

Sailing to the Model's Edge: Testing the Limits of Parameter Space and Scaling

Amy Santamaria

Walter Warwick

Alion Science and Technology

MA&D Operation

4949 Pearl East Circle, Suite 200

Boulder, CO 80301

303-442-6947

asantamaria@alionscience.com, wwarwick@alionscience.com

Keywords:

Naturalistic Decision Making, Human Performance Modeling

ABSTRACT: *Using the MS/RPD integrated modeling approach, we have modeled a variety of tasks. We typically try to capture aspects of human performance and evaluate the qualitative and quantitative fit of model behavior to human data. A collection of individual models and demonstrations of fit to human data constitute an important validation of a modeling approach. However, there are problems with focusing solely on the “good fit” and “typical model” section of model complexity and parameter space. In this paper, we argue that as modelers, we need to examine our approaches in a broader context, going beyond the comfort zone of good fit and typical models. Using a very simple “generic” model, we examined a relatively small search space, with the goal of better covering and understanding a wider range of complexity and parameter values than our typical models utilize. We investigated scaling by systematically increasing the number of cues and COAs, and we investigated a range of values for three key model parameters. We learned something about limits of scaling. In our parameter exploration, the results underscored the importance of exploring the full range of possible values because parameter values did not always affect performance and learning in a monotonic way.*

1. Introduction

Over the past ten years, we have constructed and presented models of a variety of tasks using our MS/RPD approach (Warwick, McIlwaine, Hutton, & McDermott, 2001; Warwick & Hutchins, 2004; Warwick & Fleetwood, 2006; Warwick & Santamaria, 2006; Santamaria & Warwick, 2007; 2008). Our approach combines Micro Saint task network modeling (the MS component) with underlying learning and memory mechanisms that capture key aspects of recognition-primed decision making (the RPD component) in an integrated architecture. The MS component breaks down tasks into their constituent processes, creating a kind of “dynamic flowchart,” represented as a network of tasks. The RPD component uses a multiple-trace model of long-term memory, a similarity-based recall mechanism, and simple reinforcement-based learning to set values or determine the flow of control in the task network. Using this integrated modeling approach, we typically we focus on a single task, constructing a model, trying to capture aspects of human performance, and evaluating the qualitative or quantitative fit of model behavior to the human data.

A collection of individual models and demonstrations of fit to human data constitute an important validation of a modeling approach. However, there are bigger issues to take into consideration when developing, exploring, and evaluating a modeling framework. There are problems with goodness of fit as the sole criterion (see Roberts & Pashler, 2000, Collyer, 1985). But more critically, there are problems with focusing solely on the “good fit” and “typical model” section of model size and parameter space.

Several important points related to issues of scaling are brought out in Gluck et al. (2007). The authors describe three levels of theory that are implemented in models of cognition: architecture and control mechanisms (Type 1), internal component/module implementation (Type 2) and knowledge (Type 3). Gluck et al. point out that the parameter space for each of those levels is very large and that a typical modeling effort only selects a single point at the intersection of these spaces. From their paper:

A thorough search of even a modest portion of the total possible theoretical state space will require an unprecedented amount of computing

power because of the combinatorics associated with searching a multi-dimensional space...seemingly innocuous assumptions and implementation decisions can have dramatic consequences downstream in a complex system like a cognitive architecture that interacts with a simulation environment

The tendency in modeling is to focus on “pet problems” where the model succeeds. However, the potential parameter space for any given model is huge. We modelers need to examine our approaches in a broader context, not just the “good fit” space, or comfort zone. This problem is well laid out in Best et al. (2009):

The de-facto approach to cognitive modeling is more often a focus on maximizing fit to human data. This is done through either hand-tuning based on the intuition and experience of the modeler or automated optimizing of the fit...Any of these approaches can be sufficiently successful, but they provide little data about the performance of the model outside of the ultimate parameter values used in presenting the final fit.

Best et al. also point out the benefits of such exploration of parameter space:

Information about how a model performs outside the best-fitting parameter combination provides modelers with information about...the full range of behavior possible from the model and how different parameters interact to generate possibly complex behavioral dynamics.

Our modeling approach is simpler than the typical cognitive architecture of the type Gluck et al. and Best et al. describe (e.g., ACT-R or Soar), but issues of scaling still apply. For this paper, we examined a relatively small search space with a very simple model, but our goals were similar – to cover and better understand a wider space than our typical models explore.

In a recent paper (Santamaria & Warwick, 2009), we gave an overview of our MS/RPD modeling approach, the ground we have covered and tasks we have modeled, and our vision for the next steps to take. In our “next steps” section, we promised to “systematically investigate the computational limits of our algorithms, scaling up a simple model by adding cues and courses of actions.”

To follow through on this promise, we constructed a “generic” model without built-in assumptions about tasks or processes (and the expectations that come with them); the inputs to the model are cue 1 through cue n, and the values of these cues determine the selection of one of m

courses of action (COAs). We used this model to explore issues of scaling by systematically increasing the number of cues and COAs. We went beyond the typical size for MS/RPD models, on the order of 2 cues and 2 COAs, to explore up to 15 cues and 5 COAs. Using the same generic model, we also investigated a range of values for three key model parameters: the activation exponent, the COA selection mechanism, and confidence.

2. The Generic Model: A Testbed

The generic model was developed to explore scaling and parameter space issues. Why did we construct a generic model? In our models, closed form analytic solutions are not obvious or even tractable. Even the simplest cognitive models are fairly complicated pieces of software, and they need to be explored empirically. The generic model can be incrementally scaled up in the number of cues and the number of courses of action. In this section, we describe the underlying learning, memory, and recognition mechanisms and the construction and cue structure of the generic model.

2.1 Learning, Memory, and Recognition Mechanisms

Our decision modeling mechanism was inspired by Klein’s theory of the recognition primed decision, or RPD (see Klein, 1998). It uses a multiple-trace mechanism based on the multiple-trace model of memory (see Hintzman, 1984; 1986a; 1986b). Following Klein, the major features of our modeling approach are cues and COAs, and the associations between them. Models learn the associations between cues and COAs through experience, and this accumulation of this experience can be modified by several recognition and learning parameters. These parameters include the activation exponent, the COA selection mechanism, and confidence, each of which is described in more detail below.

2.2 Construction and Cue Structure

The high-level task structure of the generic model is shown in Figure 1. The first task sets the model parameters, including number of cues, number of COAs, runtime, number of situations, and cue-to-COA mapping.

We explored several different cue-to-COA mappings in order to reduce the chance that we had hidden or “smuggled in” informative structure that essentially gave extra help to the model. Standard experimental paradigms are carefully crafted to have internal structure that is predictable and learnable. The model can latch on to certain kinds of structure, but what happens when the structure is completely arbitrary? We tested several mappings, including random assignment of cue combinations (situations) to COAs (“random”), a list-

based mapping covering all possible combinations (“alternating”), an offset list-based mapping (“offset”), and a mapping based on the location of cues in the situation vector (“left-right”). Results were similar for all mappings; the results reported in this paper used either the random or the alternating mapping.

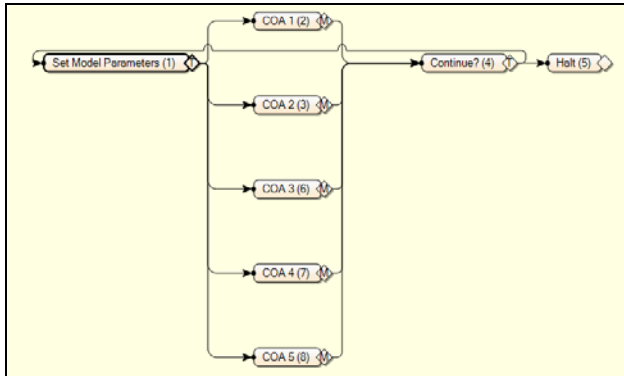


Figure 1. The task structure of the generic model.

After setting model parameters, the task network model passes control to the RPD (decision) model, which selects a COA. Figure 2 shows the screen where cues are specified in the RPD model.

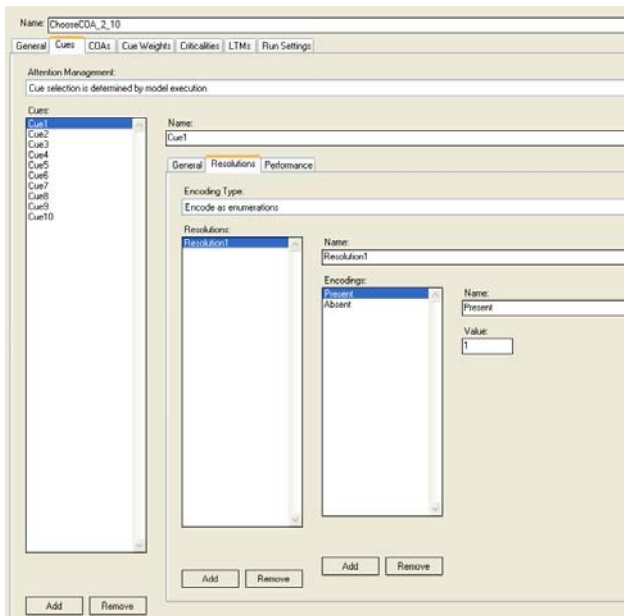


Figure 2. Specifying cues in the decision (RPD) model.

Next, the task network model resumes, goes to the “continue” task, and if runtime is not yet up, loops back to make another decision. There are no actual consequences in the task network model of choosing one COA over another other; that is why we call this a “generic” model that does not have built-in task assumptions.

3. Scaling Up Model Complexity

To test effects of scale and explore a wider range of model size than we typically investigate, we systematically changed the number of cues and the number of COAs in the model.

We tested all combinations of cues and COAs from one to five cues and from two to five COAs. To ensure that all cue situations deterministically predicted a COA, we omitted combinations with fewer cue situations than COAs. An example is the combination of three COAs and one cue (3-1); with one cue, there are two cue situations that cannot uniquely map to three different COAs. The combinations tested are listed in Table 1.

Table 1. Combinations of cues and COAs tested.

Cues	COAs			
	2	3	4	5
1	2-1	X	X	X
2	2-2	3-2	4-2	X
3	2-3	3-3	4-3	5-3
4	2-4	3-4	4-4	5-4
5	2-5	3-5	4-5	5-5

We tested each model holding confidence at medium and the activation exponent at 15. The cue-to-COA mapping was the “alternating” mapping and runtime was 500 trials. Figure 3 and Figure 4 show the results of these tests. They present the same data but group them differently, with Figure 3 showing the effect of number of COAs by grouping the models by number of cues, and Figure 4 showing the effect of number of cues by grouping the models by number of COAs.

Figure 3 shows the effect of number of COAs on learning for models with 2 cues (top left), 3 cues (top right), 4 cues (bottom left), and 5 cues (bottom right). Learning differences are very small for 2 or 3 cues. However, when the number of cues increases to 4 or 5, adding COAs slows learning. Tests with long runs showed that it takes much longer for model 5-5 to reach asymptote than for model 2-5 to reach asymptote.

Figure 4 shows the effect of number of cues on learning for models with 2 COAs (top left), 3 COAs (top right), 4 COAs (bottom left), and 5 COAs (bottom right). Again, learning differences are small for a small number of COAs but grow larger as the number of COAs increase.

4. Exploring Parameter Values

With our generic model, we explored three of the parameters that are available in the MS/RPD modeling

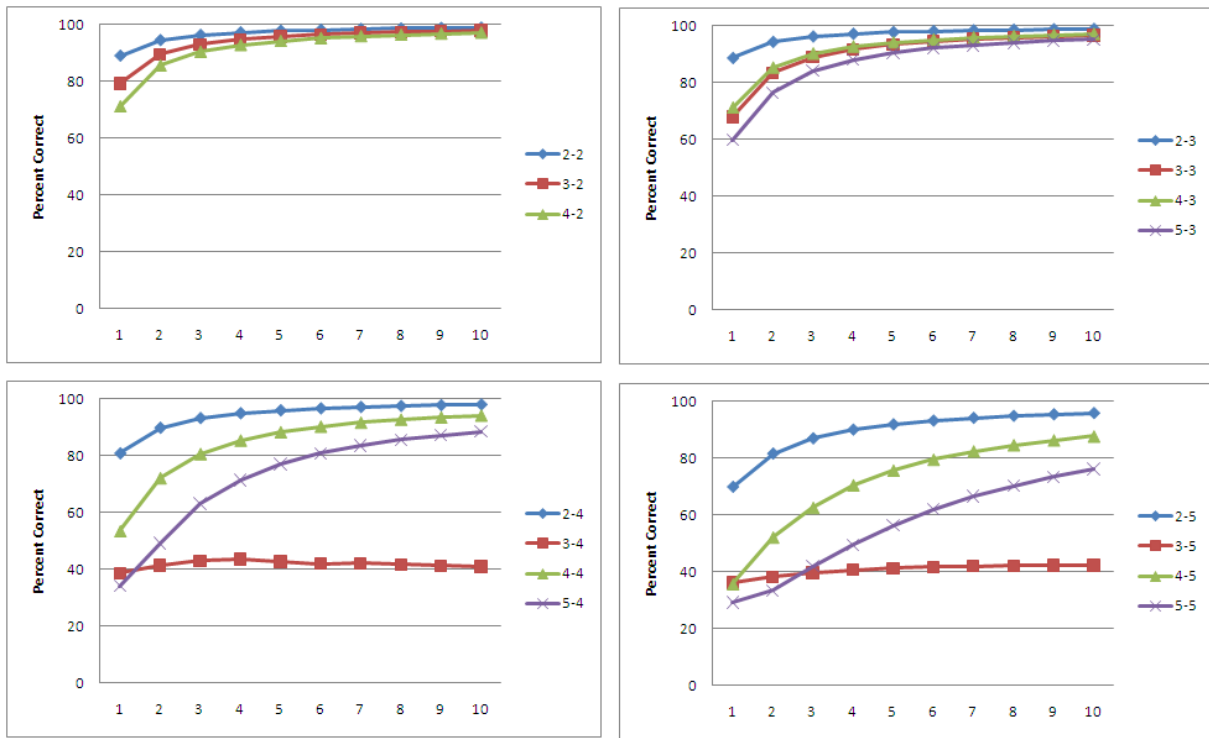


Figure 3. Effect of number of COAs on learning for 2, 3, 4, and 5 cues (left to right, top to bottom). Models are referred to as A-B, where A is the number of COAs and B is the number of cues. Time is on the x-axis (trial/50).

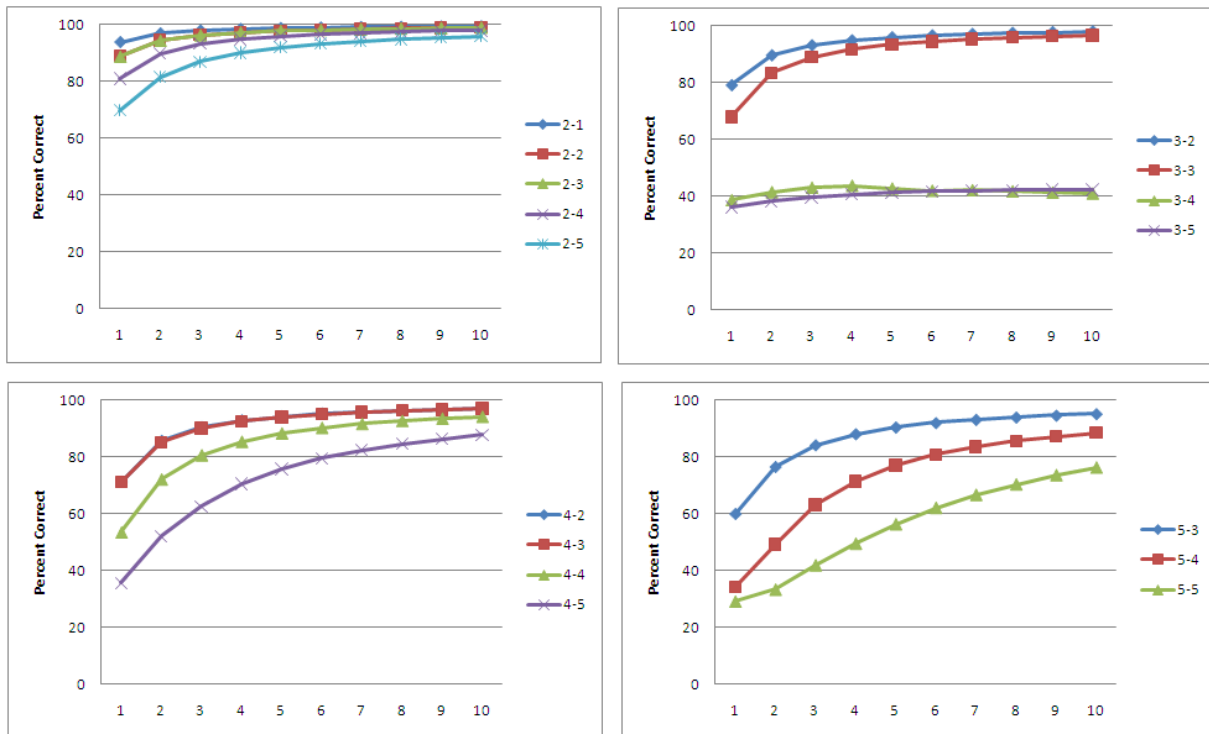


Figure 4. Effect of number of cues on learning for 2, 3, 4, and 5 COAs (left to right, top to bottom). Models are referred to as A-B, where A is the number of COAs and B is the number of cues. Time is on the x-axis (trial/50).

approach: activation exponent, COA selection mechanism, and confidence.

4.1 Activation Exponent

The first parameter we explored with the generic model was the activation exponent. Remember that the MS/RPD approach uses a similarity-based recall mechanism. The similarity value between the current episode and all the episodes in long-term memory is raised to a power, the activation exponent. The similarity value determines the proportion that each remembered episode contributes to the recognition process. A higher value for the activation exponent means that the match must be more exact for the remembered episode to contribute to the current decision.

We tested the 2-10 model (2 COAs and 10 cues), holding confidence at medium and COA selection at default. The cue-to-COA mapping was the “random” mapping, and runtime was 5000 trials. With 2 COAs, chance performance is 50 percent correct. As shown in Figure 5, all versions of the model performed above chance. A higher activation exponent yielded better performance and a faster learning curve.

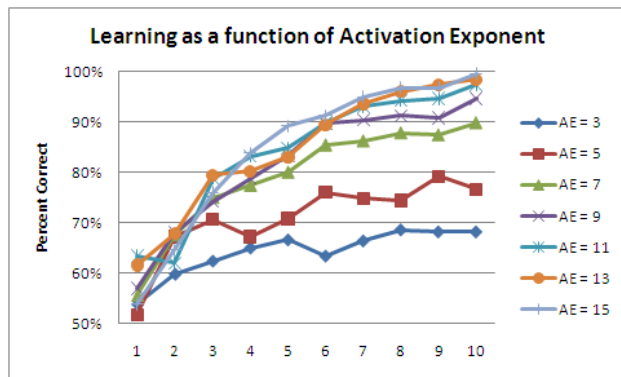


Figure 5. Learning (percent correct over time) as a function of activation exponent for the 2-10 model, for a run of 5000 trials. The x-axis is trial/500.

For this model, activation exponent is an important parameter. Holding everything else constant, it can improve overall performance from 64 percent correct to 85 percent correct. Figure 6 shows overall percent correct (across all trials) for the 2-10 model for activation exponent values of 3 to 15.

4.2 COA Selection Mechanism

The second parameter we explored with the generic model was the COA selection mechanism. The COA selection mechanism controls how the model will choose among recognized courses of action. By default, the model will always choose the COA most strongly

recognized as successful among those that exceed a recognition threshold; conversely, the model will not choose any COAs that have been recognized as unsuccessful. This selection strategy is referred to as “default”.

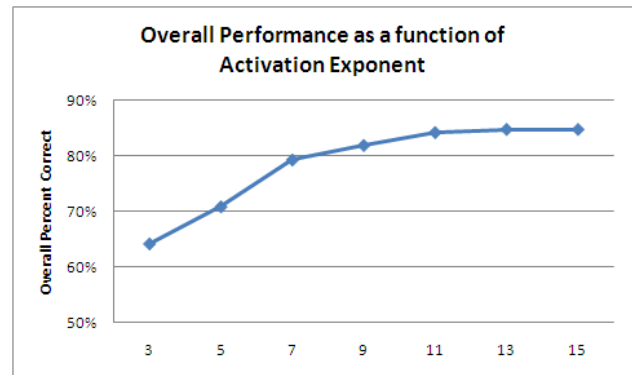


Figure 6. Overall percent correct as a function of activation exponent for the 2-10 model, for 5000 trials.

The default strategy is intended to steer the model toward the most successful COAs. The model can also employ a “fuzzy” selection strategy where it tends to choose the COA recognized as most successful, but not always. The fuzzy option uses a probabilistic draw weighted with respect to the normalized strength of recognition for each COA.

We tested the 2-10 model (2 COAs and 10 cues), holding confidence at none and COA selection at default. The cue-to-COA mapping was the “alternating” mapping. The effect of COA selection mechanism on learning for the first 200 trials is shown in Figure 7. Both default and fuzzy mechanisms result in similar performance, but they differ in the initial spin-up over the first 50 trials. On average, across a batch of ten runs, the model using the default mechanism spins up more quickly.

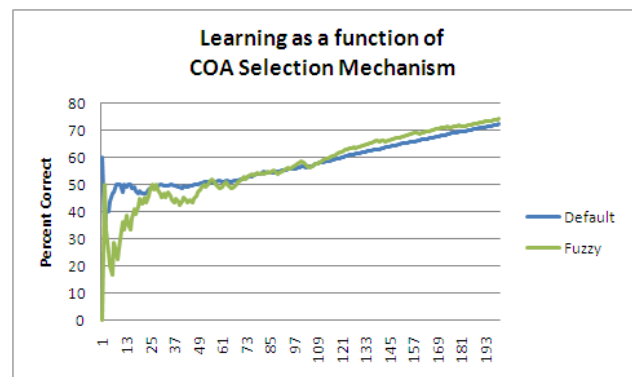


Figure 7. Effect of COA selection mechanism on learning for the first 200 trials. (Default and fuzzy are each averaged over 10 runs.)

4.3 Confidence

The third parameter we explored with the generic model was confidence. Confidence sets a threshold above which the model will recognize a COA. The lower the threshold, the less “confident” you can be that the recognition is due to systematic associations in long term memory between situations and COAs rather than the noise inherent in the similarity-based recognition process. Viewing long-term memory as a “population” of experience, the threshold corresponds to the number of standard deviations from the mean recognition value one would expect from a population of random experiences. Low confidence corresponds to one standard deviation, medium to two standard deviations, and high to three standard deviations.

The effects of confidence should show in early trials, as the model spins up. Early trials are especially important in models that are very sensitive to noise and initial effects. We have seen confidence affect early performance and spin-up in other models. However, our tests did not reveal differences in the generic model for different levels of confidence across a variety of conditions (specific results are not reported here).

5. Discussion

We used the generic model to investigate 1) scaling beyond our typical model size and 2) a range of values for several key model parameters. In the exploration of scaling, we found that we could increase either cues or COAs with only a very minor slowing of learning, but that increasing both beyond three led to a much larger slowdown in learning.

These results demonstrate the syntactic nature of the model. It is not learning anything about specific COAs or cues; it is learning about the combination of COAs and cues. This is evident in the symmetry of the effect of scaling up in number of cues and COAs on performance. It doesn't matter if the increase in decision space size is due to cues or COAs; the model is sensitive to the size of the decision space, not the source of the complexity.

In addition to the results presented here, we built models that scaled up even further: a 2 COA, 10 cue model (2-10), a 2 COA, 15 cue model (2-15), and a 5 COA, 10 cue model (5-10). The 2-10 model was able to learn to asymptote, although it took longer to reach asymptote than did models whose number of cues/number of COAs were capped at 5. The 2-15 and 5-10 models were not able to converge, even with runtimes of 25,000 trials. This was because of the very large space to learn (all combinations of cues were possible and had an assigned “correct answer”). For example, the 2-10 model had 2^{10} ,

or 1024, possible cue combinations. The 2-15 model had 2^{15} , or 32,768, and the 5-10 model had 5^{10} , or 9,765,625! When we limited the number of possible cue combinations the model could face (to 50, 100, even 500), the 2-15 and 5-10 models were able to learn without a problem. So scaling up the cue and COA space and scaling up the situation space are actually separate issues.

Two of the parameters we examined provided interesting results: activation exponent and the COA selection mechanism. The value of the activation exponent made a substantial difference in the model's learning and performance. The higher the activation exponent, the faster the learning. Differences were largest among smaller activation exponents (3 to 7), and learning curves became more similar for higher values (9-15). Overall performance (percent correct) also improved as activation exponent increased, with the largest differences at the small end of the parameter scale.

It was important to explore the full range of possible activation exponent values because they did not uniformly affect performance and learning. The lesson from our exploration of this parameter is that you need to make sure the activation exponent is high enough (maybe 7 or higher), but beyond a certain point, it does not make much of a difference in the model's performance.

The COA selection mechanism showed a difference in learning but not performance. On average, the model reached similar levels of accuracy with default and fuzzy mechanisms, but it learned faster with default, showing better performance than fuzzy on the first 50 trials.

There were two puzzling results with the generic model that have not yet been explained. The first puzzling result was that model performance on the 3 COA, 4 cue (3-4) and 3 COA, 5 cue (3-5) models stagnated at chance performance. We suspect this is an anomaly resulting from the way cues were mapped to COAs (the “right answers” for which the model was reinforced).

The second puzzling result was the absence of a result for confidence. Earlier models have shown effects of confidence, particularly on early performance and spin-up. However, the generic model failed to show an effect of confidence under a variety of conditions. An effect of confidence should show up where there are systematic associations over and above the noise present. However, in the generic model, we deliberately built random cue-to-COA mappings - this is only noise! So there are no systematic associations inherent in cue structure. Finding no effect of confidence in this model is actually a validation that we haven't smuggled in any informative internal structure or biases, providing a purer test of the model's ability to learn essentially arbitrary relationships.

6. Conclusions

In this paper, we have described our integrated modeling approach and our attempts to push its boundaries a bit. While it is important for a modeling approach to build a repertoire of single-task models validated with human performance data, we have argued that it is also important to explore beyond the "good fit" areas of parameter space and the "typical model" areas of complexity space/scale.

Examining a relatively small search space with a very simple "generic" model, we attempted to gain a better understanding of a larger space than we typically explore with our models. We learned some interesting things as we tried to scale up the model and systematically move across parameter space.

This is just the beginning of this effort. It is critical to go beyond holding all parameters but one constant in order to explore the intersection of parameter space and to understand how model parameters interact. These efforts are a very small step in an enormous and intimidating effort that is emerging in the modeling community: putting our modeling endeavors in a broader context and moving outside our modeling comfort zones.

7. References

- Best, B. J., Furjanic, C., Gerhart, N., Fincham, J., Gluck, K. A., Gunzelmann, G., & Krusmark, M. A. (2009). Adaptive mesh refinement for efficient exploration of cognitive architectures and cognitive methods. *Proceedings of the ?? International Conference on Cognitive Modeling*.
- Collyer, C. E. (1985). Comparing strong and weak models by fitting them to computer-generated data. *Perception & Psychophysics*, 38, 476-481.
- Gluck, K., Scheutz, M., Gunzelmann, G., Harris, J., & Kershner, J. (2007). Combinatorics meets processing power: Large-scale computational resources for BRIMS. In *Proceedings of the Sixteenth Conference on Behavior Representation in Modeling and Simulation* (pp. 73-83). Orlando, FL: Simulation Interoperability Standards Organization.
- Hintzman, D. L. (1984). MINERVA 2: A simulation model of human memory. *Behavior Research Methods, Instruments & Computers*, 16, 96-101.
- Hintzman, D. L. (1986a). *Judgments of Frequency and Recognition Memory in a Multiple-Trace Memory Model*. Eugene, OR: Institute of Cognitive and Decision Sciences.
- Hintzman, D. L. (1986b). "Schema Abstraction" in a Multiple-Trace Memory Model. *Psychological Review*, 93, 411-428.
- Klein, G. (1998). *Sources of Power: How People Make Decisions*. Cambridge, MA: The MIT Press.
- Roberts, S. & Pashler, H. (2000). How persuasive is a good fit? A comment on theory testing. *Psychological Review*, 107, 358-367.
- Santamaria, A. & Warwick, W. (2007). A naturalistic approach to adversarial behavior: Modeling the prisoner's dilemma. In *Proceedings of the 16th Conference on Behavioral Representations in Modeling and Simulation*.
- Santamaria, A. & Warwick, W. (2008). Modeling probabilistic category learning in a task network model. In *Proceedings of the 17th Conference on Behavioral Representations in Modeling and Simulation*.
- Warwick, W. & Fleetwood, M. (2006). A bad Hempel day: The decoupling of explanation and prediction in computational cognitive modeling. In *Proceedings of the Fall 2006 Simulation Interoperability Workshop, Orlando, FL, SISO*.
- Warwick, W. & Hutchins, S. (2004). Initial comparisons between a "naturalistic" model of decision making and human performance data. In *Proceedings of the 13th Conference on Behavior Representation in Modeling and Simulation*.
- Warwick, W., McIlwaine, S., Hutton, R. J. B., & McDermott, P. (2001). Developing computational models of recognition-primed decision making. In *Proceedings of the 10th Conference on Computer Generated Forces*.
- Warwick, W. & Santamaria, A. (2006). Giving up vindication in favor of application: Developing cognitively-inspired widgets for human performance modeling tools. *Proceedings of the 7th International Conference on Cognitive Modeling*.

Author Biographies

AMY SANTAMARIA is a Senior Cognitive Scientist at Alion Science and Technology. Her research focuses on modeling human behavior and cognition and experimentation for robotics interfaces. She received her Ph.D. in Cognitive Psychology and Neuroscience and an M.A. in Cognitive Psychology from the University of Colorado Boulder.

WALTER WARWICK is a Principal Systems Analyst at Alion Science and Technology. He is working on several projects having to do with the modeling and simulation of human behavior. He received his Ph.D. in History and Philosophy of Science, an Area Certificate in Pure and Applied Logic, and an M.S. in Computer Science from Indiana University.