Understanding the behaviour of agents and cognitive models: CaDaDis 2.1

Different representations can reveal different aspects of models. The inverse is also true: namely, that different representations *hide* different aspects of models. It is, therefore, important for multiple representations of a model to be used during development in order to better understand it. We are developing a categorical data display (CaDaDis) for visualizing the sequential behaviour of cognitive models that is flexible both in the array of visualizations possible and in the cognitive architectures that can use it. The overall goal of the project is to provide a common tool for cognitive modellers to work with visualizations of model output.

CaDaDis[1] contains Gantt, program evaluation and review technique (PERT), and modified-PERT visualizations. The firing of actions in the model (e.g. operators or rules) provides data to plot. PERT charts depict task and task duration, along with dependency information, and our recent development work has focused on refining the modified PERT chart. In its latest release (2.1), CaDaDis introduces two new visualization features that allow it to more adequately address the needs of modelers. Specifically, we have added the ability to describe a model's sequential actions in the context of another sequence, and also a means for developers to manipulate a view through automated rearrangement of the categorical display. The new features are built on top of the modified PERT chart included in the original CaDaDis release.

Figure 1 shows a view of a dTank agent running in Soar. The display shows a sequence in the run where the agent has waited, and then located an adversary to attack. The left panel contains a list of operators that have been applied. The right panel displays the order of operator applications. The x-axis of the diagram represents the time at which the operator applied while the y-axis location aligns the operator application with its name. CaDaDis generates this representation as the model runs and can save it so that it can be opened later without running the model again.

Because CaDaDis provides common visualizations of different models and architectures, it allows for clear comparisons. This can be accomplished by studying similarities and differences between the sequences that different models generate. Previously, in order to compare model behaviour in CaDaDis, one had to store the images generated and compare them side by side. The new features in CaDaDis version 2.1 better support the important task of behaviour comparison. In addition, it now supports displaying the same model with multiple displays of its behaviour. The new features allow for comparing similar models by displaying multiple similar runs on a single graph. We chose to allow for multiple runs on one (as opposed to two) visualizations so that users can compare how different models solve the same problem (taking an idea from Sun et al., 2004).[2]

Figure 2 shows the new view provided in CaDaDis 2.1. This particular example shows two independent runs of the *Eight-Puzzle* demo provided with Soar. The top line represents one run, the bottom a second. The first operator shown in this figure is the seventh in the sequence (a keen observer will notice that the window has been scrolled). Up until the twelfth operator, the runs are identical, but then the runs diverge, and one terminates. As one can imagine, a more complex model with a longer run-time can produce the same type of activity: namely, one in which the runs only differ slightly. This new capability of seeing both runs at once immediately shows where the models' behaviour differs.

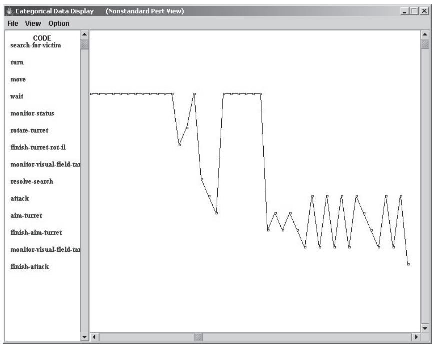

Figure 1. *A dTank agent finding an adversary in a CaDaDis-modified PERT Chart.*

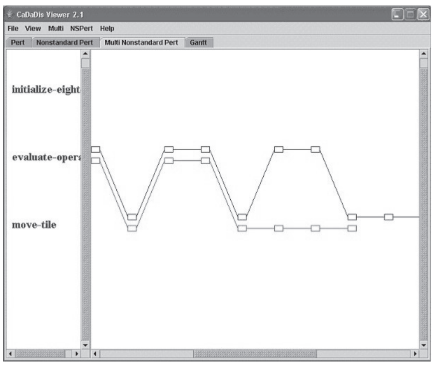

Figure 2. *Soar Eight-Puzzle in a CaDaDis multiple-run-modified PERT Chart.*

We have also added a sorting capability, whereby users can have the actions rearranged by one of several algorithms. Those currently implemented include order of arrival, frequency and reverse-frequency of application.

Now that CaDaDis can compare multiple runs on one visualization and manipulate the operator display order, we have the means to explore more advanced interactions and manipulations.

CaDaDis is available for download[4] and, for academics, it requires a no-cost license from Soar Technology, Inc. to support Vista, a visualization framework for cognitive models that serves as the underpinning architecture. CaDaDis currently provides direct support for Soar, ACT-R, and JESS and other models and architectures that use Tcl, LISP, or Java, will be able to reuse the existing application program interfaces fairly directly. CaDaDis can also load a series of actions from a file.

CaDaDis is supported by the Office of Naval Research through a subcontract from Soar Technology, #VISTA03-1. Thanks to Dwight Berry, Geoff Morgan, Andrew Reifers, Bill Stevenson, and Glenn Taylor for their comments and input.

John Daughtry and Frank Ritter
School of Info. Sciences and Technology
Pennsylvania State University, PA USA
E-mail: {jmd514, frank-ritter}@psu.edu
http://acs.ist.psu.edu

References

1. K. Tor, S. R. Haynes, F. E. Ritter and M. A. Cohen, *Categorical data displays generated from three cognitive architectures illustrate their behaviour,* **Proc. Sixth Int'l Conf. on Cognitive Modeling,** pp. 302-307, 2004.
2. S. Sun, I. Councill, X. Fan, F. E. Ritter, and J. Yen, *Comparing teamwork modeling in an empirical approach,* **Proc. Sixth Int'l Conf. on Cognitive Modeling,** pp. 388-389, 2004.
3. http://www.soartech.com/
4. http://acs.ist.psu.edu/CaDaDis/

AISB Quarterly No.119

The Newsletter of the Society for the Study of Artificial Intelligence and Simulation of Behaviour

In this issue:

Features
- **Hamilton/Evans** 2 *Undersea inspection with pilotless vehicle*
- **Dirk Van Rooy** 3 *Modelling socially-shared cognition*
- **Daughtry/Ritter** 4 *CaCaDis 2.1*
- **Roman Belavkin** 5 *Entropy/information in learning behaviour*
- **Mark Bishop** 6 *Can computers feel?*

Reviews/ previews
- **Conference** 8 *Witkowski on TAROS 2004*
- **Conference** 9 *AISB 2005 symposium details*
- **Book** 10 *Gary Jones on new Amos Tversky collection*

Community
- **Obituary** 11 *Mark Steedman on Longuet-Higgins*

Father Hacker
- *How to argue* 12

Don't forget to vote!

Voting papers enclosed with this issue

Implementing machine consciousness

The last decade has seen the emergence of the new discipline of consciousness studies, which is now well established with journals (e.g. *Cognition and Consciousness, Journal of Consciousness Studies*), societies (ASSC, the Association for the Scientific Study of Consciousness), and conferences (the biennial *Toward a Science of Consciousness*, and the annual ASSC meetings). Consciousness studies encompass a huge range of activities: these are centred around psychology, but ranging from philosophy to neuroscience.

Most recently, a new strand, machine consciousness, has emerged.[1] In the 1990s, a small number of pioneers, including Stan Franklin in AI, Igor Aleksander in electrical engineering, and Gerald Edelman in biology, had begun to examine the possibility of creating consciousness in machines or software. Growing interest in these activities led to the 2001 Banbury Workshop *Can a machine be conscious?* At the end of this highly interdisciplinary event, the twenty-odd participants were asked to vote on whether they thought machine consciousness was in fact a possibility, and all except one agreed that it was. The outcome both surprised the participants, and gave the topic an impetus which is reflected in the level of interest in the symposium *Next generation approaches to machine consciousness* at AISB 2005.

We set up the Machine Consciousness Group at Essex to explore the possibility of achieving machine consciousness in a robot through the formation and exploitation of internal models of itself and the world. This enterprise has been facilitated by the University's support, via a £4.5m SRIF (Science Research Investment Fund) grant, for the building of a new robotic arena and workshop complex. Current work is directly funded by an EPSRC (Engineering and Physical Sciences Research Council) Adventure Fund award of £493,000, shared between myself and the visual neuropsychologists Tom Troscianko and Iain Gilchrist of the University of Bristol. This project, which runs until April 2007, involves the design and construction of a humanoid robot, and the study of the nature and development of its consciousness-related processes based on internal modelling.

Why a humanoid robot? There are several reasons for this. We do not know very much about consciousness, but there has been a growing consensus that the origins of at least the lower levels of conscious phenomena are very strongly rooted in the body. If we are to produce consciousness in a machine, then its embodiment will be a critical determinant of the nature of that consciousness, and of its intelligibility and relevance to human consciousness. The robot (Cronos) is therefore being given a gross physical structure as far as possible qualitatively similar to the human body.

Figure 1 shows the prototype (modelled on the body of its designer, Rob Knight): the basic humanoid skeletal structure of the (headless) upper torso is clear. The articulated skeleton models many of the constraints and degrees of freedom of our own body. In addition, the musculature uses a mixture of passive compliance and series-elastic actuators, ensuring that the motor programs used by the robot will be similar in important ways to those used in our own brains. When complete, the torso will be mounted on a wheeled base.

Most autonomous mobile robots merely move through environments, but Cronos is being designed to be able to operate on the environment in ways comparable to those used by humans. At full extension, the arm and hand will be able

Owen Holland, University of Essex

Continued on p. 1

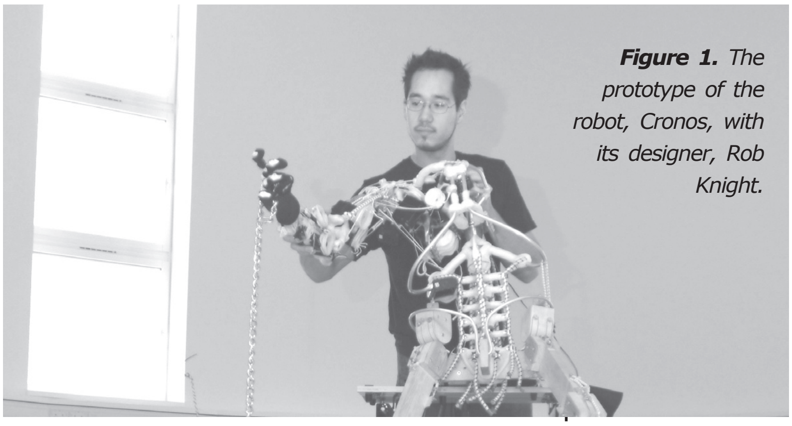

Figure 1. *The prototype of the robot, Cronos, with its designer, Rob Knight.*