A Comparative Approach to Understanding General Intelligence: Predicting Cognitive Performance in an Open-ended Dynamic Task

Christian Lebiere[1], Cleotilde Gonzalez[2] and Walter Warwick[3]

[1] Psychology Department, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh PA 15213
[2] Department of Social and Decision Sciences, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh PA 15213
[3] Alion Science and Technology, 4949 Pearl East Circle, Boulder, CO 80401
cl@cmu.edu, coty@cmu.edu, wwarwick@alionscience.com

Abstract

The evaluation of an AGI system can take many forms. There is a long tradition in Artificial Intelligence (AI) of competitions focused on key challenges. A similar, but less celebrated trend has emerged in computational cognitive modeling, that of model comparison. As with AI competitions, model comparisons invite the development of different computational cognitive models on a well-defined task. However, unlike AI where the goal is to provide the maximum level of functionality up to and exceeding human capabilities, the goal of model comparisons is to simulate human performance. Usually, goodness-of-fit measures are calculated for the various models. Also unlike AI competitions where the best performer is declared the winner, model comparisons center on understanding in some detail how the different modeling "architectures" have been applied to the common task. In this paper we announce a new model comparison effort that will illuminate the general features of cognitive architectures as they are applied to control problems in dynamic environments. We begin by briefly describing the task to be modeled, our motivation for selecting that task and what we expect the comparison to reveal. Next, we describe the programmatic details of the comparison, including a quick survey of the requirements for accessing, downloading and connecting different models to the simulated task environment. We conclude with remarks on the general value in this and other model comparisons for advancing the science of AGI development.

Introduction

The evaluation of an AGI system can take many forms. Starting with Turing (e.g., Turing, 1950), the idea that artificial intelligence might be "tested" has led quite naturally to a tradition of competition in AI in which various systems are pitted against each other in the performance of a well-specified task. Among the most famous include the Friedkin prize for a machine chess player that could beat the human chess champion (Hsu, 2002), the robocup soccer competition for autonomous robots (Asada et al., 1999) and the DARPA Grand Challenge race across the desert (Thrun et al., 2006). A similar, but less celebrated trend has emerged in computational cognitive modeling, that of model comparison. As with AI competitions, model comparisons invite the development of different computational cognitive models on a well-defined task. However, unlike AI where the goal is to provide the maximum level of functionality up to and exceeding human capabilities, the goal of model comparisons is to most closely simulate human performance. Thus, usually, goodness-of-fit measures are calculated for the various models. Also, unlike AI competitions where the best performer is declared the winner, model comparisons center on understanding in some detail how the different modeling "architectures" have been applied to the common task. In this regard model comparisons seek to illuminate general features of computational approaches to cognition rather than identify a single system that meets a standard of excellence on a narrowly defined task (Newell, 1990).

In this paper we announce a new model comparison effort that will illuminate the general features of cognitive architectures as they are applied to control problems in dynamic environments. We begin by briefly describing the general requirements of a model comparison. Next, we describe the task to be modeled, our motivation for selecting that task and what we expect the comparison to reveal. We then describe the programmatic details of the comparison, including a quick survey of the requirements for accessing, downloading and connecting different models to the simulated task environment. We conclude with remarks on the general value we see in this and other model comparison for advancing the science of AGI development. Although everyone loves a winner, the real value of a model comparison is found in its methodological orientation. Indeed, given the inherent flexibility of computational abstractions, understanding the workings of a particular cognitive system, much less judging its usefulness or "correctness," is not easily done in isolation.

General Requirements of a Model Comparison

We have gained direct experience from a number of modeling comparisons projects, including the AFOSR AMBR modeling comparison (Gluck & Pew, 2005) and

the NASA Human Error Modeling comparison (Foyle & Hooey, 2008). We have also entered cognitive models into multi-agent competitions (Billings, 2000; Erev et al, submitted) and organized symposia featuring competition between cognitive models as well as mixed human-model competitions (Lebiere & Bothell, 2004; Warwick, Allender, Strater and Yen, 2008). From these endeavors, we have gained an understanding of the required (and undesirable) characteristics of a task for such projects.

While previous modeling comparison projects did illustrate the capabilities of some modeling frameworks, we found that the tasks were often ill-suited for modeling comparison for a number of reasons:

- The task demands a considerable effort just to model the details of task domain itself (and sometimes, more practically, to connect the model to the task simulation itself). This often results in a model whose match to the data primarily reflects the structure and idiosyncrasies of the task domain itself rather than the underlying cognitive mechanisms. While this task analysis and knowledge engineering process is not without merit, it does not serve the primary purpose of a model comparison effort, which is to shed light upon the merits of the respective modeling frameworks rather than the cleverness and diligence of their users.
- The task is defined too narrowly, especially with regard to the data available for model fitting. If the task does not require model functionality well beyond the conditions for which human data is available, then the comparison effort can be gamed by simply expanding effort to parameterize and optimize the model to the data available. This kind of task puts frameworks that emphasize constrained, principled functionality at a disadvantage over those that permit arbitrary customization and again serves poorly the goals of a modeling comparison.
- The task is too specialized, emphasizing a single aspect, characteristic or mechanism of cognition. While this type of task might be quite suitable for traditional experimentation, it does not quite the kind of broad, general and integrated cognitive capabilities required of a general intelligence framework.
- No common simulation or evaluation framework is provided. While this allows each team to focus on the aspects of the task that are most amenable to their framework, it also makes direct comparison between models and results all but impossible.
- No suitably comparable human data is available. While a purely functional evaluation of the models is still possible, this biases the effort toward a pure competition, which emphasizes raw functionality at the expense of cognitive fidelity.

This experience has taught us that the ideal task for a model comparison is:

- lightweight, to limit the overhead of integration and the task analysis and knowledge engineering requirements
- fast, to allow the efficient collection of large numbers of Monte Carlo runs
- open-ended, to discourage over-parameterization and over-engineering of the model and test its generalization over a broad range of situations
- dynamic, to explore emergent behavior that is not predictable from the task specification
- simple, to engage basic cognitive mechanisms in a direct and fundamental way
- tractable, to encourage a direct connect between model and behavioral data

Like other enduring competitive benchmarks of human cognition that have kept on driving the state of the art in some fields (e.g. Robocup), the key is to find the right combination of simplicity and emergent complexity. We believe the task we have selected, described in detail below, meets these requirements and strikes the right combination between simplicity and complexity. In fact, in our own pilot studies, we have encountered significant challenges in developing models of the task that could account for even the basic results of the data and our models have consistently surprised us by their emergent behavior, and even minor changes in task representation have had deep consequences for model behavior (Lebiere, Gonzalez, & Warwick, under review). We expect the same will be true for other participants in this effort.

The Dynamic Stocks and Flows Task

In dynamic systems, complexity has often been equated with the number of elements to process at a given time: goals, alternatives, effects and processes (Brehmer & Allard, 1991; Dorner, 1987). Researchers have investigated the problems and errors that people make while dealing with this type of complexity in dynamic systems to foster our understanding of decision making. However, dynamic systems manifest another type of complexity that is less well known, that is *dynamic complexity* (Diehl & Sterman, 1995). This type of complexity does not depend on the number of elements to process in a task. In fact, the underlying task could be superficially simple depending on a single goal and one element to manage and make decisions. Dynamic complexity follows from the combinatorial relationships that arise from the interactions of even a few variables over time.

Gonzalez & Dutt (2007) and Dutt & Gonzalez (2007) have investigated human performance in dynamically complex environments using a simple simulation called the dynamic stocks and flows (DSF). DSF (see figure 1) is an interactive learning tool that represents a simple dynamic system consisting of a single *stock* in which the rate of accumulation is a function of time; *inflows* increase the level of stock, and *outflows* decrease the level of stock. The goal in DSF is to maintain the stock within an acceptable range over time. The stock is influenced by external flows (External Inflow and Outflow) that are out of the user's control, and by user flows (User Inflow and Outflow) that the player of DSF decides on in every time period.

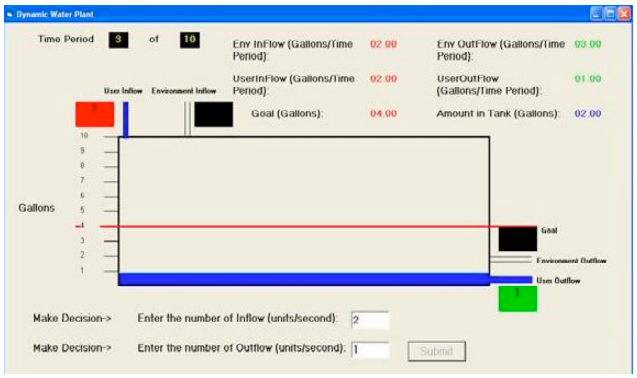

Figure 1: DSF Interface

A stock, inflows and outflows are the basic elements of every dynamic task, at the individual, organizational, and global levels (for a discussion of the generality of this structure in complex dynamic systems, see Cronin, Gonzalez, & Sterman, 2008). For example, the structure of the task discussed here, is currently being used to investigate the problems of control of the atmospheric CO2, believed to lead to Global Warming (Dutt & Gonzalez, 2008).

Despite its seeming simplicity, controlling the DSF is very difficult for most subjects (Gonzalez & Dutt, 2007; Dutt & Gonzalez, 2007). For example, Cronin, Gonzalez & Sterman (2008) found that in a sample of highly educated graduate students with extensive technical training nearly half were unable to predict the qualitative path of a stock given very simple patterns for its inflow and outflow. Subject learning was slow and ultimately sub-optimal even in the simple conditions of the task, for example, that of controlling the system due to an increasing inflow and zero outflow (Gonzalez & Dutt, 2007; Dutt & Gonzalez, 2007). Moreover, Cronin and Gonzalez (2007) presented subjects with a series of manipulations related to the form of information display, context of the task, incentives and others factors intended to help the subject understand the task, demonstrating that the difficulty in understanding the DSF is not due to lack of information or the form of information presentation.

For all the difficulty subjects have controlling DSF, the task environment itself is easily modeled and extended. The state of the task environment is completely determined by the functional relationship among inflow, outflow, user action and stock, while the functions themselves can be modified in direct ways. For example, stochastic "noise" can be added to the functions that control environmental inflow and outflow to explore the effects of uncertainty; the addition of different or variable delays between user actions and outcomes changes the nature of the dynamic complexity; finally, the task lends itself to the exploration of team or adversarial performance simply by allowing another agent to control the environmental inputs and outputs.

Participating in the DSF Model Comparison

Participation in this model comparison begins with a visit to the DSF Model Comparison website: http://www.cmu.edu/ddmlab/ModelDSF. There, potential participants will be asked to register for the competition. Registration is free, but is required so that we can plan to allocate adequate resources to the evaluation of the participants' models (as described below).

At the website, participants will find a more detailed description of the DSF task and a free downloadable version of the task environment. The DSF task environment requires a Windows platform and can be run in two modes. First, the DSF can be run as a live experiment so that participants can interact with exactly the same task environment the subjects used in the experiments. In this way, modelers can gain hands-on experience with the task and use this experience to inform the development of their own models. Second, the DSF environment can be run as a constructive simulation, without the user interface, in a faster-than-real time mode with the participants' computational models interacting directly with the task environment.

The DSF uses a TCP/IP socket protocol to communicate with external models. Details about the "client" requirements and the communication syntax will be available on the website, along with example software code for connecting to the DSF.

Once participants have established a connection to the DSF environment, we invite them to calibrate their models running against the "training" protocols and comparing model performance against human performance data. Both the training protocols and data will be available from the website. In this way, participants will be able to gauge

whether their models are capable of simulating the basic effects seen in human control of the DSF task. Our past experience suggests that this will lead to an iterative development process where models are continually refined as they are run under different experimental protocols and against different data sets.

Model comparison begins only after participants are satisfied with the performance they have achieved on the training data. At that point, participants will submit executable version of their model through the website to be run against "transfer" protocols. As we indicated above, the DSF task supports several interesting variants. We are currently running pilot studies with human subjects to identify robust effects under these various conditions. The choice of specific transfer conditions will be entirely at our discretion and submitted models will be run under these conditions as-is.

Our goal for this blind evaluation under the transfer condition is not to hamstring participants, but to see how well their models generalize without the benefit of continual tweaking or tuning. Assessing robustness under the transfer condition is an important factor to consider when we investigate the invariance of architectural approaches. That said, goodness-of-fit under the training and transfer conditions is not the only factor will use in our comparison effort. In addition to submitting executable versions of their models, we will require participants to submit written accounts of their development efforts and detailed explanations of the mechanisms their models implement. As we discuss below, this is where model comparisons bear the most fruit. Again, based on our past experience, we recognize that it is difficult to explain the workings of a cognitive model to the uninitiated, but it is exactly that level of detail that is required to understand what has been accomplished.

On the basis of both model performance and written explanation, we will select three participants to present their work at the 2009 International Conference on Cognitive Modeling (http://web.mac.com/howesa/Site/ICCM_09.html). We will also cover basic travel expense to that conference. Finally, participants will be invited to prepare manuscripts for publication in a Special Issue of the Journal for Cognitive Systems Research (http://www.sts.rpi.edu/~rsun/journal.html) devoted to the topic of model comparison.

Model Comparison as Science

The call for the development of an artificial general intelligence is meant to mark a turn away from the development of "narrow AI." From that perspective, a model comparison might seem to be an unwelcome return to the development of one-off systems engineered to excel only on well-specified highly-constrained tasks. It would be a mistake, however, to view the outcome of a model comparison as merely a matter of identifying the approach that produces the best fit to the human performance data on a specific task. Rather, a goodness-of-fit measure is only a minimum standard for the more detailed consideration of the computational mechanisms that lead to that fit. Insofar as these mechanisms implement invariant structures, they shed light on the general nature of cognition. But the devil is in the details; understanding whether an architecture actually constrains the modeling approach and thereby shed some insight into the general features of cognition, or whether it merely disguises the skill of the clever modeler is never easy.

This difficulty is compounded by several other factors. First, as Roberts and Pashler (2000) have pointed out, good human performance data are hard to come by and it is harder still to see these data, by themselves, can undergird an *experimentum crucis* among different modeling approaches. Simply put, even good fits to good data will underdetermine the choices of architecture. This is not merely a problem of loose data, but is also due to one of the defining insights of computation, that of Turing equivalence and the related notion in the philosophy of mind of multiple realizability. The fact that any algorithm can be implemented by any number of Turing-equivalent mechanisms all but guarantees some degree of underdetermination when we consider the relationship between model and theory. Unless one is willing to engage in a question-begging argument about the computational nature of mind, Turing equivalence does not guarantee theoretical equivalence when it comes to cognitive modeling and different computational mechanisms will come with different theoretical implications.

While some might argue that this latter problem can be addressed by fixing the appropriate level of abstraction this otherwise sound advice has had the practical effect of allowing the modeler to decide what the appropriate relationship is between model and theory. Moreover, proposing an abstraction hierarchy, no matter how elegant or appealing, is not the same thing as discovering a natural kind, and it remains an empirical endeavor to establish whether the abstractions we impose really carve the nature of cognition at the joints. Thus, correspondence is too often asserted by fiat, and notions like "working memory," "situation awareness" "problem detection" and such are reduced to simple computational mechanisms without any serious theoretical consideration. Those concerned about the implementation of a general intelligence are left to wonder whether if-then-else is all there is to it.

A number of tests for a general theory of intelligence have been advanced (e.g. Cohen, 2005; Selman et al, 1996;

Anderson & Lebiere, 2003). A key common aspect is to enforce generality in approach, in order to prevent special-purpose optimization to narrow tasks and force integration of capabilities. One can view that strategy as effectively overwhelming the degrees of freedom in the architecture with converging constraints in the data. However, precise computational specifications of those tests have to tread a tight rope between requiring unreasonable amounts of effort in modeling broad and complex tasks and falling back into narrow task specifications that will again favor engineered, optimized approaches. This model competition is our attempt at testing general cognitive capabilities in an open-ended task while offering low barriers to entry.

We see model comparison as one solution to these problems. Model comparison is not just a practical solution for understanding how different systems work, but a theoretical prescription for identifying invariant structures among different approaches, seeing how they are, in fact, applied to specific problems and a way of seeing past the buzzwords to the mechanism that might finally illuminate what is needed to realize an artificial general intelligence.

References

Anderson, J. R. & Lebiere, C. L. 2003. The Newell test for a theory of cognition. *Behavioral & Brain Sciences 26*, 587-637.

Asada, M., Kitano, H., Noda, I., and Veloso, M. 1999. RoboCup: Today and tomorrow – What we have have learned. *Artificial Intelligence, 110*:193–214.

Brehmer, B., & Allard, R. 1991. Dynamic decision-making: The effects of task complexity and feedback delay. In J. Rasmussen, B. Brehmer & J. Leplat (Eds.), *Distributed decision making: Cognitive models of cooperative work* (pp. 319-334). Chichester: Wiley.

Billings, D. 2000. The First International RoShamBo Programming Competition. *ICGA Journal*, Vol. 23, No. 1, pp. 42-50.

Cohen, P. 2005. If Not Turing's Test, Then What? *AI Magazine 26(4)*: 61–67.

Cronin, M., & Gonzalez, C. 2007. Understanding the building blocks of dynamic systems. *System Dynamics Review.* 23(1), 1-17.

Cronin, M., Gonzalez, C., & Sterman, J. D. 2008. Why don't well-educated adults understand accumulation? A challenge to researchers, educators and citizens. In press. *Organizational Behavior and Human Decision Processes*.

Diehl, E., & Sterman, J. D. 1995. Effects of feedback complexity on dynamic decision-making. *Organizational Behavior and Human Decision Processes, 62*(2), 198-215.

Dutt, V. & Gonzalez, C. 2008. Human Perceptions of Climate Change. The 26th International *Conference of the System Dynamics Society*. (pp.). Athens, Greece: System Dynamics Society.

Dutt, V. & Gonzalez, C. 2007. Slope of Inflow Impacts Dynamic Decision Making. The 25th International *Conference of the System Dynamics Society*. (pp. 79). Boston, MA: System Dynamics Society.

Erev, I., Ert, E., Roth, A. E., Haruvy, E., Herzog, S., Hau, R., Hertwig, R., Stewart, T., West, R., & Lebiere, C. (submitted). A choice prediction competition, for choices from experience and from description. *Journal of Behavioral Decision Making*.

Foyle, D. & Hooey, B. 2008. *Human Performance Modeling in Aviation*. Mahwah, NJ: Erlbaum.

Gluck, K, & Pew, R. 2005. *Modeling Human Behavior with Integrated Cognitive Architectures*. Mahwah, NJ: Erlbaum.

Gonzalez, C., & Dutt, V. 2007. Learning to control a dynamic task: A system dynamics cognitive model of the slope effect. In *Proceedings of the 8th International Conference on Cognitive Modeling*, Ann Arbor, MI.

Hsu, F-H. 2002. *Behind Deep Blue: Building the Computer that Defeated the World Chess Champion.* Princeton University Press.

Lebiere, C., & Bothell, D. 2004. Competitive Modeling Symposium: PokerBot World Series. In *Proceedings of the Sixth International Conference on Cognitive Modeling*, Pp. 32-32.

Lebiere, C., Gonzalez, C., & Warwick, W. (submitted). Emergent Complexity in a Dynamic Control Task: Model Comparison.

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

Roberts, S. and Pashler, H. 2000. "How Persuasive Is a Good Fit? A Comment on Theory testing." *Psychological Review* 107(2): pp358-367.

Schunn, C. D. & Wallach, D. 2001. Evaluating goodness-of-fit in comparisons of models to data. Online manuscript. http://lrdc.pitt.edu/schunn/gof/index.html

Selman, B., Brooks, R., Dean, T., Horvitz, E., Mitchell, T., & Nilsson, N. 1996. Challenge problems for artificial intelligence. In *Proceedings of the 13th Natl. Conf. on Artificial Intelligence (AAAI-96)*, Portland, OR, pp. 193-224.

Thrun, S. et al. 2006. Stanley, the robot that won the DARPA Grand Challenge. *Journal of Field Robotics, 23(9)*, 661–692.

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

Warwick, W., Allender, L., Strater, L., & Yen, J. 2008. AMBR Redux: Another Take on Model Comparison. Symposium given at the *Seventeenth Conference on Behavior Representation and Simulation*. Providence, RI.