How a Modeler’s Conception of Rewards Influences a Model’s Behavior: Investigating ACT-R 6’s Utility Learning Mechanism

Christian P. Janssen, Wayne D. Gray and Michael J. Schoelles

cjanssen@ai.rug.nl, grayw@rpi.edu, schoem@rpi.edu

1: Department of Artificial Intelligence, University of Groningen
2: Cognitive Science Department, Rensselaer Polytechnic Institute

Temporal difference learning has recently been introduced as the new utility learning mechanism in ACT-R 6 (e.g., Fu & Anderson, 2004). Common practices for using it still have to emerge. In this study we take a first step by investigating two critical aspects of utility learning: the location and size of rewards.

As a case study we use the Blocks World task (Gray et al., 2006). In this task subjects have to copy a pattern of eight blocks, depicted in a target window, by moving blocks from a resource window to a workspace window. Information in each of the windows is covered by a gray rectangle and only becomes available when subjects move the mouse cursor into the window area. In addition, the information in the target window only becomes available after waiting for a lockout time of 0, 400 or 3200 milliseconds (manipulated between subjects). As the size of the lockout time increases, subjects tend to study and place more blocks per visit to the target window.

Previous attempts in modeling the task in ACT-R 5 did not provide good fits to human data. Analysis indicated that this might be because ACT-R 5’s expected value equation can only handle binary feedback (Gray, Schoelles, & Sims, 2005). As ACT-R 6’s utility learning mechanism is not limited to binary feedback, its use seems more promising.

ACT-R 6 Models of the Blocks World task

The ACT-R 6 models of the Blocks World task are kept close to the ACT-R 5 models, but also take benefit of new ACT-R 6 features. Crucially, the model has eight encode-x productions that determine its strategy: the number of blocks (x, ranging between 1 and 8), which the model will study during a visit to the target window. Using utility learning, the model tries to learn which encode-x strategies/productions lead to the best overall performance.

We tested six different models that have the same parameter settings and production rules, but differ in two aspects: the location of the reward and the size of the reward. The location of the reward can either be once per trial after completing the whole trial or each time that the model has tried to place blocks in the workspace window and either finishes the trial, or starts studying blocks in the target window again. The location of the reward is important for utility learning, as the utility of a production converges towards the size of the experienced reward minus the average time between the firing of that production and the time the reward was triggered (Anderson, 2007).

The second manipulation between models is the size of the rewards. Due to space limitations we will not describe each of these models in detail, but fundamentally the manipulations differ in what aspect of the task the model conceptualizes as a reward. On the one hand, a reward can be expressed in how good the model performs the task itself: how many blocks does it place after a visit to the target window, and how many blocks does it study but forget? Different models have different reward functions, but in general the rewards range between -8 (all blocks studied, none placed) and 8 (all blocks placed). A totally different conception is to express rewards in terms of how fast the model performs the task (or specific parts of it). Different models have different reward functions, but in general the rewards are negative: the more time the model spends on the trial (or on specific parts of it), the more negative the reward is. In this case rewards range between 0 and about -80.

Results and Discussion

As shown in our example above, the modeler’s conception of the rewards of a task has a big influence on the reward size. The reward size has a big influence on the utility of productions, and this has a big influence on the behavior of a model. In the end, the modeler’s conception of rewards has a big influence on the model’s behavior.

In our simulations of the Blocks World task, each model behaves different from others. Despite the broad exploration of model types, none provides a good fit to the human data. Some models seem to be at least as good as the best ACT-R 5 models (Gray et al., 2005). Unlike the ACT-R 5 models, these models do not require changes to the architecture. They require a different conception of what a reward is.

References


