Abstract - This study aims to characterize the effects of adaptivity within an intelligent tutoring system (ITS). An ITS authoring tool was used to create adaptive and non-adaptive (sequential) versions of lessons on programmable logic controller timer instructions. In the Adaptive versions, the content and number of practice questions asked varied based on student performance. The Sequential versions were identical to the Adaptive versions except that they asked practice questions from a predetermined list. Students in both groups showed statistically significant improvement in post-test performance, and there was no difference between groups in the amount of improvement. Students in the Sequential group answered practice questions more accurately than students in the Adaptive group. However, students in the Adaptive group required significantly less time to complete the lessons. Correlation data suggest that the Adaptive lessons were more efficient for knowledgeable students.

This research suggests that adaptive approaches can make learning more efficient, but are not necessarily more effective than pedagogically sound non-adaptive approaches. It also demonstrates effective data collection and analysis techniques for studying ITS features such as adaptivity. Use of these techniques can facilitate the design of instructionally effective ITSs and inform design of other tools for ITS research and evaluation.

Index Terms – intelligent tutoring system, adaptivity, programmable logic controller, evaluation.

INTRODUCTION

Intelligent tutoring systems (ITSs) are generally known for their ability to adapt instruction to the needs of individual students. However, the meaning of the word “adapt” and approaches to creating adaptive instruction can vary widely. Arguably, even non-intelligent computer-based tutorials can provide an instructionally effective level of adaptivity by the use of branching to provide different instructional presentations based on student performance. Since ITSs can be extremely expensive and time-consuming to develop, it can be useful to assess the effects of the adaptivity they provide.

The goal of the study reported here was to characterize and evaluate the effects of adaptivity within an intelligent tutoring system designed to teach about timer instructions, a type of command used in programmable logic controller (PLC) programs.

BACKGROUND

Typically, an ITS consists of several modules, including a domain knowledge base, a student model, a teaching strategies module, and an interface [1]. The student model provides a basis for adapting instruction to the needs of the student. It can be used to evaluate the student’s intent, predict student behavior, and help the teaching module to select appropriate individualized instructions and questions [2].

Since learning is a cognitive skill acquisition process, it is helpful to view the student model in terms of cognitive theories such as Anderson’s ACT-R theory [3]. Anderson proposes that there are two types of knowledge: declarative and procedural.

Declarative knowledge basically consists of facts, usually acquired by reading or some other experiences. It has no state. For example, the fact “TON stands for timer on delay” stands alone. A student can read it and know what TON stands for.

In contrast, procedural knowledge is more contextual. Many cognitive theories, including ACT-R theory, represent procedural knowledge as a set of production rules that associate problem states and problem-solving goals with actions and consequent state changes. Appropriately applying these rules leads the correct solution of the target problem. Declarative knowledge facilitates use of procedural knowledge for problem solving. Hence, when a student makes a mistake or fail to solve a given problem, it may because he or she doesn’t know the required declarative knowledge, or has not yet formed the necessary procedural knowledge from his or her declarative knowledge.

Based on this theory, to model a student, we need to be able to both trace his/her problem solving path (model...
tracing) and keep track of the declarative knowledge the student has acquired thus far (knowledge tracing). That is, a student model should both:
1. Record student behavior (i.e., declarative knowledge modeling); and
2. Analyze the student’s problem solving path and predict student actions (i.e., problem-solving modeling or “cognitive diagnosis”).

**Declarative Knowledge Modeling**

In declarative knowledge modeling, the system simply records student’s behavior—e.g., by keeping track of which questions were answered correctly or incorrectly. The system provides answers and feedback based on student actions, but does not attempt to justify the student’s behavior. The model is basically a repository for student’s beliefs about the learning material [4]. Complex inferencing is not required.

This approach to student modeling is appropriate for relatively predictable environments. The knowledge base contains a certain amount of domain knowledge and the student’s knowledge is a subset. The student’s actions may also be used to detect misconceptions relative to the knowledge in the knowledge base. This type of model can be used to teach basic concepts, definitions and terminology. True/false or multiple-choice questions are usually used to evaluate the student’s learning progress and feedback is provided for each question. Although effort may be made to analyze reasons for misconception or missing knowledge, the system primarily focuses on correctness rather than elaboration and diagnosis. This approach is relatively straightforward to design and implement.

An example of a system that uses declarative modeling is Ozdemir and Alpaslan’s intelligent tutoring system for student guidance in web-based courses [5]. This is a web-based teaching agent that helps students learn concepts by providing individualized navigational support. However, this system lacks a mechanism for handling misconceptions, which is very important issue in student modeling.

Rosic, Glavinic, and Stankov describe a distributed tutor-expert system [6]. Like most systems that use a declarative approach, this system stores a user’s behavior and test results in a database. The system provides adaptive instruction by evaluating every step in student testing. After each step, the system generates new questions depending on a student’s results. However, this system does not employ sophisticated methods of handling students’ misconceptions.

**Problem-Solving Modeling**

Problem-solving modeling is more complicated and intelligent than declarative knowledge modeling. The system not only records a student’s behavior, but also traces the student’s problem solving path. The student model is based on some theory or artificial intelligence method and can be used to predict student responses.

According to Wenger, this type of student model should reflect those aspects of student behavior and knowledge that affect the student’s learning ability [1]. Also the model must determine how to assign blame in the presence of a student’s failure to solve a problem in which more than one skill is necessary for solution development, and limit the number of hypotheses about the student’s erroneous behavior so a combinatorial explosion of possibilities does not occur. Finally, the model should be able to work in the presence of noisy data, such as when the student makes errors because of fatigue or overload. This type of model can be used to predict what the student will do next based on the student’s problem-solving history.

ITSs using this student modeling technique are usually problem solving agents. Student are asked to solve some problem by using the knowledge they were just taught. The reason why a student makes a mistake may be more complicated than when a student only needs to memorize a concept. Hence, it takes more effort to develop this type of student model.

INTUITION is an example of an ITS that uses problem-solving modeling in the context of a business management simulation game [7]. Its student model keeps a record of a student’s misconceptions or missing concepts that have been diagnosed. It includes information on the context in which each misconception or missing concept has been diagnosed and how many times it occurred. This is used to adapt remedial intervention to the needs of the student.

Nwana and Coxhead developed an ITS to help student learn to add fractions [8]. This system is able to diagnose skills that a student is missing and present an appropriate lesson.

**Assessing Effects of Adaptivity**

Although several evaluations of ITS effectiveness have been conducted at a general level, there has been little emphasis on specifically evaluating the effects of the adaptive elements within these systems. This may be because it can be time-consuming and/or difficult to develop an equivalent but non-adaptive control treatment for evaluation purposes, especially for ITSs that utilize problem-solving modeling.

The goal of this study was to characterize and evaluate the effects of adaptivity within an intelligent tutoring system designed to teach about timer instructions, a type of command used in programmable logic controller (PLC) programs.

The ITS was developed using an intelligent tutoring system authoring tool called XAIDA [9]. XAIDA was developed for the Air Force Research Laboratory in the early 1990s to enable maintenance technicians (i.e., non-instructional design experts) to rapidly develop instructionally effective courseware. An expert uses XAIDA’s development program to create a knowledge base of facts. XAIDA then uses an instructional delivery program that draws upon the developed knowledge base in
presenting instruction and generating adaptive practice sessions.

XAIDA’s student model uses declarative knowledge modeling. It marks facts that it thinks the student knows, based on student performance on the practice questions. It also detects and models misconceptions. Facts that XAIDA thinks a student doesn’t know and misconceptions tend to be asked about more often than facts a student seems to know. Thus no two students will be asked exactly the same set of practice questions, and more knowledgeable students will be asked fewer questions than less knowledgeable ones. XAIDA also generates detailed response-specific feedback whenever a student answers a question incorrectly.

XAIDA was selected because its teaching methods are appropriate for teaching certain aspects of programmable logic controller knowledge—namely, characteristics of PLC commands. The authors have used XAIDA successfully to teach students in previous studies [10, 11].

In addition, because XAIDA was developed for research, it includes several features that make it an excellent tool for studying ITS development and use. One of these is a user action recording capability. Another is the ability to allow researchers to replace XAIDA’s adaptive practice sessions with a canned list of questions. We used these two features to develop equivalent adaptive and non-adaptive lessons to explore the following questions:

- Did use of the adaptive approach result in better learning?
- Did use of the adaptive approach result in more efficient learning?
- Did students exhibit different question-answering behaviors under the two approaches?
- Did students prefer one approach over the other?

**METHODOLOGY**

This section describes the participants, materials, and procedures for this study.

**Participants**

Participants were 34 undergraduate students enrolled in an upper-level course on manufacturing automation and robotics in an engineering program at a major U.S. university.

**Materials**

For the adaptive treatment, we used three XAIDA lessons previously developed to teach about PLC timer instructions [11]. Each of these lessons consisted of a short presentation about three timer instructions followed by an adaptive practice question session. The length and emphasis of the practice question sessions are determined by a Question Generator module. The Question Generator individualizes practice by using a model of what the system believes the student knows at any given time to select what to ask about next (i.e., a student model). Thus it is unlikely that two students will experience practice in exactly the same way.

For the non-adaptive treatment (which we will henceforth refer to as the sequential treatment), we developed three XAIDA lessons that were exactly the same as the three adaptive lessons EXCEPT that the practice questions in the three new lessons were arranged in a fixed sequence. Figure 1 shows a sample question sequence from a sequential lesson.

The following instruments were used for data collection and to evaluate student’s mastery of lesson content:

- **Pre and post-tests.** Two parallel 30-item multiple-choice tests were developed for use in pre- and post-testing students’ knowledge of timer instructions.
- **User action logs.** We used XAIDA’s user action recording capability to collect data such as which questions users answered, number of questions attempted, number of questions answered correctly, and time spent in each lesson.
- **Opinion survey.** A six-question opinion survey asked students to rate various characteristics of the lessons on a 7-point Likert scale. Figure 2 contains sample questions from the tests and opinion survey. Students were also given the opportunity to write comments about the lesson material.

**Procedure**

The study took place during the students’ 1½-hour lab period. There were three lab periods with 11-12 students each. During each lab, students took the pretest on timer lesson content. They were then randomly assigned to take either the Adaptive or the Sequential lessons. After completing three Adaptive or Sequential lessons, they took the post-test and completed the opinion survey.

**DATA ANALYSIS AND RESULTS**

This section describes the data analysis procedures and results in terms of the objectives of this study.

**Did the adaptive approach result in better learning?**

To answer this question, we performed two-tailed paired t-tests using the pre- and post-test data to see if there was statistically significant score improvement between tests. Because of time constraints and late arrivals, some students were not able to take the pre-test or the post-test. In cases where subjects were not present for both tests, we only used complete pairs. The null hypothesis $H_0$ was that there would be no change.

The analysis results revealed that the null hypothesis was rejected for both the Adaptive and Sequential groups ($p < .05$, see Table I). This suggests both treatments resulted in significant improvement in learning.
We also performed an ANOVA test of means with unknown sample variance to see if there was a difference in the post-test means between the Adaptive and Sequential groups. There was no significant difference in post-test performance between the two groups (p = 0.5677).

**Did the adaptive approach result in more efficient learning?**

To answer this question, we calculated the average time spent in practice question sessions for each of the three lessons by the Adaptive and Sequential groups. The average time spent per lesson by the Adaptive group was 10.5 minutes. The average time spent per lesson by the Sequential group was 13.2 minutes. We then performed an ANOVA test of means with unknown sample variance to see if there was a significant difference between the two groups in terms of time spent per lesson. The Adaptive group spent significantly less time per lesson than the Sequential group (p = 0.0325).

A possible reason for this difference is that the Adaptive approach makes it possible for knowledgeable students to finish more quickly, because it allows them to exit the lesson as soon as they demonstrate mastery of the material (they are not restricted to a “lock-step” presentation). To evaluate this possibility, we computed correlations between time students spent in each lesson and their pre- and post-test scores.

For students in the Adaptive group, the correlations were significant and in a negative direction: for the pretest, r = -0.6855 (p < .001) and for the post-test, r = -0.6697 (p < .0001). This suggests that students with higher pre-test scores in the Adaptive group tended to take less time to complete lessons.

Furthermore, for students in the Sequential group, the correlations were not significant: for the pretest, r = -0.1036 (p = .59) and for the posttest, r = -0.0919 (p = .63). This suggests that for the Sequential condition, there is no relationship between pre or post-test scores and time needed to complete a lesson.

**Did the two groups exhibit different question-answering behaviors?**

We were curious to see if the different questioning approaches might somehow affect students’ behavior in answering questions. For example, perhaps students in one group might be more likely to answer questions without taking time to think carefully about the correct answer. To investigate this issue, we decided to look for differences between the two groups in terms of time spent per question and percent of questions answered correctly per lesson.

To compare time spent per question, we computed the number of questions answered per minute by each student for each lesson. We found that students in the Adaptive group answered about 2.02 questions per minute versus 1.78 questions per minute for students in the Sequential group. However the difference was not significant (p = 0.21).

To compare accuracy, we divided the number of questions answered correctly by the number of questions attempted for each lesson for each student. We found that students in the Adaptive group answered practice questions correctly about 76.3% of the time, whereas students in the Sequential group answered practice questions correctly about 85.4% of the time. This difference was significant (p=0.0420).

**Did students prefer one approach over the other?**

We were curious to know if the two approaches might affect students’ perceptions of the lessons differently. To investigate this issue, we looked for differences between the two groups in terms of their ratings of the timer lessons on the opinion survey. The opinion survey consisted of six questions asking students to rate the lessons on a 7-point Likert scale. We compared the two groups’ mean Likert ratings for each question. We found that for all six questions, the Adaptive group rated the lessons more favorably than the Sequential group. However, none of the differences were significant at the p < 0.05 level; the probabilities were all in the 0.14 to 0.20 range.

Because of the limited time available to conduct this study, we received only a few student comments. A few students in both conditions complained that the practice questions were boring and repetitive. One student in the Adaptive condition wished that the questions would start with simple concepts and gradually get harder (which is what the Sequential lesson does).

**DISCUSSION**

The data reveal that students in general—and the more knowledgeable students in particular—tended to finish lessons more quickly in the Adaptive treatment than in the Sequential treatment. This suggests that the Adaptive approach is more efficient for good students. This is probably because it allows them to exit the lessons as soon as they demonstrate knowledge of the facts to be mastered, rather than forcing them to plow through a predetermined list of questions.

The data also show that students learned as a result of their exposure to the timer ITS regardless of which group they were in. Both groups showed a significant rise in test score from the pre-test to the post-test. Moreover, there was no significant difference in post-test performance between the Adaptive and Sequential groups. The best explanation for this seems to be that the structure of the Sequential lesson was pedagogically sound. The questions in the Sequential treatment were grouped by topic and arranged in the order of presentation. The practice questions provided detailed, answer-specific feedback for both correct and incorrect answers. Finally, the Sequential lessons included a lot of questions, so there was sufficient opportunity for students to practice and learn from the feedback.
An interesting finding was that students in the Sequential group tended to answer questions more accurately during practice. Does this suggest that they were more focused than students in the Adaptive group? If so, why did the two groups perform similarly well on the post-test? The best explanation we can think of at this time is that the questions in the Sequential lessons were easier to answer because they were presented in the order in which the material was presented. In contrast, the sequence of questions in the Adaptive treatment is more unpredictable, especially at the beginning of a practice session, when XAIDA supposes that the student doesn’t know any of the facts and randomly selects what to ask about. This may be akin to the difference between studying a vocabulary list in a fixed sequence versus using flash cards. The random ordering of flash cards makes it more challenging to recall correct answers. In addition, as XAIDA gets a better fix on what the student knows and doesn’t know, it begins to focus specifically on asking about facts the student doesn’t know—which also makes it more likely that students will miss questions. The experience of answering questions incorrectly is not usually considered to be pleasant, but it can be an effective and efficient way to learn.

The opinion survey data suggested that students preferred the Adaptive to the Sequential approach, but were not conclusive.

CONCLUSION AND FUTURE DIRECTIONS

This study suggests that adaptivity can make learning more efficient, but that an adaptive approach is not necessarily more effective than a pedagogically sound non-adaptive approach. Further studies are needed before any final conclusions can be made regarding which method is more effective.

In the future, we would like to repeat this study with more students and to either reduce the number of activities or increase the time available to conduct the study. This will make it possible for all students to complete all the study activities more comfortably. Collecting more data, especially student comments, will help us to better assess if students prefer one approach to another. We will also analyze the content and sequence of the questions being asked, in addition to calculating statistics such as time spent per lesson and number of questions answered per minute. This may help us to gain more insight about issues such as why students in the adaptive condition tended to miss practice questions more often than students in the sequential group did, and to what extent did making errors affect students’ learning and opinions about their instructional experience. Studies with students from different areas would also be useful.

Finally, we would like to create web-based ITSs that incorporate research-friendly features such as XAIDA’s data recording capability. Although it is possible that such capabilities will tell us things we don’t want to know about the cost-effectiveness of ITS development, we believe that they can also provide greater insight on little-researched areas such as the effects of adaptivity on students’ attitude and behavior.

ACKNOWLEDGMENT

This material was supported by a National Science Foundation Course, Curriculum, and Laboratory Improvement (CCLI) grant (No. 0088873) and a gift from Rockwell Automation. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation or Rockwell Automation. XAIDA is a prototype intelligent tutoring system authoring tool originally developed by MATCOM (http://www.matcomcorp.com) for the Air Force Research Laboratory (AFRL/HEA) under Contract No. F41624-93-F5002. We thank Dr. Henry Halff, XAIDA’s project director, for his foresight in designing XAIDA to accommodate research projects such as this one.

REFERENCES


1. TON stands for: __________
2. TOF stands for: __________
3. RTO stands for: __________
4. The time delay period for TON is: 1) from time the rung goes TRUE until DN bit is turned on; 2) from time the rung goes FALSE until DN bit turns off; 3) from time the rung goes TRUE until DN bit is turned on.
5. The time delay period for TOF is: 1) from time the rung goes TRUE until DN bit is turned on; 2) from time the rung goes FALSE until DN bit turns off; 3) from time the rung goes TRUE until DN bit is turned on.
6. The time delay period for RTO is: 1) from time the rung goes TRUE until DN bit is turned on; 2) from time the rung goes FALSE until DN bit turns off; 3) from time the rung goes TRUE until DN bit is turned on.
7. Select all that apply. The instruction(s) for which the time delay period is from time the rung goes TRUE until DN bit is turned on is: 1) TON; 2) TOF; 3) RTO.
8. Select all that apply: The instruction(s) for which the time delay period is from time the rung goes FALSE until DN bit turns off is: 1) TON; 2) TOF; 3) RTO.

FIGURE 1
SAMPLE LIST OF QUESTIONS FROM SEQUENTIAL LESSON

SAMPLE TEST QUESTION
Select all answers that apply. A TOF instruction can be used:
(1) for maintenance functions
(2) in diagnostic programs
(3) to create a short pulse at the beginning of a longer input condition
(4) to create longer output functions derived from short input functions
(5) to delay start of function for a defined period of time from start of some other function
(6) to generate a short pulse at the end of a long input function

SAMPLE OPINION SURVEY QUESTION
The practice questions helped me to learn the material.
Strongly disagree 1 2 3 4 5 6 7 Strongly agree

FIGURE 2
SAMPLE TEST AND OPINION SURVEY QUESTIONS

TABLE I
COMPARISON OF PRE- AND POST-TEST MEANS (2-TAILED T-TEST)

<table>
<thead>
<tr>
<th></th>
<th>Pretest mean</th>
<th>Post-test mean</th>
<th>Difference</th>
<th>Probability</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adaptive (n=11)</td>
<td>61.5</td>
<td>77.0</td>
<td>15.5</td>
<td>0.0153</td>
<td>Reject H₀</td>
</tr>
<tr>
<td>Sequential (n=10)</td>
<td>63.5</td>
<td>79.9</td>
<td>16.4</td>
<td>0.0016</td>
<td>Reject H₀</td>
</tr>
</tbody>
</table>