Top-Down versus Bottom-Up Control of Cognition in a Task Switching Paradigm

Michael J. Schoelles (schoem@rpi.edu)
&
Wayne D. Gray (grayw@rpi.edu)
The CogWorks Lab; Cognitive Science Department
Rensselaer Polytechnic Institute; Troy, NY 12180 USA

Introduction

In this poster we explore the ways in which differences in a cognitive architecture facilitate different approaches to modeling task switching. In a predominately goal-driven, or top-down, architecture (ACT-R 4.0, see Anderson & Lebiere, 1998), a goal is pushed onto the goal stack and continues to control cognition until it is popped off the goal stack. At the point where the goal for one task has been popped, it is easy to model a conflict between two competing productions, one of which pushes the goal for task A and the other of which pushes the goal for task B. This competition between goals may follow the firing of a series of productions that explicitly check various states in the environment (such as scores on task A and task B). Hence, the urgency of task A versus task B may change dynamically to reflect the current state of the task environment.

Goal-driven architectures are notoriously insensitive to stimulus driven or bottom-up changes. Indeed, the goal-hierarchy of ACT-R 4.0 precluded unplanned interruptions. Although it is possible to build parallel productions that pop a given goal if a given environmental contingency occurs, this type of flexibility seems more like an explicitly acquired top-down strategy than a bottom-up approach. More fundamentally, unless the modeler explicitly constructs a complete set of parallel productions, such architectures are effectively deaf and blind to any and all events that are not specifically part of the currently attended task.

ACT-R 5.0 (Anderson, Bothell, Byrne, & Lebiere, 2002) preserves ACT-R 4.0's top-down mechanisms while expanding the range of bottom-up ones. First and most fundamentally, environmental changes may automatically affect ACT-R's perceptual buffers. For example, when an auditory event occurs, it occupies the auditory location buffer. This buffer contains the information that an auditory event (i.e., a sound) has occurred, but it does not contain the meaning of that sound. (For meaning to be processed, auditory attention must be directed to the auditory location. Only then is the meaning of this event placed in the aural state buffer where it is available to cognition.)

The aural location buffer has a flag that indicates whether the sound it contains has been attended to or not. The process of filling the auditory location buffer and setting the buffer's flag to nil is assumed to result from purely perceptual processes that do not involve central cognition.

A second bottom-up approach is provided by ACT-R 5.0's elimination of the requirement that every production have a goal. Productions can be written that specify conditions for various buffers, but that do not require a specific goal. Such productions are extremely general and can compete with goal-specific productions whenever their conditions are met, regardless of the current goal.

We believe that these changes in the architecture provide an interesting way of combining top-down and bottom-up processes in the control of task switching. Below we introduce the Argus Prime (Schoelles & Gray, 2001) simulated task environment (Gray, 2002) and discuss our most recent empirical study using Argus Prime. We then briefly discuss our implementation of task switching and present results that compare top-down, bottom-up, or both top-down and bottom-up task switching with human data.

Argus Prime

Argus Prime is a classification task. It entails dynamic decision-making in the context of a radar-tracking task that requires the interaction of human cognitive, visual, and motor operations. The classification task interface depicts radar console for tracking airborne targets.

The subject selects a target by moving the cursor to its icon and clicking. When a target is selected, the attributes of the target appear in an information window. The subject must convert the attribute data to a 1-7 threat value. The subject is taught an algorithm to do this.

Summative feedback is given by a percentage. This percentage is updated each time a target crosses a sector boundary. It represents cumulative performance over all targets. Immediate feedback for each target classification is given in some experimental conditions.

In addition to the classification task, Argus Prime can include an additional (dual) task. In the current study, we used the Alpha task.

For the Alpha task the system says a letter of the alphabet once every four sec. Subjects respond by pressing the "x" key if the current letter (n) occurs earlier in the alphabet than the immediately prior letter (n-1) or the "c" key if it occurs later. Overall performance is given as a percentage of the number correct to the total number of opportunities. This score is updated continuously and displayed, on the menu bar, to the right of the classification score.

Dual Task Model

A single-task Argus Prime model was developed in ACT-R 5.0 and was tuned to the data obtained from an earlier Argus Prime study (i.e., not from the study reported here). This model is of the classification task only. It does about as well as humans do and accurately mimics human performance across four interface conditions (for more information, see, Schoelles, 2002).

A dual-task model was implemented by modifying the single-task Argus Prime model to account for top-down and bottom-up task switching. Top-down task switching is initiated in two ways. First, after classifying one target, the goal of classifying another target competes with the goal of doing the Alpha task. Second, while classifying a target, the model will check the scores on both the classification and Alpha tasks. If the difference is large enough (controlled by a parameter) then the model will switch to the Alpha task.

Bottom-up task switching to the Alpha task is initiated by a production that specifies conditions on the aural location buffer and aural state buffer but does not specify a condition on the goal buffer. The specified conditions are that the Audition Module is not currently active and there is an audio event that has not been previously attended. The aural location buffer contains an audio event as a result of the system saying a letter.

When this production executes, it requests the Audition Module to encode the sound event (i.e. focus aural attention on the event). It also sets a goal to perform the Alpha task. In addition, it clears the aural location buffer so that the same production will not match on the next cycle.

The bottom-up production also checks that the current task is not the Alpha task, since there is no need to switch to it if it is already being performed. A different solution might be an architecturally based mechanism to lock out certain interrupts.

Results

Figure 1 compares the mean number of task switches for 24 humans against task switches for each of three versions of the model. Model-top-down executes only the top-down strategy. Model-bottom-up executes only the bottom-up strategy. Model-both is the model executing both a top-down and bottom-up task switching strategy. As can be seen, implementing both strategies greatly improves the model performance.

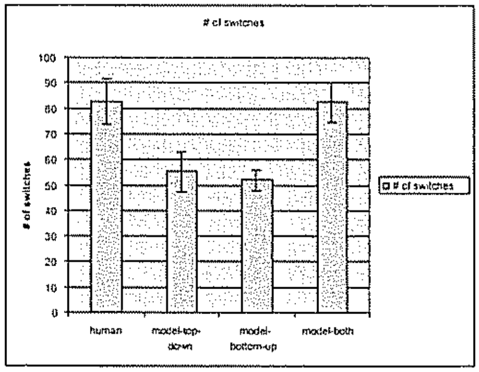

Figure 1: Comparing # of task switches.

Conclusions and Discussions

Our preliminary results suggest that purely goal-driven architectures (such as ACT-R 4) cannot adequately account for task switching. In contrast, architectures that provide a role for bottom-up as well as top-down processes (such as ACT-R 5) can. Cognitive modeling allows us to explore the tension between top-down and bottom-up strategies, and to explore how the mix of such strategies is affected by subtle changes in the design of the task environment. At ICCM-5 we will discuss the limits and success of our efforts in more detail.

Acknowledgements

This work was supported by Air Force Office of Scientific Research Grant F49620-97-1-0353 and Office of Naval Research grant N000140310046.

References

Anderson, J. R., Bothell, D., Byrne, M. D., & Lebiere, C. (2002). *An integrated theory of the mind*. Retrieved October 17, 2002, from http://act-r.psy.cmu.edu/papers/403/IntegratedTheory.pdf

Anderson, J. R., & Lebiere, C. (Eds.). (1998). *Atomic components of thought*. Hillsdale, NJ: Lawrence Erlbaum Associates.

Gray, W. D. (2002). Simulated task environments: The role of high-fidelity simulations, scaled worlds, synthetic environments, and microworlds in basic and applied cognitive research. *Cognitive Science Quarterly, 2*(2), 205-227.

Schoelles, M. J. (2002). *Simulating Human Users in Dynamic Environments*. Unpublished doctoral dissertation, George Mason University, Fairfax, VA.

Schoelles, M. J., & Gray, W. D. (2001). Argus: A suite of tools for research in complex cognition. *Behavior Research Methods, Instruments, & Computers, 33*(2), 130-140.