N96- 14987

The Effects of a Dynamic Graphical Model During Simulation-Based Training of Console Operation Skill

John D. Farquhar
Department of Instructional Technology
University of Georgia

J. Wesley Regian
Technical Training Research Division
Air Force Armstrong Laboratory

ABSTRACT

LOADER is a Windows-based simulation of a complex procedural task. The task requires subjects to execute long sequences of console-operation actions (e.g., button presses, switch actuations, dial rotations) to accomplish specific goals. The LOADER interface is a graphical computer-simulated console which controls railroad cars, tracks, and cranes in a fictitious railroad yard. We hypothesized that acquisition of LOADER performance skill would be supported by the representation of a dynamic graphical model linking console actions to goal and goal states in the "railroad yard." Twenty-nine subjects were randomly assigned to one of two treatments (i.e., dynamic model or no model). During training, both groups received identical text-based instruction in an instructional-window above the LOADER interface. One group, however, additionally saw a dynamic version of the bird's-eye view of the railroad yard. After training, both groups were tested under identical conditions. They were asked to perform the complete procedure without guidance and without access to either type of railroad yard representation. Results indicate that rather than becoming dependent on the animated rail yard model, subjects in the dynamic model condition apparently internalized the model, as evidenced by their performance after the model was removed.

INTRODUCTION

This paper is concerned with the communication and representation of knowledge that aid in the acquisition of console operation skill. Console operations, which are common in industrial and defense settings, require the execution of sequences of control-panel actions such as the pressing of a button or the rotation of a dial. Console operation skill is arguably constructed of various knowledge types or forms of knowledge. While general theories of cognition and learning have asserted that all knowledge can be described as either declarative or procedural (Anderson, 1983), theories specific to console-operation knowledge have made additional distinctions (Polson & Kieras, 1984; Card, Moran, & Newell, 1980).

One central component in the theories of console-operation knowledge is an element describing the goal of the operation. The study reported in this paper examined the effects of communicating various goal states of a console-operation task through a dynamic graphical model. The console-operation task performed by subjects was a computer-simulation called LOADER. The LOADER interface is a graphical computer-simulated console which controls railroad cars, tracks, and cranes in a fictitious railroad yard. LOADER is designed to be a laboratory analog of procedural console operations and process control tasks. The advantage of using the LOADER simulation for this study was the capability to represent goal states in a dynamic visual display.

CONSOLE OPERATION KNOWLEDGE

The theory of console-operation knowledge described by Polson and Kieras (1984) classifies that knowledge into two components: the *job-task representation,* and *how-it-works knowledge.* How-it-works knowledge is knowledge of the underlying structure or processes of a device. The job-task representation includes *job-situation knowledge* (knowledge of tasks that the device can perform) and *how-to-do-it knowledge* (knowledge of how to carry out those tasks).

Overall, the job-task representation describes the relationship between the capabilities of the device and the processes for carrying out those tasks. A specific job goal (or capability) will identify an appropriate job-selection rule (or process) that should be performed. Additionally, how-it-works knowledge will aid the selection of appropriate rules when those rules can be inferred from the internal structure of a device (Kieras & Bovair, 1984).

Using the LOADER simulation as an example, a specific goal might be the positioning of a rail car to the appropriate dock. This goal identifies a sequence of control panel operations such as enabling control of the rail lines and switching the correct track. The LOADER interface displays the outcomes of the action with a bird's-eye view of the rail yard.

The GOMS model (Card, Moran, & Newell, 1980) asserts four categories of device-operation knowledge: (1) Goals, (2) Operations, (3) Methods, and (4) Selection rules. Goals and subgoals define the outcomes of a task. Operations are the individual actions that are performed by the user and the device to reach the goals and subgoals. Methods are the procedures, or sets of operations, performed in achieving each subgoal. Selection rules assert appropriate methods as determined by the subgoal.

Using the LOADER example again, the goal of loading a particular rail car will be made up of various subgoals (e.g. positioning rail cars, lifting and loading canisters). Each action performed on the way to achieving the goal, such as an individual switch setting, is an operation. A set of operations, that when performed sequentially achieve a goal or subgoal, is a method. Selection rules include information about goals and methods in order to match appropriate methods with the goals they achieve. These "rules of selection" determine methods from asserted goals.

Selection rules can be formalized as a *production rule.* Production rules, or productions, are condition-action pairs often represented in the form: **if** condition, **then** action (Anderson, 1983). Within LOADER, the goal of positioning a rail car from rail line #1 to rail dock #3 would assert the set of operations that would include Enabling Rail Car Control, Setting Rail Dock #3, and Engaging Rail Car South. The LOADER interface displays the initial and final arrangement of the rail cars through a dynamic graphical model.

The use of a graphical model in the acquisition of console and device operation has been found to be beneficial when the model addresses knowledge of the internal operations of the device (or how-it-works knowledge). Kieras and Bovair (1984) found that a *device model,* affected the acquisition of device operation skills by increasing the speed of acquisition, accuracy of retention, and speed of performance. While this model described the **internal** representation of device mechanisms, the study reported here investigated the use of an **external** representation of device goals and goal states.

Additional research in the area of knowledge communication and representation of console and procedural skill has found that some types of knowledge are less beneficial. In a study by Reder, Charney, and Morgan (1986), certain types of instructional elaborations facilitated the acquisition of procedural computer skills while others did not. Syntactic elaborations (i.e. how-its-done knowledge) improved learning outcomes, whereas conceptual elaborations (i.e. what-it-is knowledge) produced little effect.

Conclusions from previous research indicates that the type of knowledge communicated affects the performance outcomes in the acquisition of console operation skill. Additionally, the presentation of a device model which represents internal device mechanism aids the acquisition of procedural skill associated with console operation. The study described below was concerned with the effects of providing a meaningful model of goals and goal states within the context of interactive practice sessions.

METHOD

Design

The independent variable under investigation was the availability of a dynamic graphical model during acquisition of the task. The dynamic graphical model display, centered at the top of the screen, provided a birds-eye view of the loading task which dynamically responded to control panel actions. Animated sequences demonstrated realistic responses of crane-console interactions. During the practice trials of the task, the model was displayed to one of the two treatment groups. The immediately following retention test, however, required that both groups perform the task without access to the model. Subjects were randomly assigned to one of the two treatments (model and no-model groups).

Subjects

Twenty-nine subjects supplied through a temporary employment agency were paid for their participation in the study. Selection was limited to high-school graduates between the ages of 18 and 27 years. One subject was excluded from failure to complete the final test trial. Subjects completing the experiment early were given additional assignments.

Materials

A console operations task was designed to loosely represent a real-world task in remote crane control arm operation. The task (LOADER) involves the operation of a console to lift, transport, and load canisters from storage bins into rail cars. While skilled console operators follow an interacting set of proscribed procedures, a single 51-step procedure was selected for this experiment.

To provide instruction, practice, and testing for the task, a computer simulation was developed. The simulation displays a complete control panel consisting of a set of 29 interrelated buttons, knobs, and switches. The controls respond to clicks of a mouse with numerous switch and knob settings, meter readings, indicator lights, and an occasional beep. All panel components are appropriately labeled and organized by function on the lower half of the computer screen (see figure 1).

The upper half of the computer screen is reserved for on-screen messages, additional user options (i.e. "Assistance"), and the dynamic graphical display. On-screen messages include the practice or test trial number, a description of the step to perform (e.g. "Set Crane Control to 'Enable'"), and feedback messages for incorrect performance (e.g. "Incorrect. Click as indicated by the arrow.").

Selecting the "Assistance" option reveals a large red arrow indicating the location of the next button or switch, or knob position. An incorrect action in the procedure would automatically select the assistance option and display the red arrow.

The simulation software was developed using the ToolBook programming environment and was delivered via a laboratory of 22 individual computer stations. The stations were equipped with Compaq 486/33L microprocessors, VGA-quality monitors, keyboards, and right-handed, three-button mice.

Procedure

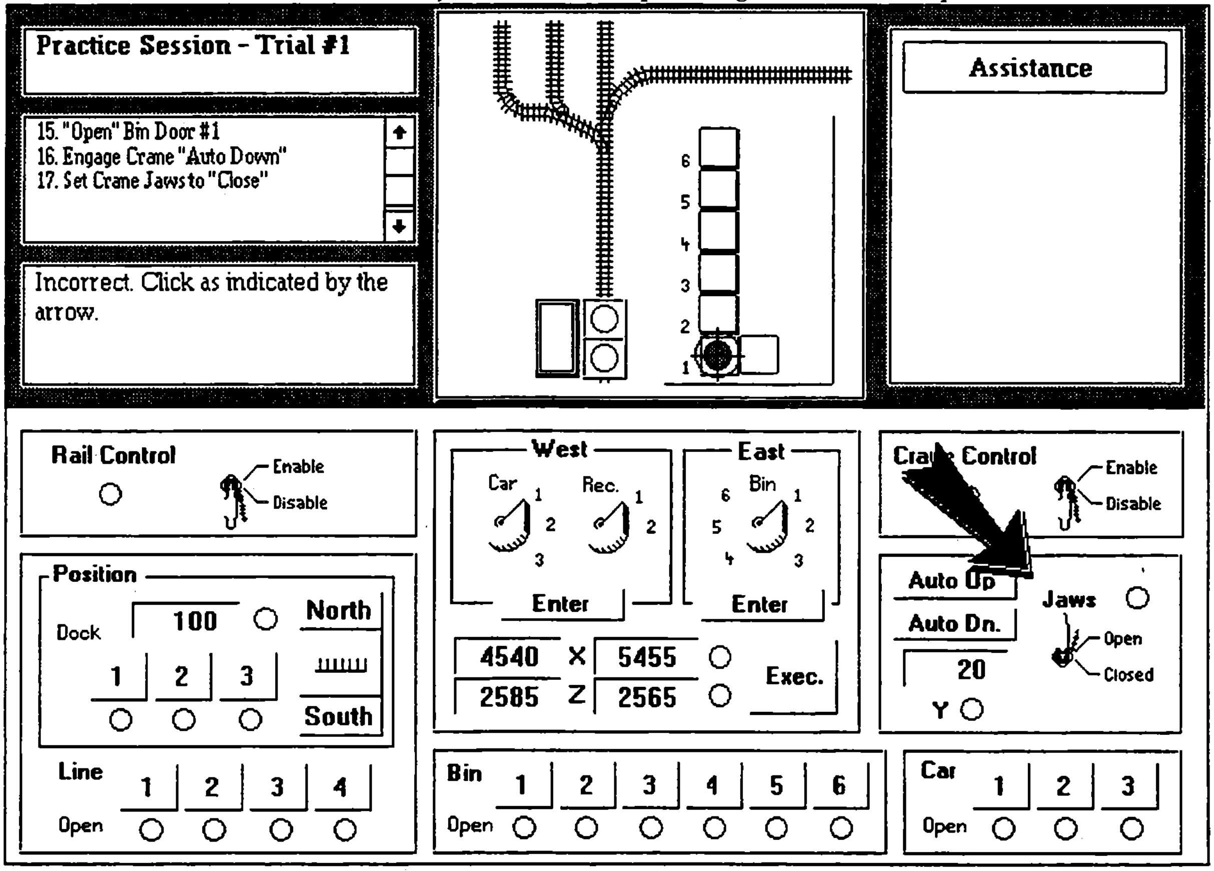
Subjects were brought into the laboratory in groups of 15 individuals. One of two treatments was randomly selected for the group. After a brief orientation in use of the equipment, the subjects were instructed to proceed through the program at their own pace. The program included three phases: an instructional phase, a practice phase, and a testing phase.

The instructional phase provided an introduction to the task of operating the crane control arm. This brief tutorial introduced and explained the task of crane control operations through a dynamic graphical model or display. The graphical model illustrated how the actions of the crane might appear as if the situation was viewed through a window. The instruction did not provide any specific instructions in crane operation, nor did it show or mention the related control panel. Finally, this phase concluded with directions of how the user should interact with the system during the practice and test phases.

During the practice phase, subjects completed 6 trials of the crane control procedure. A text prompt appeared on the screen to direct the subject in performing the individual steps. An on-screen "Assistance" option was available that when selected indicated the location of the next button or switch with a large red arrow.

In this practice phase, one treatment group had available the dynamic graphical model, an on-screen display identical to the graphical model presented in the instruction phase. During practice, however, this model represented crane arm-control panel interactions through dynamic changes of the display. The dynamic model was not present for the no-model group.

Following six practice trials, subjects entered the test phase. In this phase, subjects made three attempts at performing the crane control procedure without the aid of text prompts or assistance. Neither treatment group had available the dynamic graphical model during testing trials. If an error was made during testing, the red assistance arrow would direct the subject to the next step, forcing ultimate correct performance.



Practice Session - Trial #1

15. "Open" Bin Door #1
16. Engage Crane "Auto Down"
17. Set Crane Jaws to "Close"

Incorrect. Click as indicated by the arrow.

Assistance

Dock 100

4540 X 5455

2585 Z 2565

20

Figure 1 LOADER Interface.

RESULTS

Multiple analysis of variance on repeated measures were performed on completion times and errors. Mean completion times and errors for the nine trials (six practice, three test) are shown in Figures 2 and 3 respectively. Nearly equivalent completion times and errors for the model and no-model groups were recorded for each of the six practice trials. Therefore, differences between times [F(1,27) = .018, p = .8929] and errors [F(1,27) = .708, p = .4074] were insignificant. Test trials, however, yielded significant effects between groups for both completion times, [F(1,27) = 12.33, p = .0016] and errors [F(1,27) = 21.342, p = .0001].

These results indicate the following:

- the model and no-model groups performed equivalently during the 6 practice sessions,
- the model group performed the task faster than the no-model group during testing,
- the model group performed fewer errors than the no-model group during testing.

DISCUSSION AND SUMMARY

The presence of a dynamic model representing the goals of the procedural task during the acquisition phase of a procedural skill appears not to slow the learning process. In addition, the presence of the model significantly improves the performance of the skill with regard to the time and errors made during a testing phase even though the testing phase consisted of performance without the presence of the graphical model. That is, in the process of learning to carry out a specific sequence of actions to accomplish a goal, the operator came to imagine the corresponding events in the railroad yard, even if the operator could not actually see the yard while operating the console.

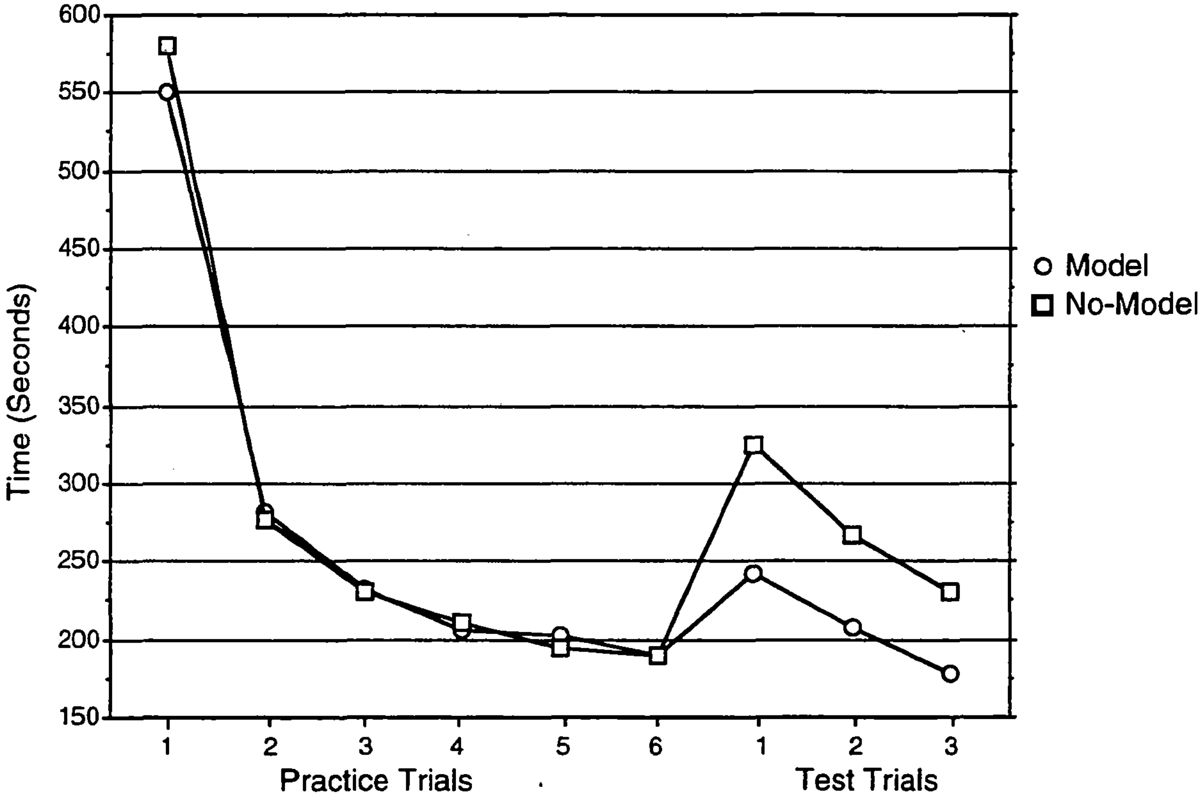


Figure 2 Time per trial.

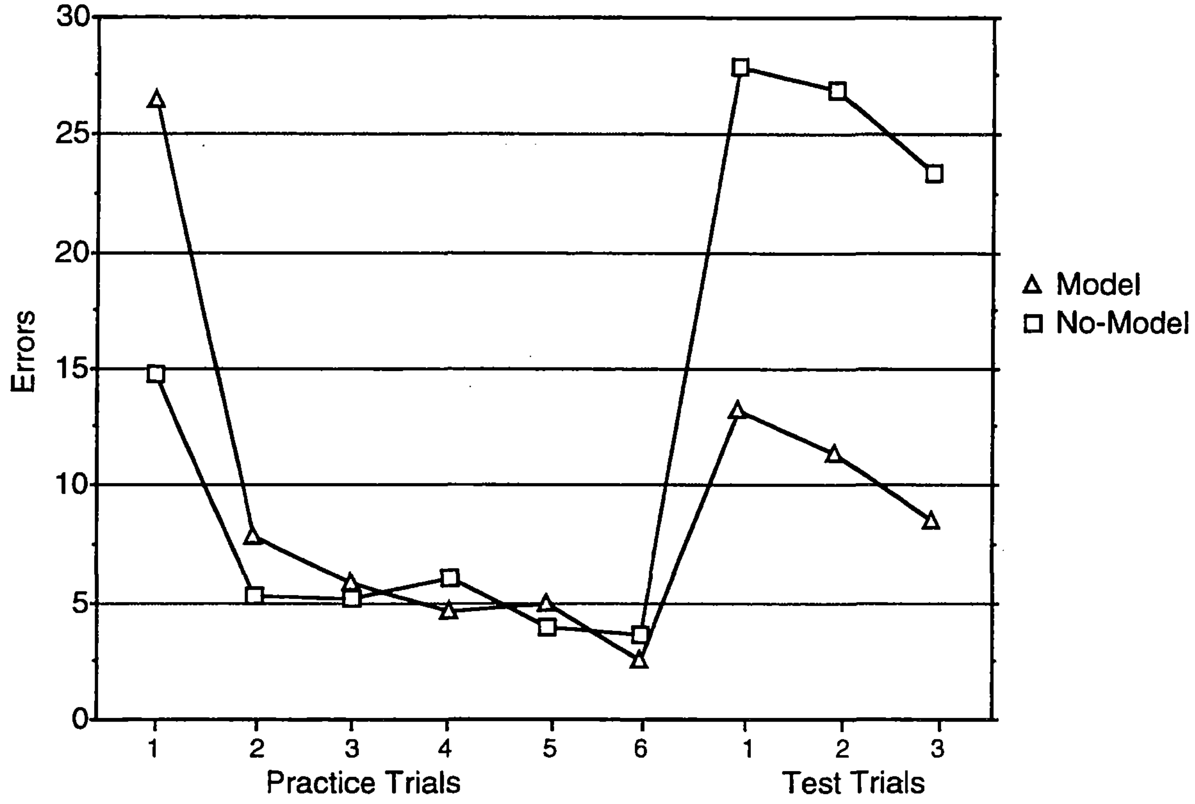


Figure 3 Errors per trial.

REFERENCES

Anderson, J.R. (1983). *The architecture of cognition.* Cambridge, MA: Harvard University Press.

Card, S.K., Moran, T.P., & Newell, A. (1980). Computer text editing: An information-processing analysis of a routine cognitive skill. *Cognitive Psychology,* 12, 32-74

Kieras, D.E., & Bovair, S. (1984). The role of a mental model in learning to operate a device. *Cognitive Science, 8*(3), 255-273

Polson, P.G. & Kieras, D.E. (1984). A formal description of users' knowledge of how to operate a device and user complexity. *Behavior Research Methods, Instruments, & Computers. 16*(2), 249-255

Reder, L.M., Charney, D.H., & Morgan, K.I. (1986). The role of elaborations in learning a skill from an instructional text. *Memory and Cognition, 14*(1), 64-78

AUTHORS

John D. Farquhar is a doctoral student of Instructional Technology at the University of Georgia.

J. Wesley Regian is Senior Scientist with the Technical Training Research Division at the Air Force Armstrong Laboratory. Address correspondence to: John D. Farquhar, Dept. of Instructional Technology, 630 Aderhold, University of Georgia, Athens, Georgia, 30602, or e-mail: johnf@moe.coe.uga.edu.

Funds for this research were provided by the U.S. Air Force, Technical Training Research Division, Brooks Air Force Base, San Antonio, Texas and the Air Force Office of Scientific Research, Bolling Air Force Base, Washington, D.C.