Mechanisms of Slips in Display-Based Human-Computer Interaction: A Model-Based Analysis

Muneo Kitajima[a] and Peter G. Polson[b]

[a]National Institute of Bioscience and Human-Technology, 1-1 Higashi Tsukuba Ibaraki 305, JAPAN

[b]Institute of Cognitive Science, University of Colorado, Boulder, CO 80309-0344, U.S.A.

1. ERRORS OF SKILLED USERS IN HUMAN-COMPUTER INTERACTION

A puzzling and frequently-ignored fact in the human-computer interaction literature is that skilled users have surprisingly high error rates (10 ~ 15%). Card, Moran and Newell (1983) studied individual skilled users performing two tasks, manuscript editing and electronic circuit design editing. The manuscript editing experiment involved a detailed evaluation of a single expert user doing 70 edits presented in marked up manuscript. Errors were made on 37% of the command sequences describing edits. Over half of the errors were detected and corrected during generation of the editing commands. 21% (15 out of 70) of the commands issued by this very skilled user generated the wrong result and required additional edits to correct these errors. In a second study of a single expert carrying out an electronic circuit design editing task, the user had an error rate of 14% on 106 edits. Hanson, Kraut, and Farber (1987) studied 16 researchers and managers who were intermediate and expert level users of UNIX performing document preparation tasks and e-mail. They logged over 10,000 commands. The overall error rate was 10% with error rates ranging from 3% to 50% on different commands.

The experiments briefly reviewed here are representative of results from a wide range of studies in the human-computer interaction literature. Error rates for expert users range from 5 to 20%. In all studies of experts, users eventually produced the correct results. Approximately 50% of the errors are detected during the generation of a command and corrected. Detection and correction of errors is an integral part of expert skill.

The literature on errors has concluded that there are two qualitatively different types of errors (Norman, 1981; Reason, 1990). The first is errors of commission, or mistakes. Such errors are committed by users who are carrying out a novel tasks and fail to immediately discover the correct action sequence. The other is slips, where users have the correct intention but fail to successfully execute the correct action sequence. Most part of errors described above is slips.

Sellen (1990) reviews classes of models that provide principled, qualitative accounts for slips. She argues that all of these models have a hierarchical representation of action sequences that include representations of top-level task goals and lower-level goals that actually control execution of elementary actions. Reason (1990) argues that control of attention is a critical determinant for generating correct performance from a hierarchical representation of action sequences that include representations of top-level task goals and lower-level goals that actually control execution of elementary actions. Failure to adequately attend to the ongoing process and coordinate the interaction between the various schema causes a wrong low-level schema to become activated, generating related but incorrect actions for the current task. In HCI tasks, the users could be focusing on the task of composing new text or drawing a figure, and so on. This would lead to insufficient attention being allocated to subtasks involved in operating the interface.

Card, et al. (1983) proposed that experts accept high error rates in order to increase their productivity, because for them error recovery can be done easily and rapidly. Experts trade speed for accuracy, causing slips.

In this paper, two mechanisms of slips, *attention failures*, and *speed-accuracy tradeoffs* are simulated by a comprehension-based cognitive model of display-based human-computer interaction proposed by Kitajima and Polson (1992, 1994a, to appear), showing that they could account for the rate of slips made by skilled users interacting with graphical user interfaces (Kitajima and Polson, 1994a, 1994b).

2. A MODEL OF DISPLAY-BASED HCI

The model developed by us (Kitajima and Polson, 1992, 1994a, to appear) is shown in Figure 1. The model elaborates Hutchins, Holland, and Norman's (1986) action theory framework which consists of the following four basic components:

(1) goals representing what the user wants to accomplish which are a schematic outline of the action sequence that will accomplish the task,

(2) *a task environment* which is the world that reacts to the user's actions and generates new responses by modifying the display,

(3) *the stage of evaluation* comprised of the processes that evaluate and interpret the display, and

(4) *the stage of execution* comprised of the processes that select and execute actions that affect the world.

Our model of the Hutchins, Holland, and Norman's (1986) action theory incorporates goals, two processes for the stage of evaluation and two for the stage of execution.

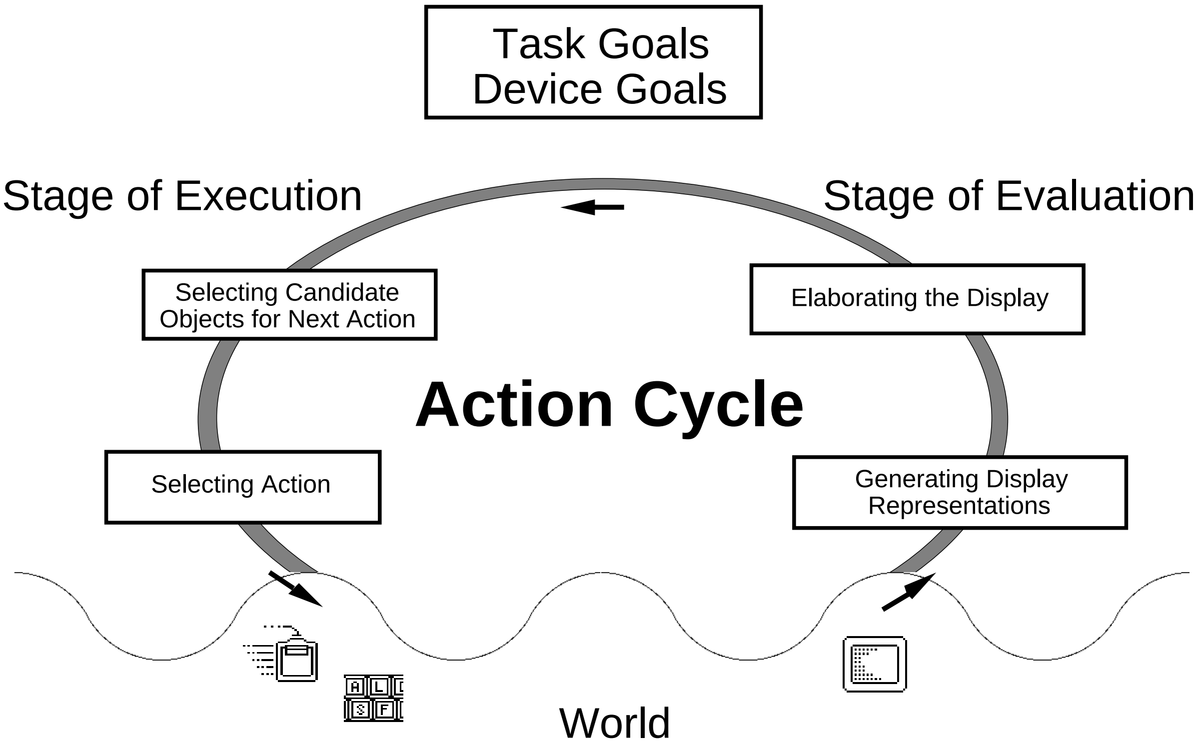

Figure 1. The comprehension-based cognitive model of skilled use of graphical user interfaces, mapped onto Hutchins, Hollan, & Norman's (1986) action cycle.

2.1. Task Goal and Device Goal

The model assumes that skilled users have a schematic representation of the task that is in the form of a hierarchical structure involving two kinds of goals: *task goals* and *device goals*. Our goal representation is taken directly from the Yoked State Space Hypothesis proposed by Payne, Squibb, and Howes (1990). Payne, et al. assume that discovering how to carry out a task involves searching of two problem spaces. The first is a space of possible task states. The second is a space of possible device states that are required to achieve a given task state. We assume that each task goal is associated with one or more device goals. The device goals specify device states that must be achieved in order to satisfy an associated task goal.

Given a task goal and its associated device goals, the model simulates a sequence of action selections as follows.

2.2. Generating Display Representations

At first, the model generates a representation of the display. The display representation only includes information about identity of each object on the display and its appearance, e.g. highlighted, pointed-at, dragged, etc. *No* information about what actions can be taken on the object, or its meaning and relationships to other objects in the display is included in this initial display representation.

2.3. Elaborating the Display

All such information is generated by the elaboration process which retrieves information from long-term memory by *a random memory sampling process*. The retrieval cues are the representations of the current display, the task goal and the device goals. The probability that each cue retrieves particular information in a single memory retrieval process is proportional to the strength of the link between them. The model carries out multiple memory retrieval in a single elaboration process. A parameter, *the elaboration parameter*, controls the number of times each argument in the display and goal representations is used as retrieval cues[1].

The retrieved information elaborates the display representation, providing information about interrelationships between display objects, relationships between the task and display objects, and other attributes of display objects. *The elaborated display representation* is model's evaluation of the current display in the context defined by the task goal and the device goals.

2.4. Selecting Candidate Objects for Next Action

In the stage of execution, the model first limits its attention to a few number of screen objects out of ~100 objects displayed on the screen. These screen objects are candidates for the next action to be operated upon. The candidate object selection is performed on the basis of the evaluation, defined by the elaborated display representation. The model uses the spreading activation mechanism to select candidate objects. The process is dominated by two factors: the strengths of links from the representation of the goals, which is parametrized by a parameter, *the attention parameter*, and the number of propositions that are necessary to bridge the goals and the candidate objects[2].

---

[1]The model represents goals and display in propositions, like (is-on-screen OBJECT12). In the memory sampling process, the argument, such as OBJECT12, is used to retrieve information from long-term memory that has OBJECT12 as its argument.

[2]The model assumes the argument overlap mechanism to link up propositions. For example, the two propositions, (is-on-screen OBJECT12) and (has OBJECT12 CalculatorMenuItem), are linked by the shared argument, OBJECT12.

2.5. Selecting Action

The model considers all possible actions on each candidate object. The model incorporates 18 possible actions[3], such as "moving the mouse cursor to a menu item in order to display a pull-down menu." The process is dominated by the same two factors described above.

Furthermore, the action representations include conditions to be satisfied for their execution. The conditions are matched against the elaborated display representations. Some conditions are satisfied by the current screen, others by information that was retrieved from long-term memory in the elaboration process. For example, the model cannot select an action to double click a document icon for editing unless the icon is currently pointed at by the mouse cursor and the information is available that the icon can be double clicked. Observe that if information about a necessary condition is missing from an elaborated display representation, the model cannot perform that action on the *incorrectly* described object.

3. HOW THE MODEL ACCOUNTS FOR ERRORS

In a set of experiments we conducted so far, where a graph drawing task was simulated, we found that the model could cause errors due to the following three reasons.

The first is that the process of selecting candidate objects for the next action fails to include the correct object on the list of candidate objects. The second possible cause of errors is that the correct action fails to become the highest activated action among executable actions. In the model's terms, these kinds of errors are ascribed to both or either of small values of the attention parameter (A), and /or missing bridging knowledge that had to be retrieved from long-term memory (B).

The third is that the elaboration process fails to incorporate all of the conditions for the correct action in the elaborated display representation. Low values of the elaboration parameter cause this error (C). Parameter values in the range of 12 to 20 caused the model to simulate error rates in the range of 10% to 20% (Kitajima and Polson, 1994, to appear). We argue that the elaboration parameter describes a speed-accuracy tradeoff process where low values of the parameter reduce the amount of time taken by the elaboration process.

4. COMPARISON WITH OTHER MODELS

The strength of the model is that the model generates correct actions as well as occasional errors without assuming a special set of mechanisms to produce erroneous actions. In this

---

[3]Representations of actions define different functions of single physical actions in many different contexts. For simulating a graph drawing task, the model defines eighteen cognitive actions on six physical actions; Move-Mouse-Cursor, Single-Click, Double-Click, Hold-Mouse-Button-Down, Release-Mouse-Button, and Type.

respect, the model is strikingly different from typical models of expert performance and error (Anderson, 1993; Reason, 1990; Card, et al., 1983). Typical models assume that skilled performance is mediated by detailed, large grain size action plans stored in long-term memory. Card, et al. (1983) refers to them as methods; Reason (1990) assumes that skilled performance is mediated by action schemata (Norman, 1981). Thus they have to be equipped with erroneous plans to generate errors. The grain size of action is much smaller in our model, at the level of individual pointing action. When the model makes an error, it has attempted to select a correct action based on incomplete knowledge, and/or insufficient attention. The incorrect action will be highly constrained by the user's current goals, the current state of the display, and the partial knowledge that was successfully retrieved from long-term memory. The candidate objects and the next action selected by a simulation are the model's best selections given the context represented by the elaborated display representation.

REFERENCES

Anderson, J. R. (1993). *Rules of the mind*. Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The Psychology of Human-Computer Interaction.* NJ: Lawrence Erlbaum Associates.

Hanson, S. J., Kraut, R. E., & Farber, J. M. (1984). Interface design and multivariate analysis of UNIX command use. *ACM Transactions on Office Information Systems,* **2**, 42-57.

Hutchins, E.L., Hollan, J.D., & Norman, D.A. (1986). Direct manipulation interfaces. In Norman, D.A. & Draper, S.W., Eds. User Centered System Design. Hillsdale, NJ: Lawrence Erlbaum Associates.

Kitajima, M. and Polson, P.G. (1992). A computational model of skilled use of a graphical user interface. *Proceedings of CHI'92 Conference on Human Factors in Computing Systems*. NY: ACM, pp. 241-249.

Kitajima, M. & Polson, P. G. (1994a). A comprehension-based model of correct performance and errors in skilled, display-based human-computer interaction. *ICS Technical Report*. 94-02. Boulder, CO: Institute of Cognitive Science, University of Colorado.

Kitajima, M. and Polson, P.G. (1994b). A model-based analysis of errors in display-based HCI. *Conference Companion CHI'94*. pp. 301-302.

Kitajima, M. & Polson, P. G. (to appear). A comprehension-based model of correct performance and errors in skilled, display-based human-computer interaction. *International Journal of Human-Computer Studies.*

Norman, D. A. (1981). Categorization of action slips. *Psychological Review,* **88**, 1-15.

Payne, S. J., Squibb, H. R., & Howes, A. (1990). The nature of device models: The yoked state hypothesis and some experiments with text editors. *Human-Computer Interaction,* **5**, 4, 415-444.

Reason, J. (1990). *Human Error*. New York, NY: Cambridge University Press.

Sellen, A. *Mechanism of human error and human error detection.* Unpublished doctoral dissertation (1990).