S9-53

20209

N91-27097 15

TRAINING EFFECTIVENESS OF AN INTELLIGENT TUTORING SYSTEM FOR A PROPULSION CONSOLE TRAINER

Final Report

NASA/ASEE Summer Faculty Fellowship Program - 1990

Johnson Space Center

H 2086788

Prepared by:	Debra Steele Johnson, Ph.D.
Academic Rank:	Assistant Professor
University & Department:	University of Houston Department of Psychology Houston, Texas 77204

NASA/JSC	
Directorate:	Information Systems
Division:	
Branch:	Software Technology
JSC Colleague:	Robert T. Savely
Date Submitted:	August 10, 1990
Contract Number:	NGT-44-005-803

ABSTRACT

A formative evaluation was conducted on an Intelligent Tutoring System (ITS) developed for tasks performed on the Propulsion Console. The ITS, which was developed by AFHRL primarily as a research tool, provides training on use of the Manual Select Keyboard (MSK). Three subjects completed three phases of training using the ITS: declarative, speed, and automaticity training. Data was collected on several performance dimensions, including training time, number of trials performed in each training phase, and number of errors. Information was also collected regarding the user interface and content of training. Suggestions for refining the ITS are discussed. Further, future potential uses and limitations of the ITS are discussed. The results provide an initial demonstration of the effectiveness of the Propulsion Console ITS and indicate the potential benefits of this form of training tool for related tasks.

INTRODUCTION

Intelligent Tutoring Systems (ITS's) have been developed for a variety of tasks, ranging from geometry to LISP programming.[1] However, little systematic evaluation has been conducted on these training systems. Additional research is needed to systematically examine the effectiveness of ITS's both during (formative evaluation) and upon completion (summative evaluation) of software development.[2] Conducting a formative evaluation enables the developer to determine whether the tutor is operating as planned and make program modifications as necessary. A summative evaluation is focused on assessing the training effectiveness of a completed ITS.

The ITS under consideration is still under development. Thus, a formative evaluation was conducted to provide information on the functioning of the system and the effects of the ITS on student learning. Although there is not general agreement yet about which specific evaluation methods are preferred and how to implement them, the information collected at least partially addresses internal and external evaluation issues.[2] Internal evaluation addresses how the tutoring systems functions. External evaluation addresses the educational impact of the tutoring system on students. The primary focus of the current project is on external evaluation, although some information on internal evaluation is also provided. In addition, given that a formative evaluation was being conducted, the focus of data collection was more on process measures than outcome measures. Formative evaluations tend to rely more on process measures (e.g., patterns of task activities) rather than outcome measures (e.g., task performance upon completion of training).[3]

BACKGROUND ON PROPULSION CONSOLE ITS

An ITS is under development by AFHRL which simulates use of the Manual Select Keyboard (MSK) on the Propulsion Console used by flight controllers. The purpose of the ITS development project was to develop a tutoring system for a high performance task.[4] A high performance task is one in which the knowledge required is small, but extensive practice is required to proceduralize the set of skills involved. This type of task is often performed in situations in which high risk or expense is involved.[4] Thus, it may be difficult to provide extensive training in the actual work environment. An ITS provides students an opportunity to proceduralize a set of skills in a safe and relatively inexpensive environment.

The MSK was selected for the training domain because it represents a high performance task. Although little task information is required, extensive practice is required to automate performance. Flight controllers need to automate use of the MSK so that most or all of their attention is available for performing other important console tasks. In addition, the MSK ITS provides a demonstration of a training system that could be expanded to other Propulsion Console tasks, highlighting future potential training benefits for Propulsion flight controllers. Finally, the MSK ITS has implications for other flight controllers because the MSK is used not only on the Propulsion Console but also on other flight control consoles to perform similar tasks. Thus, this tutoring system has potential training benefits for flight controllers in general.

The MSK ITS includes a domain expert (i.e., an expert model), a trainee model, a training session manager, a scenario generator, and an user interface.[1] The domain expert includes information on how to perform the task. The trainee model includes a record of student performance. The training session manager provides information to the student on performance accuracy and speed. The training session manager also determines the amount and form of remediation to provide. The scenario generator provides variations of the task actions to the student. Finally, the user interface enables the student to interact with the system—for this ITS using a 3-key mouse and five function keys. (The function keys were only used during automaticity training.)

As mentioned above, the training session manager provides information on errors and determines the remediation. The error messages and remediation provided depend on the phase of training. Training is provided in three phases: declarative, speed, and automaticity training. In the declarative phase, task action steps are first described; guided examples are then provided, followed by unguided examples. Guided examples require students to complete an action step following a prompt. Unguided examples require students to complete all steps in an action without being prompted at each step. To complete declarative training, the student must correctly perform two consecutive guided examples, then two unguided examples. Speed training requires students to perform actions correctly and within a specified amount of time. Finally, automaticity training requires students to perform dual tasks correctly and at a specified speed. The primary task is the performance of MSK actions. The secondary task involves correctly responding to patterns of beeps. For both speed and automaticity training, training is completed when the student has accurately performed each task action twice and within a specified amount of time.

Error messages are provided immediately following an incorrect step during initial training (i.e., during guided examples) and following completion of a set of action steps during later training (i.e., during unguided examples, speed, and automaticity training). If an error is made during training, the student is remediated to the previous level of training. For example, if an error is made during speed training, the student is given an unguided example; if an error is made on an unguided example, s/he is provided a guided example to perform. In addition, the amount of tutoring content provided increases for successive occurrences of a given error during guided examples. This is consistent with recommendations made by other researchers.[5, 6] Remediation during automaticity training occurs only if speed and accuracy criteria are not met, rather than after each occurrence of an error.

METHOD

Task Overview

The MSK ITS trains students to perform five console operations: TV Channel (TV Chan), Display Request (Disp Req), Display Decoder Drive (DDD), Analog Event System (AES), and Flight Select (Flt Sel). In addition, two operations have variations. The AES operation includes: Select (AES Sel) and Deselect (AES Des). The DDD operation includes: Select (DDD Sel), Release (DDD Rel), Reset Operation (DDD Reset Op), Reset Critical (DDD Reset Crit), Select Drive (DDD Sel Drive), Select Datatype (DDD Sel Data), and Select Lamp (DDD Sel Lamp). Thus, the student learns a total of 12 task actions relating to 5 console operations. In the current project, the criterion for promoting students from speed training to automaticity training is two actions completed without error and in less than 20 seconds on each of the five console operations. The criterion for completion of automaticity training is two actions completed without error and in less than 40 seconds on each of the five console operations. Moreover, the actions must be completed with 100% accuracy on the secondary task (responding to beep patterns). Students were asked to respond to two target beep patterns (e.g., long-long-short, short-long-short) and not respond to false alarms (i.e., any of the remaining five beep patterns). Beep patterns were administered at 3-second intervals.

Subjects and Procedure

Three students completed training on the MSK ITS. Two students were flight controllers: one was a certified flight controller in on-board navigation with 3 years experience; one was a novice flight controller on the trajectory console with 6 months experience. Both were familiar with the MSK as used on their console, but unfamiliar with its use on the Propulsion console. The third student was a researcher in ITS's with no console experience. Students were asked to complete training on the ITS. All instructions were provided by the tutoring system. Additional informal observations and comments were collected from the students on ITS content, functioning, and the user interface.

Measures

Performance data was collected by the ITS. For Level 1 (Declarative) training, performance measures included number of trials and number of errors. For Level 2 (Speed) training, performance measures included number of trials, number of errors, number of successful trials, number of trials during remediation, and number of errors during remediation. A successful trial was operationalized as completing an action correctly in less than 20 seconds. For Level 3 (Automaticity) training, performance measures included number of trials, number of errors, number of successful trials, number of remediation cycles, number of trials during remediation, number of errors during remediations. A successful trial was operationalized as completing an action correctly in less than 40 seconds with 100% accuracy in responding to beep patterns. Two successful trials of each of the five operations was required to complete speed training and to complete automaticity training. Results will be reported by task action for Level 1 training, but across actions for Levels 2 and 3. In addition, performance data was collected by action type during Levels 2 and 3 training to assess performance speed.

RESULTS

For Level 1 (Declarative) training, the three students performed between 45 and 54 trials with between 6 and 9 errors, averaging 49 trials and 7.3 errors (see Table 1). Further, the students required between 90 and 100 minutes to complete Level 1 training. Moreover, each student completed additional training on DDD tasks beyond the two consecutive, correct trials. It is unclear why the ITS administered the additional task trials. In some cases the additional trials involved actions on which students had previously made no errors. Further information is needed to clarify this issue.

For Level 2 (Speed) training, the three students performed an average of 30 trials (see Table 2). Results are reported across task actions for this training level. Students required an average of 48.3 minutes to complete the training. During this time, students made 3.7 errors on average. Remediation trials were administered after each error. On average, students completed 21.7 trials during remediation, making an average of 3.7 errors during the remediation trials.

One interesting point is that the remediation provided following an error did not necessarily correspond to the action on which the error was made. For Flt Sel, TV Chan, and Disp Req, an error was followed by remediation trials on the same action. However, an error on AES Sel was often followed by remediation trials on AES Des. Similarly, an error on one of the 7 DDD actions was often followed by remediation trials on other DDD actions but not on the DDD action on which the error was made. It would seem more beneficial to provide remediation on the action on which the error was made.

Another interesting point is that students were not returned to speed training following two consecutive, correct actions although this was the criterion stated. For example, Student 1 correctly performed the Disp Req action 7 consecutive times and DDD Res action 5 times before returning to speed training. Student 2 correctly performed the Disp Req action 9 consecutive times before being returned to speed training and correctly performed the Disp Req action 3 times following a second error and the AES Sel action 3 times. Student 3 also correctly performed actions 3 to 4 consecutive times during remediation trials before returning to speed training. Additional information is needed on the decision rules used by the ITS for providing remediation.

TABLE 1. - PERFORMANCE IN LEVEL 1 (DECLARATIVE) TRAINING.

	Student 1		Student 2		Student 3	
Operation/ Variation	# of Trials	# of Errors	# of Trials	# of Errors	# of Trials	# of Errors
Flt Sel	8	2	7	3	11	3
Disp Req	4	0	4	0	6	2
TV Chan	4	0	4	0	4	0
AES Sel	6	2	5	1	4	0
AES Des	3	0	7	1	2	0
DDD Sel	4	0	2	0	3	1
DDD Rel	2	0	4	1	6	1
DDD Reset Op	2	0	4	0	3	0
DDD Reset Crit	4	1	2	0	2	0
DDD Sel Drive	2	0	2	0	6	1
DDD Sel Data	7	2	2	0	3	1
DDD Sel Lamp	2	0	2	0	5	1

For Level 3 (Automaticity) training, the three students performed an average of 35.3 trials (see Table 3). They required an average of 75 minutes to complete the training. During this time, students made an average of 8.3 errors on actions. In Level 3 training, remediation trials were administered not after each error, but rather if a performance criteria was not met. The criteria involved beep response accuracy, performance speed (i.e., > 40 seconds), and action errors. Only one student received remediation during Level 3 training, performing 18 trials and making 3 errors across 2 remediation cycles.

It is interesting to note that unlike previous training levels, students required substantially different amounts of time to complete Level 3 training. The dual task paradigm was quite novel for the two flight controllers, at least partially explaining the differing time requirements. These two students also reported finding performing two tasks at one time difficult. The more experienced task controller, however, appeared to have less difficulty. This may be due to greater familiarity and experience with MSK tasks and console use.

One other issue relating to automaticity training is that the ITS did not terminate training upon satisfying the performance criteria for two of the students. The performance criteria was two successful trials of each of the five operations (see above). Additional information is needed to determine why training was not terminated when expected.

TABLE 2. - PERFORMANCE IN LEVEL 2 (SPEED) TRAINING.

	Student 1	Student 2	Student 3
# of Trials	28	37	25
# of Errors	2	3	6
# Successful Trials	11	11	10
# Remediation Trials	16	18	31
# Remediation Errors	3	2	6
Time Required (Min.)	35	50	60

TABLE 3. - PERFORMANCE IN LEVEL 3 (AUTOMATICITY) TRAINING.

	Student 1	Student 2	Student 3
# of Trials	29	53	24
# of Errors	9	10	13
# Successful Trials	10	21	13
# Remediation Cycles	0	2	0
# Remediation Errors	NA	3	NA
Time Required (Min.)	50	150	25

Two additional analyses were conducted. First, performance speed was plotted against amount of task practice. One would expect students' performance to fit a learning curve during Level 2 and possibly Level 3 (as they learn to perform the secondary task). However, logarithmic functions did not explain as much of the performance variance as expected, ranged from 4% to 58% variance accounted for (see Figure 1).

Thus, a second analysis was conducted to determine whether discrepancies from the expected learning curve could be explained in terms of amount of practice and response speed on specific task actions. The 12 actions each had 6 task steps—with the exceptions of TV Chan (5 steps) and Disp Req (7 steps). However, it may have taken students longer to perform actions they were less familiar with (i.e., had received fewer trials of practice on). For Level 2 training, though,

TABLE 4. - PERFORMANCE PRACTICE AND SPEED IN LEVEL 2 TRAINING

	Student 1		Student 2		Student 3	
Operation/ Variation	# of Trials	Average Response Time	# of Trials	Average Response Time	# of Trials	Average Response Time
Flt Sel	3	22.8	7	25.8	4	18.7
Disp Req	9	24.0	3	21.9	4	22.3
TV Chan	3	17.9	3	26.6	2	14.9
AES Sel	3	22.8	6	23.6	3	23.2
AES Des	0	NA	2	25.5	2	20.6
DDD Sel	5	24.1	3	19.0	0	NA
DDD Rel	1	21.9	5	20.7	2	19.0
DDD Reset Op	0	NA	1	42.5	0	NA
DDD Reset Crit	0	NA	0	NA	0	NA
DDD Sel Drive	0	NA	1	25.4	0	NA
DDD Sel Data	2	22.8	0	NA	1	21.5
DDD Sel Lamp	0	NA	2	35.6	1	23.7

the results do not indicate clear differences between response times for either amount of practice (i.e., number of trials) or action type (see Table 4). Similarly, for Level 3 training, the results do not indicate clear differences between response times for either amount of practice or action types (see Table 5). No specific pattern of response time differences were observed across students in either training Levels 2 or 3 with the exception that speed on a specific action type increased with additional task practice. For example, Student 1 increased response time on Disp Req from 39.9 to 16.1 seconds across 7 trials of practice.

TABLE 5. - PERFORMANCE PRACTICE AND SPEED IN LEVEL 3 TRAINING

	Student 1		Student 2		Student 3	
Operation/ Variation	# of Trials	Average Response Time	# of Trials	Average Response Time	# of Trials	Average Response Time
Flt Sel	3	32.8	7	41.3	3	25.3
Disp Req	3	24.5	10	33.6	4	25.0
TV Chan	6	20.4	9	18.4	2	29.0
AES Sel	2	33.0	4	24.8	3	32.8
AES Des	1	30.2	2	28.2	1	19.0
DDD Sel	1	34.8	4	26.9	0	NA
DDD Rel	1	40.5	3	22.0	1	29.5
DDD Reset Op	1	26.3	0	NA	0	NA
DDD Reset Crit	0	NA	2	43.4	1	16.6
DDD Sel Drive	0	NA	1	50.1	1	51.2
DDD Sel Data	0	NA	0	NA	1	31.0
DDD Sel Lamp	2	27.6	1	26.3	1	41.7

The lack of systematic differences in response times in different action types was unexpected given that smaller amounts of practice were received on some actions—most notably the 7 DDD actions. Indeed, as

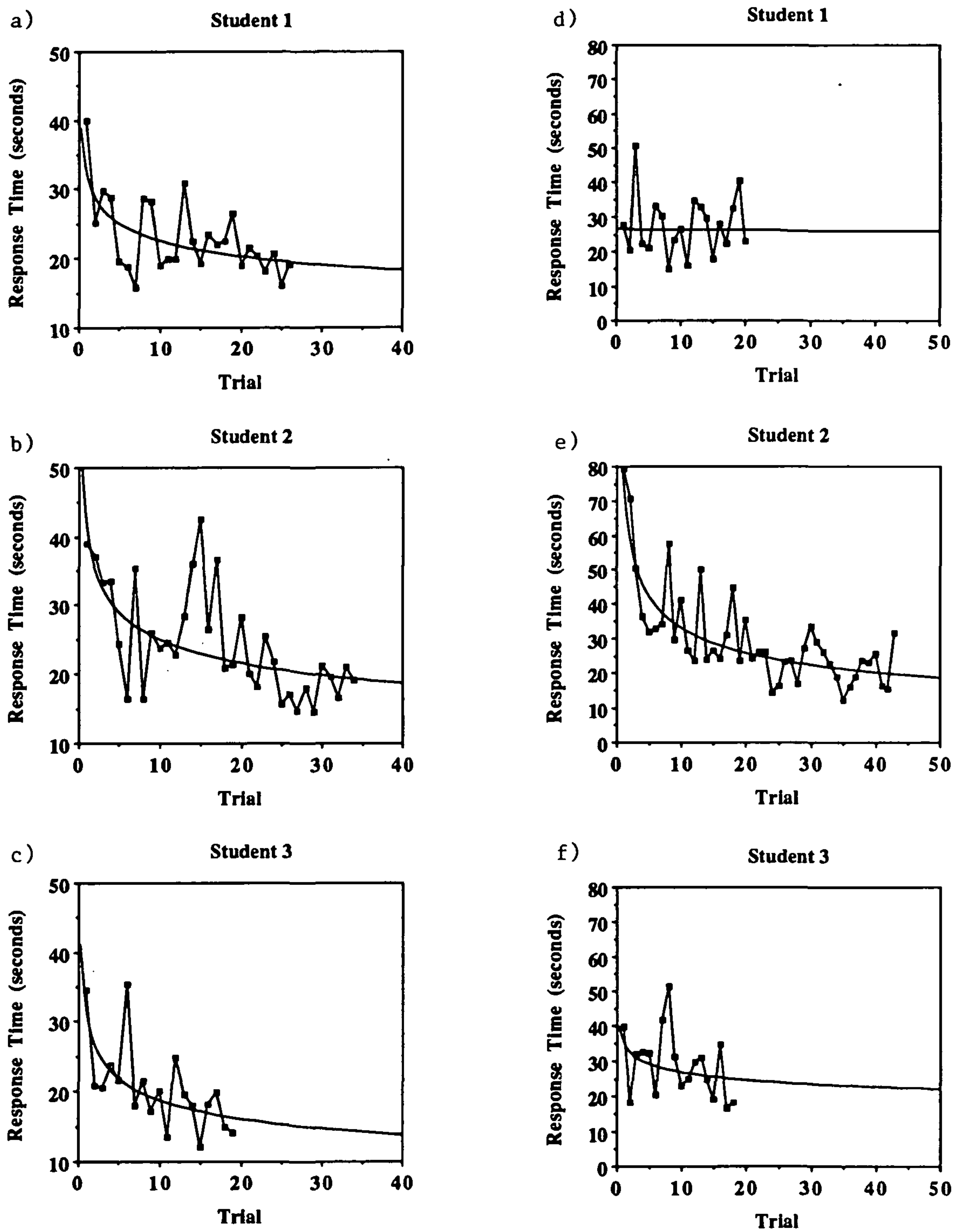

Figure 1.- Response time for task actions across trials. Speed training in panels a, b, & c. Automaticity training in d, e, & f.

shown in Tables 4 and 5, no practice at all was received on some actions in Levels 2 and 3 training. It was expected that the smaller amount of practice would result in substantially increased response times. This result was not observed. However, students did report more difficulty performing the 7 DDD actions and suggested the provision of additional training on these actions.

Finally, additional informal observations and comments were obtained from the students on ITS content, functioning, and the user interface. Several comments addressed training content. Specifically, the students noted that action steps did not have to be performed in the trained sequence on the job. Flight controllers using the MSK on the Propulsion or other consoles may perform the steps of an action in a variety of acceptable sequences. This was known by the software developer. However, it was necessary to require action steps to be performed in a specific sequence to facilitate the automaticity training. The required step sequences did not affect the novice flight controller although the experienced flight controller reported difficulty performing the steps in the required sequence. She had learned to use the MSK using alternate but acceptable sequences on the job and reported that her previous experience interfered with task performance on the ITS. Due to the small sample size (n=1) it is not possible to draw conclusions, though, regarding the possible interference between previous task experience and current ITS task performance. In addition, the experienced flight controller noted that it is unnecessary to perform some of the steps required for different actions after the console has been initialized for a flight (Flt Sel). Another comment addressed the amount of task practice provided on the DDD actions, suggesting that additional practice be provided for each action. This system currently treats the 7 DDD actions (and the 2 AES actions) as part of one operation, which may explain why remediation following an error in speed training did not necessarily match the erroneous action. Finally, the novice flight controller reported that the training was useful, providing information and experience she had not yet obtained on the job. Similarly, the experienced flight controller reported the ITS had potential training benefits for Propulsion Console and other flight controllers although she recommended modifying the task content to more closely resemble the job and address additional components of the job.

The students also commented on ITS functioning. One issue raised was that it was unclear what the criterion was for being promoted from one phase of declarative training to the next. For example, the flight controllers expressed some frustration about having to complete multiple guided and unguided trials on a given action before moving to the next action. This resulted in part from feedback messages stating that the student was demonstrating effective performance and then stating that additional practice would be provided on that action. It may be appropriate to indicate to students how much additional practice they can expect (e.g., they will be asked to complete one additional trial or to successfully complete two consecutive trials). Additional explanation may also be appropriate during speed and automaticity training. For example, students did not initially realize during speed training that the trial started as soon as the "Goal" (i.e., the action assigned) was displayed on the screen. One student thought the trial began (the timer started) when she clicked the mouse the first time during the action. Moreover, students did not realize what the performance criteria were for successful completion of speed or automaticity training. It may be appropriate to give students more information about what performance levels are necessary to complete speed and automaticity training. Other student comments indicated that students did not understand the purpose of the secondary task during automaticity. Additional explanation could be provided regarding the purpose of secondary task performance.

Finally, student comments addressed the user interface. One issue raised was the use of scrolling rather than refreshing the tutoring/information window. Declarative information and task assignments were made in a window at the lower right portion of the screen. Students reported difficulty reading the instructions provided, often rereading a portion of the window because it was unclear where new information or instructions appeared in the window. Refreshing the window when additional information or instructions appear would resolve this issue. A second issue related to the use of color. That is, students reported difficulty seeing the red cursor (an arrow) against the purple background. A third issue involved use of the mouse. Students were initially unclear regarding the different functions of the left, middle, and right mouse keys; the keyword descriptions provided in the MSK display window were apparently not sufficient. A brief statement explaining this could be provided at the start of training. A related issue was that two mouse keys (left and right) were required to key in numbers. Students suggested allowing the numbers to cycle from 9 to 0 (and 0 to 9) so that one mouse key could be used to change numbers, although students appear to want one key (e.g., left key) to cycle downward and a second key (e.g., right key) to cycle upward.

DISCUSSION AND CONCLUSIONS

The results indicate that students learned to perform 12 MSK actions using the ITS. Further, they were able to successfully complete training within approximately four hours. However, it is not clear that students received sufficient task practice to automatize the skill. Using the current 20-second and 40-second speed requirements during speed and automaticity training, respectively, subjects performed any given action a maximum of 37 times and as few as 2 times. To ensure automaticity it may be necessary to implement more stringent speed constraints which would result in additional task trials. Further, additional information is needed to determine how the ITS functions in terms of promoting students from one training level to the next and terminating training. Also, some revisions to training content may be appropriate to make ITS tasks more closely resemble job actions.

In terms of the ease of use, students required little or no assistance in using the ITS. The ITS provided instructions and task assignments which students could follow without outside assistance. However, some clarification or additional explanation may be appropriate to ensure that students understand the performance expectations and progress of training. Some modifications may also be appropriate to improve the interface, especially in terms of window refreshing and use of color.

Finally, the results and student comments provide an indication of potential training benefits of the ITS and modifications which may further improve this training system. The results indicate that students learn the training content. In addition, both flight controllers reported that the training content was useful, especially for novice flight controllers. Moreover, the ITS has potential benefits for flight controllers on other consoles given the similarity of MSK use across consoles. These potential benefits could be further increased by expanding the training content to other console activities.

REFERENCES

[1] Wenger, E. (1987). Artificial Intelligence and Tutoring Systems. Los Altos, CA: Morgan Kaufmann.

[2] Littman, D. & Soloway, E. (1988). Evaluating ITS's: The cognitive science perspective. In M. C. Polson & J. J. Richardson, Foundations of Intelligent Tutoring Systems. Hillsdale, NJ: Erlbaum.

[3] Goldstein, I. L. (1986). Training in Organizations: Needs Assessment, Development, and Evaluation. Monterey, CA: Brooks/Cole.

[4] Fink, P. K. & Overby, M. A. (1987). Research in Intelligent Tutoring Systems for Knowledge Poor Domains. (SwRI Proposal 14-4874a). San Antonio, TX: Southwest Research Institute.

[5] Burton, R. R. & Brown, J. S. (1982). An investigation of computer coaching for informal learning activities. In D. Sleeman & J. S. Brown (Eds.), Intelligent Tutoring Systems. NY: Academic Press, 79-88.

[6] Reiser, B. J., Anderson, J. R., & Farrell, R. G. (1985). Dynamic student modelling in an intelligent tutor for LISP programming. Proceedings of the 9th International Joint Conference on Artificial Intelligence (Vol. 1, pp. 8-14).