „water“ or „energy“ - situations to satisfy its basic needs. Additionally Psi should learn to avoid dangerous situations. The „situations“ are composed of elements like houses, trees, bushes etc. In the upper right corner of fig. 4 an example of a „situation“ is visible. „To behave“ in such an environment means to manipulate the respective parts of a given situation or to move from one situation to the other one by applying the appropriate operators.

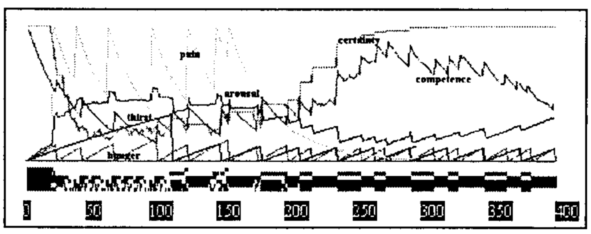


Fig. 5: **Psi's „fate“ in a new environment.**

In fig. 5 some of the internal parameters of Psi when exposed to a new environment are visible. You may observe that first Psi cannot avoid painfull situations and is not able to care for its existential needs („thirst“ for instance increases from cycle 1 to cycle 100 continually as Psi is not able to find water within this time period). But after some learning Psi becomes able to avoid painfull situations and has acquired the capabilities to care for itself.

A COMPARISON BETWEEN HUMAN BEHAVIOUR ANDTHE BEHAVIOUR OF PSI IN THE BIOLAB-GAME

The capability of understanding, explicating and predicting empirical phenomena might help to estimate the value of a theory. The study presented is examining whether the Psi-model succeeds in replicating human behaviour in a complex task.

For that aim we used the scenario BioLab to compare the behaviour of Psi with the behaviour of experimental subjects. We were interested in the similarities and differences between „artificial“ and human behaviour. Differences would possibly point out that basical assumptions of the theory have to be revised. Furthermore the comparison helps to detect the limits of the model explaining human behaviour.

In summary the behavioural test has two objectives: first the results may contribute to the evaluation of the Psi model and the underlying theoretical assumptions. Second the results can give hints to the improvement and the completition of the model of action regulation. By confronting the model with reality necessary modifications and elaborations might be detected.

The scenario BioLab

In the „Biological Laboratory for sugar-based Energy Production“ („BioLab“ factory) subjects are asked to produce certain types of molasses to generate electricity or heat. To modify the molecular structure of the molasses they can use different kinds of catalysts. Under certain conditions, however, the adding of catalysts may cause contaminations. As a result a cleaning of the reactors is necessary. Neither electricity nor heat can be produced until this work has finished. Therefore it is useful to avoid such situations.

The BioLab-system corresponds a maze formally. Subjects can move from one situation to another by using catalysts as operators. They change the structure of the molasses respectively to their actual position in the maze. The amount of operators consists of ten catalysts, some of them needing specific conditions to work. The situations consist of a combination of six dimensions each of them having two valences: either zero or one. This will lead to 64 different situations each represented by a specific combination of these digits.

It is possible to divide the structure of the maze into eight circles, each of them having the valences of the first three dimensions in common. As the eight situations within the circles are highly combined with each other, it is rather simple to move from one situation to another (see fig. 6). In order to leave a circle, it is however essential to have one specific combination of the dimensions four to six. Only this specific situation allows changing between the circles.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

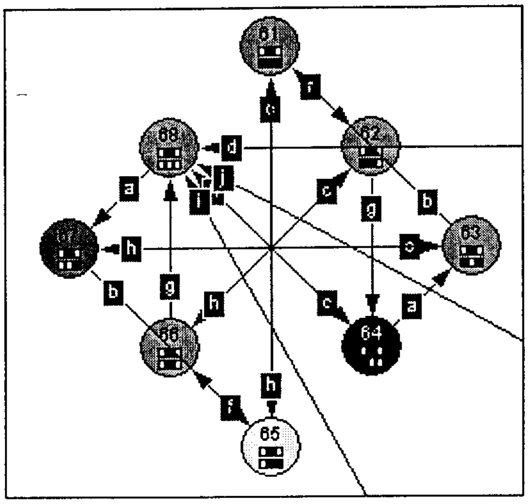


Fig. 6: **The structure of the maze consisting of eight circles, built up by eight situations (figure shows one of eight sections).**

The subjects do not know the formal structure of the maze. They have to explore the BioLab. The situations are visualized by pictures showing the molecular structure of the molasses on the screen. The situation is shown by the characteristics of the molasses in two tanks: they vary with respect to amount, colour and bubbles (see fig. 7).

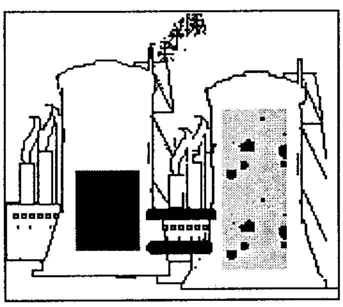


Fig. 7: **The situations of BioLab represented by the different structure of molasses in two tanks.**

To produce energy, it's inevitable to find a way from electricity to heat production and vice versa. Their need for energy is represented by two bars: one showing the actual need for electricity and the other one showing the system's need for heat. The urgency of producing electricity and heat is symbolized by the length of the bar. For example when a subject reaches a situation which provides electricity, the bar will be filled up, no matter how empty it was before. Until the reload is going to happen the electricity resources will be decremented over time.

Electricity as well as heat can be produced in each of the eight circles of the maze. To gain energy a specific combination of the dimensions four to six is essential. As a consequence of getting to situations of satisfaction several times, they will be exhausted. Therefore it is important to find alternatives and to adapt the behaviour to environmental changes.

In summary, handling the BioLab requires capacities of complex problem-solving. Subjects have to explore and regulate a dynamic system with two appetetive and one aversive aims. While they are working on the BioLab game they are coping with a problem identical to the environment of the autonomous agent Psi. Now let's have a look how efficient the laboratory is conducted and how the subjects in contrast to the Psi-model learn to use the catalysts in an effective manner.

The comparison of human and artificial behaviour: efficacy of need satisfaction and of catalysts use

The results presented rely on an experiment conducted with 12 subjects each of them playing the BioLab game for one hour. Each of the subjects had to play under two experimental conditions: first they had to think aloud, second they had to keep tacit. After half an hour of playing the experimental condition changed. Variing the sequence of the two instructions, the subjects were randomly divided into two groups. Most of the subjects were students of psychology from the University of Bamberg.

In general the task was neither too easy nor too difficult for the subjects. All of them succeeded in finding situations where energy production is possible, at least by chance. One subject succeeded in exploring the whole structure of the maze. He/she could intentionally change from one circle to another and has found a efficient way to move from electricity to heat production within the circles.

For a useful comparison between the behaviour of Psi with the behaviour of the subjects we had to parallelize parameters of environment as well as of action time. Whereas the subjects carried out about six actions per minute, Psi conducted more than sixty at the same time. For this reason only the first 360 actions of the model's behaviour protocol were evaluated.

Let us have a look upon the efficiency of managing the BioLab problem: One value representing the performance is the score achieved at the end of the run. Starting with zero, the account increases with a hundred points whenever electricity or heat is produced. Whenever the lab is contaminated, the account decreases by fifty points. Every thirty minutes the account is lowered by one point and finally every use of a catalyst costs one point either.

These statistical results illustrate that human subjects are capable of managing the lab rather good. The mean account is 1314 points after 60 minutes. The variance between the subjects, however, is huge. The subject with the best perfomance gained 2108 points, whereas the worst performance achieved 217 points. The effi-

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

ciency of the model run is even lower: Psi could only manage to get 120 points in the game. The rather bad performance does not rely on a greater number of contaminations (see tab. 1). Moreover the results of the Psi model show a less effective use of catalysts and therefore a lower rate of needs satisfaction.

	Subjects			Psi
	mean	minimum	maximum	value
account of points	1314.58	217	2108	120
number of contaminations	10	2	16	8

Tab. 1: **Statistic values representing the effeciency of needs satisfaction.**

One value representing the successful use of the operators is the percentage of effective catalyses. Psi used as much catalysts as the average subjects. In contrast to the subjects, however, only 15% out of these caused the molasses to change its characteristics.

The following figure shows a boxplot about the results of the subjects and Psi. The subjects were subdivided in two groups: one of them starting with the instruction „thinking aloud", the other one tacit. The bar in the box indicates the median, within the box there are 50% of the subjects represented. The „whiskers" of the box mark the 25th and the 75th percentile of the distribution. Remarkably the performance of the Psi model would be placed within the area marked by the whiskers in the tacit group. Compared to the subjects thinking aloud Psi's performance is significantly low. Its performance is contrasted by the subject „Ellobo" who achieved the best efficacy of the whole sample.

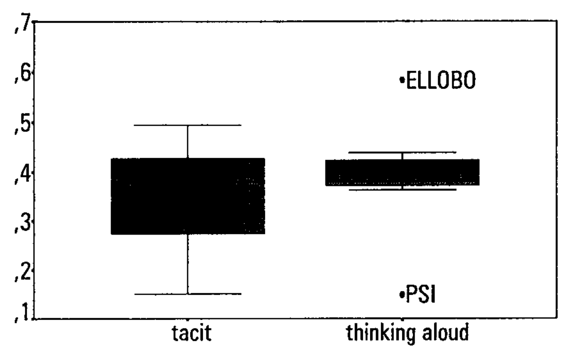


Fig. 3: **The percentage of effective catalyst use between PSI and the subjects. See text.**

First results of single-case studies

Comparing human and artificial behaviour with respect to statistical values will not be sufficient to evaluate a model. Furthermore we tried to replicate the behaviour of each individual by varing the starting parameters of the simulation. By this we created different personalities.

As long as emotional reactions and their impact on information processing are concerned, first results reveal similarities between the model's and the subjects' behaviour.

According to the assumptions of the Psi model subjects show a specific way of action organization: at the beginning they mainly apply a strategy which can be described as „trial-and-error". In the following stage, catalysts are used with respect to success or failure in the past. As a consequence catalysts leading to need satisfaction will be used more frequently in the future, whereas catalysts leading to neutral situations or without any effect will be taken less frequently. Finally catalysts producing contamination will be used more carefully.

As soon as environmental conditions are explored sufficiently, the subjects as well as Psi start making plans. Single action sequences are combined to chains. After gaining a high competence in managing the lab, people as well as our artificial system have an amount of automatisms available. The Rasmussen-system (1983) can be discovered in both: human and artificial behaviour.

Remarkably when trying to replicate the behaviour of single subjects we suceeded in modelling subjects with a rather poor performance, p.e. a quite anxious person producing contamination by the first action he/she made. As a result the subject avoided the catalyst for more than half an hour and as a consequence was not able to produce electricity.

In contrast to more successful subjects the PSI-simulation lacks the capability to reflect on its own behaviour. For this reason strategic flexibility and analogies (i.e. the adoption of learned behavioural sequences on similar situations) can not be found in the simulation runs of out artificial system but in human behaviour.

CONCLUSION

Exploring the similarities and differences of the behaviour of Psi and human behaviour, we found remarkably parallels between the behaviour of Psi and the behaviour of humans. Similar situations provide difficulities for both: humans and Psi. Moreover in comparable situations the model's emotional expression resembles to the expression of the subjects.

There are striking differences as well as similarities. For instance though the planning procedure of Psi is sometimes rather close to what is observable in human behaviour, shows striking differences to human thinking.

Mainly self-reflection is missing. Humans more or less frequently change their thinking and planning procedures by considering the records of their own thinking, analyzing the structure of these records and altering it. Psi is not able to do this. We believe that this is due to the fact that Psi is not able to speak. This „inner dialogue“ is one important aspect of higher cognitive functioning in humans. Therefore Psi should be provided with natural language too in order to get the ability of an inner dialogue.

References

Atkinson, J. W. & Birch, D. (1978) (2nd ed.). Introduction to Motivation. New York: Van Nostrand.

Dörner, D. (1994). Eine Systemtheorie der Motivation. (A system theory of motivation). In: Kuhl, J. & Heckhausen, H. (1996): Enzyklopädie der Psychologie – Motivation, Volition und Handlung. Göttingen: Hogrefe, pp. 329 - 357.

Dörner, D. (1997). Motivation in Artificial and Natural Systems. In: Hara, F. & Yoshida, K. (Eds.), Proceedings of International Symposium on SYSTEM LIFE. Tokyo: The Japan Society of Mechanical Engineers & Inoue Foundation for Science, pp. 17 - 22.

Dörner, D. & Hille, K. (1995). Artificial Souls: Motivated Emotional Robots. In: IEEE Conference Proceedings, International Conference on Systems Man, and Cybernetics; Intelligent Systems for the 21st Century. Vancouver, Volume 4 to 5, pp. 3828 - 3832.

Hille, K. (1997). Die künstliche Seele. Analyse einer Theorie. (An Artificial Soul. Analysis of a theory). Wiesbaden, Germany: Deutscher Universitätsverlag.

Schaub, H. (1997). Modelling Action Regulation. Poznan Studies in the Philosophies of the Science and the Humanities, Vol. 56, pp. 97 - 136.

Boulding, K.E. (1978). Ecodynamics. Beverly Hills: Sage.

Steels, L. (1993). The Biology and Technology of Intelligent Autonomous Agents. Berlin: Springer.

Rasmussen, J. (1983). Skills, Rules, Knowledge: Signals, Signs and Symbols and Other Distinctions in Human Performance Models. IEEE - Transactions, Systems, Man, Cybernetics, SMC 13, S. 257 - 267.