Probing the Paradox of the Active User: Asymmetrical Transfer May Produce Stable, Suboptimal Performance

Wayne D. Gray, V. Daniel Veksler, & Wai-Tat Fu

Carnegie Mellon University
Cognitive Science Department
[grayw, vekslv] @rpi.edu

The “paradox of the active user” (Carroll & Rosson, 1987) is the persistent use of inefficient procedures in interactive environments by experienced or even expert users when demonstrably more efficient procedures exist. In this study we examine the procedures that people adopt in response to minor changes in interface design and how these procedures adapt, or do not adapt, when the design changes.

Prior Knowledge versus Perceptual-motor Effort

For this work, subjects programmed one of two nearly identical VCR simulations. To program a setting, the subject must first click on the setting’s radio button and then use the up or down arrow to reach the target value. As suggested by Figure 1, to set the start time the subjects must set start-hour, start-10min, and start-min. To set the end time, they set end-hour, end-10min, and end-min. In the interface used here the three radio buttons to set start time are in one row and the three buttons to set end time are in another row.

In this study we manipulated whether or not the interface had buttons above each column of radio buttons. For the button (BTN) condition (shown in the Figure 1), if a subject had just finished programming, for example, start-hour and now wished to program start-10min, they would first have to deselect the current column button and then select the next column button before they could click-on the “Start-10min” radio button. In the no button (noBTN) condition, subjects could select any radio button at any time.

How easy is it to manipulate the decomposition of the task-to-device rule hierarchy for a particular device? How persistent would the influence of practice in one task environment (either BTN or noBTN) be when subjects were transferred to the other task environment?

We hypothesized that due to the role of prior knowledge, subjects in the noBTN condition would adopt a by-row strategy in which they would program all of start time (hour, 10min, and min) before going off to program something else. In contrast, the BTN condition increases the perceptual-motor cost of this “natural” by-row strategy by requiring subjects to click column buttons on and off prior to selecting a radio button in another column. Consequently, the BTN interface would seem to encourage an unnatural by-column strategy (i.e., setting start- and end-hour, then start- and end-10min, and then start- and end-min).

Strategies Adopted on BTN and noBTN Interfaces

Each subject programmed 8 different shows to the criterion of two successful trials per show. Half of the subjects (32) programmed the first four shows with BTN and half with noBTN. For the last four shows they switched interface conditions. The dependent variable in this report was the strategy used by subjects to program time; either by-row or by-column.

By trial 4, 30/32 noBTN(tr4) subjects used the by-row strategy. In contrast, 21/32 BTN(tr4) subjects used the by-column strategy. A very simple manipulation of the perceptual-motor cost resulted in dropping a strategy that was congruent with prior knowledge for a strategy that in some way ran contrary to prior knowledge.

Moreover, further analysis revealed that such minor interface differences affect effort and performance, as well. The average time spent memorizing the target values on the hard interface (BTN) was ~10.4% higher than on the easy interface (noBTN). This resulted in ~20% less programming errors on the hard interface than on the easy.

Transfer from BTN to noBTN and vice versa

The results for trial 8 (see Table 1) show that when transferring from a hard interface (BTN) to an easy interface (noBTN), the methods acquired with the hard interface persist. However, in going from easy to hard, subjects quickly adapt to the hard interface.

From the subject’s perspective, during the second phase of the study (shows 5-8) the perceptual–motor cost of the by-column strategy greatly decreased and the memory cost stayed about the same. Hence, a strategy that worked well under the conditions of the BTN task environment still worked and was easier to implement under the noBTN task environment.

Summary

People do not use one tool or one piece of software. Rather, we work in multiple task environments and our cognitive processes seem adapted to these environments, as opposed to particular subtasks in any given environment.

Acknowledgments

The research and writing was supported by a grant from the Office of Naval Research ONR #N000140310046.

References