

Intelligent Tutoring System Authoring Tool for Manufacturing Engineering Education*

SHENG-JEN ('TONY') HSIEH

Department of Engineering Technology and Industrial Distribution, Texas A&M University,
College Station, TX 77843, USA. E-mail: hsieh@entc.tamu.edu

PATRICIA YEE HSIEH

System Administrative and General Offices, The Texas A&M University System, College Station,
TX 77840, USA

Intelligent tutoring systems (ITSs) have potential for making computer-based instruction more adaptive and interactive, but development in the area of manufacturing engineering has been rare. However, recent developments in the area of ITS authoring tools may make this technology more accessible. The objectives of this study were to: 1) evaluate the feasibility of faculty course development using an ITS authoring tool; and 2) evaluate the instructional effectiveness of the developed courseware. An ITS authoring tool called XAIDA was used to develop a tutorial on how to use a computer numerical control (CNC) machine. This paper summarizes the results of a preliminary evaluation conducted with 25 undergraduate manufacturing engineering students. The results suggest that instructional development in XAIDA is feasible and quick, and that students learned from and enjoyed the tutorial.

INTRODUCTION

MODERN MANUFACTURING requires the use of expensive and sophisticated equipment. Many manufacturing engineering programs prepare students for industry by exposing them to the types of tools and equipment they will need to use. An example might include the use of programmable logic controllers, sensors and instrumentation, robots, and machine vision systems for process automation system design as part of a course on Manufacturing Automation and Robotics. A course on Production Systems and Computer-Aided Manufacturing uses CAD/CAM software (e.g. FeatureCAM) and computer numerical control (CNC) machines. Courses on Quality Assurance and Electronics Manufacturing require use of expensive machines such as surface mounted placement machines (SMT) to practice the electronic manufacturing process. However, obstacles such as high student/faculty ratios, limited access to the lab, (limiting practice using equipment to scheduled lab times), limited resources to support students outside labs and the classroom and limited equipment to support lab assignments often hinder the learning experience. It is not unusual for a lab class of 10–20 students to share 3–5 sets of equipment.

There has been an increasing trend in manufacturing engineering education to improve education and alleviate equipment shortage problems by harnessing computer technology. Bengu [1]

describes an ongoing effort to improve entry-level engineering education by using multimedia tools, courseware, and the World-Wide Web for presenting instructional and program information, providing simulations, and administering assignments and exams. Shiue and others [2] developed a laboratory environment to support courses on Manufacturing Processes, Intermediate Manufacturing, and Concurrent Engineering. This environment integrates a variety of computer-based tools, including Computer Aided Design and Manufacturing workstations, CNC machines, robotic assembly stations, and 3-D modeling packages. Byrkett and Ettouney [3] present a model (based on their nine years of experience incorporating Computer Integrated Manufacturing into their engineering curriculum) for incorporating technology into an engineering curriculum in a way that involves students and faculty in the integration and development activities.

Several educational institutions have recently developed web-based systems to support various aspects of distance education. For example, Choren and others [4] developed a Web-based learning environment called AulaNet. AulaNet utilizes a groupware approach to facilitate

- easy reuse of existing educational materials,
- the development of learning communities of teachers and students using online communications tools (e.g., email, chat, discussion lists),
- teamwork via coordination mechanisms (such as calendar management tools),
- development of online testing and grading tools.

* Accepted 2 April 2001.

Doube [5] describes the development of a browser based system called WebFace, that supports distance education by coordinating the distribution of paper-based materials. Magoulas [6], Papanikolaou and others [7] describe development of techniques to allow adaptive lesson presentation in Web-based courses.

In this investigation, the focus is on the use of a computer to deliver instruction. In theory, computers should be able to provide individualized instruction, much as a human tutor would. In addition, unlike human instructors, they can be available around the clock, wherever there is a computer. Although they cannot replace the experience of using actual equipment, they can teach much of the preliminary information students need to know about equipment before using it, so that students spend lab time more productively and efficiently.

We have a particular interest in a class of computer based instructional systems known as intelligent tutoring systems that holds great promise for manufacturing engineering education in the future. An intelligent tutoring system (ITS) is a computer-based instructional system that utilizes artificial intelligence techniques to represent the knowledge needed to teach about a subject. A classic ITS consists of four components:

- an expert module that represents the knowledge to be taught,
- a pedagogical module that represents the teaching strategies employed by the system,
- a student model that represents what the student knows about the subject at any given point in the instruction,
- a student interface module that allows the system to interact with the student [8].

The advantage of a knowledge-based approach is the ability to update and manipulate content easily, as well as allowing the revision or addition to the system of instructional strategies. This contrasts with traditional computer-based instruction, where the knowledge and instructional strategies are fixed within a predetermined instructional presentation.

Development of intelligent tutoring systems specifically for manufacturing engineering subjects has seen little progress. Early ITS development efforts during the late 1970s and early 1980s began to explore computational and communication issues related to getting a computer to teach like a human [8, 9]. Examples of ITSs developed during this period include:

- WEST (game playing)
- SOPHIE (electronics troubleshooting)
- GUIDON (medical diagnosis)
- WHY (geography)
- STEAMER (power plant theory of operation)
- QUEST (physics problem-solving)

- EXCHECK (mathematical proofs)
- BUGGY (arithmetic problem-solving).

There was never any intention that these systems would be used to teach manufacturing related subjects, or even be used outside the laboratory. The primary purpose of development efforts in the mid 1980s was to support military training topics such as electronics troubleshooting [10, 11], tactical thinking [12], and high performance tasks such as flying a fighter plane [13]. Although the beginnings of a trend toward developing systems for actual educational use was evident; none of the systems in this period supported manufacturing topics either. This might be because ITS development required substantial investment in human and computer resources during this period; and the manufacturing industry, unlike the military, does not typically support instructional technology R&D.

Since the mid 1980s, however, the number of intelligent tutoring system development efforts has multiplied to the point where it is no longer feasible to write an exhaustive survey of the field. In addition, intelligent tutoring systems have begun to leave the laboratory, and field evaluations have become more common (e.g. Koedinger and others). Although intelligent tutoring systems in the manufacturing domain received no mention in a number of recent articles [15, 16, 17], there is evidence of change in this area. Development of specific intelligent tutoring systems that teach visual inspection [18], nondestructive inspection [19], unit operations and process systems [20], design for manufacturing [21], progressive die design [22], and factory management [23] is either currently underway or already completed. However, we found no reports of use of intelligent tutoring technology to teach about computer numerical control machines or any other manufacturing equipment.

In recent years, the focus of ITS research has shifted from the development of systems dedicated to a particular subject to the development of ITS authoring tools accessible to non-programmers. A number of these authoring tools might be suitable for teaching about manufacturing equipment, including Safari [24], Rapid ITS Development Environment (RIDES) [25], and the Experimental Advanced Instructional Design Advisor (XAIDA) [26].

ITS authoring tools generally offer a development interface that allows developers to create a knowledge base; and an instructional delivery interface that presents information from the knowledge base to the student. The knowledge representation format and instructional approach of the developed ITS varies from tool to tool; with specialized tools developed for a particular class of content and approach. For example, RIDES allows instructors to develop simulations of equipment; students can then use the simulations to learn about the equipment and to practice procedures [26]. The Goal-Based Scenario Builder

facilitates development of a curriculum that uses a particular teaching strategy, namely, learning-by-doing through investigation [27].

XAIDA is a prototype of an authoring tool originally designed to allow expert maintenance technicians to develop ITSs for maintenance topics. XAIDA consists of four sub-tools (called shells), each of which uses a different scheme for representing and teaching a specific aspect of maintenance knowledge. The Physical Characteristics shell, for example, uses a knowledge base organized as a semantic net to explain about the parts of a device or system. Thus, when used to explain about equipment, the nodes of the network are parts, and the information about part characteristics (such as sub-parts, functions, and connections) is associated with each part. The developer’s task is to populate the knowledge base with information that explains about the system. XAIDA can then apply built-in algorithms and templates for instructional presentation, question generation and feedback, and student modeling to generate an interactive, adaptive lesson automatically, based on the information provided.

The purpose of this study was to investigate the feasibility of using an ITS authoring tool for manufacturing engineering courseware development. The authors used XAIDA for development because of its availability, and because other instructors have used it successfully in a variety of domains [26, 28]. Central questions included:

1. How suitable is XAIDA for development of a computer-based curriculum with manufacturing engineering-related content?
2. How quickly can a tutorial be developed using XAIDA?
3. Is courseware developed by XAIDA instructionally effective?
4. How will students react to XAIDA-generated courseware?

We also hoped that, since XAIDA is a knowledge-based authoring tool, and since there is often a restriction in the knowledge such tools can represent, this study could give us a sense for the difficulties inherent in the use of such tools for manufacturing engineering education.

METHOD

The authors selected XAIDA, an ITS authoring tool, because it is free for educational use in its current (prototype) form, and used it to develop a tutorial explaining how to set up a computer numerical control (CNC) machine to machine a part. The first author teaches a course (Numerical Control and Computer-Aided Manufacturing) for which there is only one CNC machine available for lab classes of 15–20 students, and this provided an opportunity to use setup of the CNC machine as a topic. Because students must take turns using the machine, and because they are often learning about it as they use it, they do not have enough time to become proficient. The intention of the tutorial was to familiarize students with the procedure before they actually use the machine, so that they get more out of their experience with the actual equipment. The tutorial can also serve as a means of review. The following sections describe the development and evaluation of the tutorial.

Tutorial development

The authors also hoped that, since XAIDA is a knowledge-based authoring tool, and since such tools are often restricted in the knowledge they can represent, this study would provide a sense for the difficulties inherent in the use of such tools for manufacturing engineering education. As previously stated, the purpose of this shell was to represent knowledge about the physical characteristics of a device or system. The designers chose to identify the nodes of a network as parts, and associated information about part characteristics such as sub-parts, functions, and connections with each part. In our case, since the subject of our tutorial was a procedure, we decided that the nodes in our network would be steps in the procedure, and associated step-performance related information, such as which menu to use to access controls, with each step.

The authors organized the process of setting up the CNC machine into five major steps:

- Step 1: home the axes;
- Step 2: load the program;
- Step 3: set up the tools;

Table 1. Information associated with the “Preview Program” step

Sub-Steps/Notes	Illustration
This step allows you to verify that your program behaves as you intended before you actually machine the part.	Blackboard
On the Basic menu, press F5 to preview your program.	Arrow pointing to F5 soft key on Basic menu
In the blue VIEW SIZE box, you will see the settings for the View window. Make sure the settings are a little larger than needed to display the work piece. Enter numbers as needed, hitting Enter after each line.	VIEW SIZE box
After the last line, a blank grid will appear. Press F2 (PAUSE) to preview you program step by step. You can also press and hold F2 to preview the program without a pause.	Blank grid
Watch the cutting path to make sure that it is doing what you intended. If the cutting path is not consistent with your design, check your program. When you are finished previewing your program, press the Escape (Esc) key to return to the Basic menu.	Show cutting path in preview window.

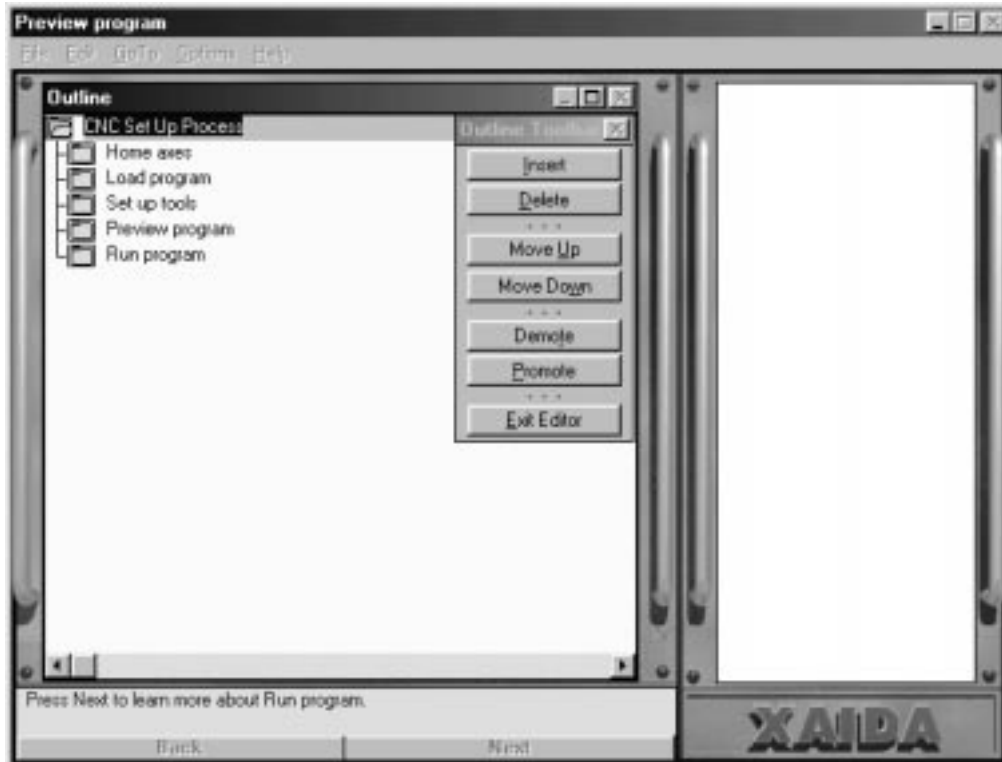


Fig. 1. Outline editor in XAIDA Develop.

- Step 4: preview the program;
- Step 5: run the program.

Table 1 contains the information associated with step 4, preview program.

The authors identified and produced appropriate

illustrative graphics and images, such as screen shots, as well as identifying important information related to the procedure and formulating it into multiple-choice and true/false practice questions. Note that this latter step would not have been required if the authors had used XAIDA-PC

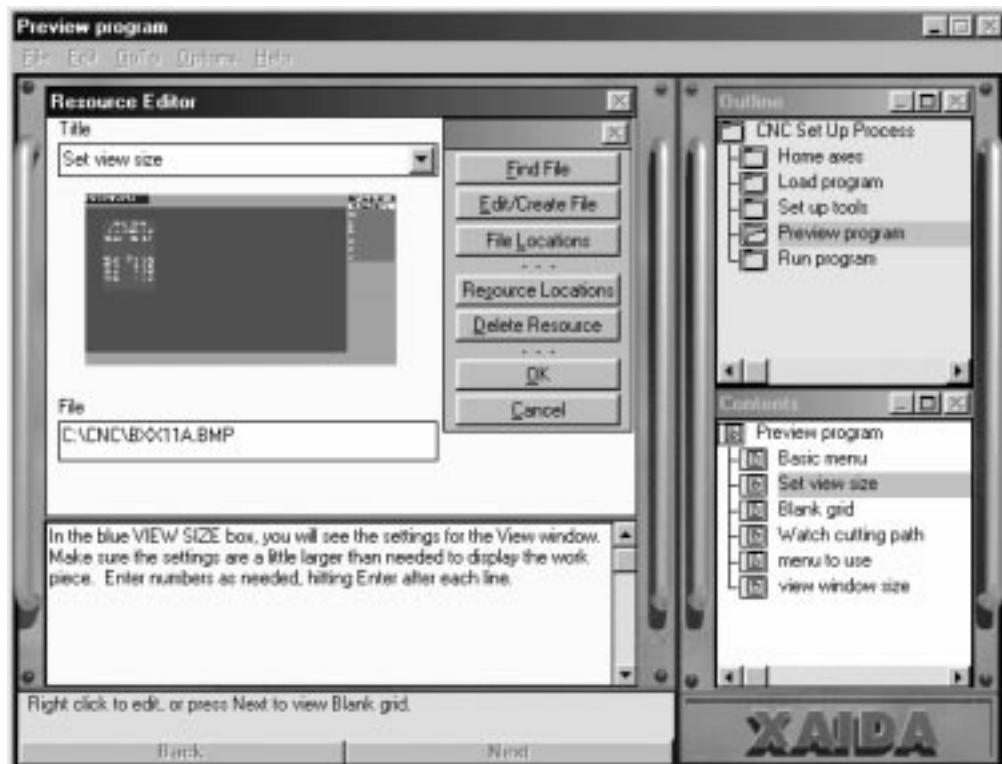


Fig. 2. Resource editor in XAIDA Develop.

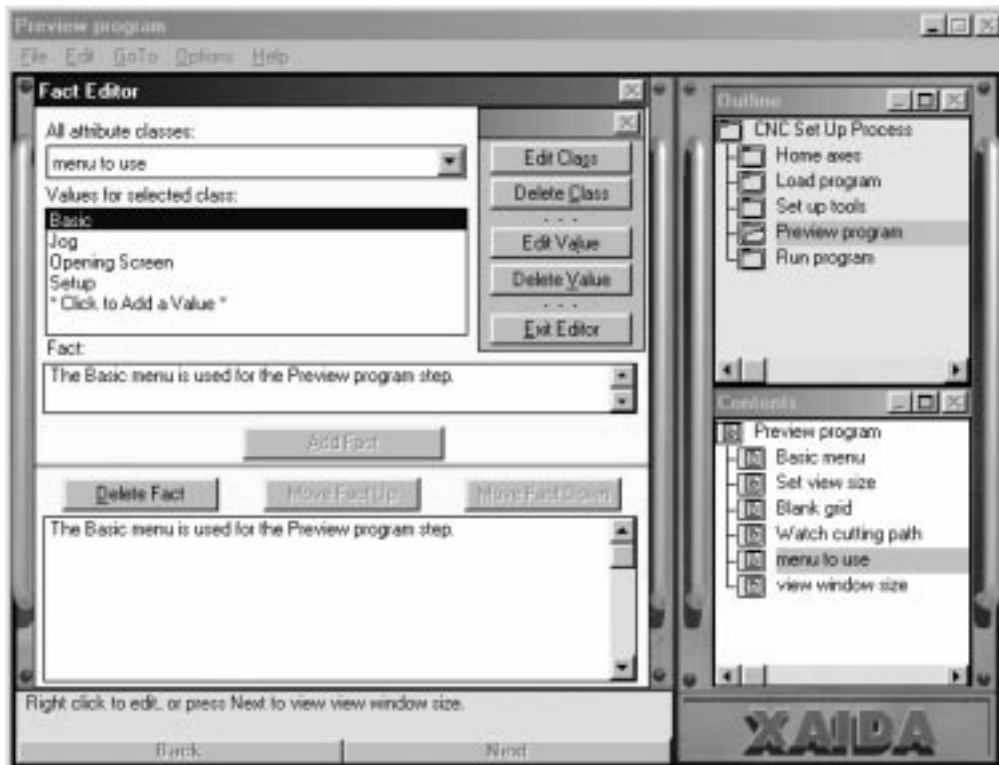


Fig. 3. Fact editor in XAIDA Develop.

exactly as its designers intended, i.e. to teach about the physical characteristics of a device. This is because the structure of XAIDA-PC's templates assumed that the information in its knowledge

base pertained to parts of a system. However, with extra planning, XAIDA-PC can accept other forms of content for teaching purposes.

The authors used the following method to enter

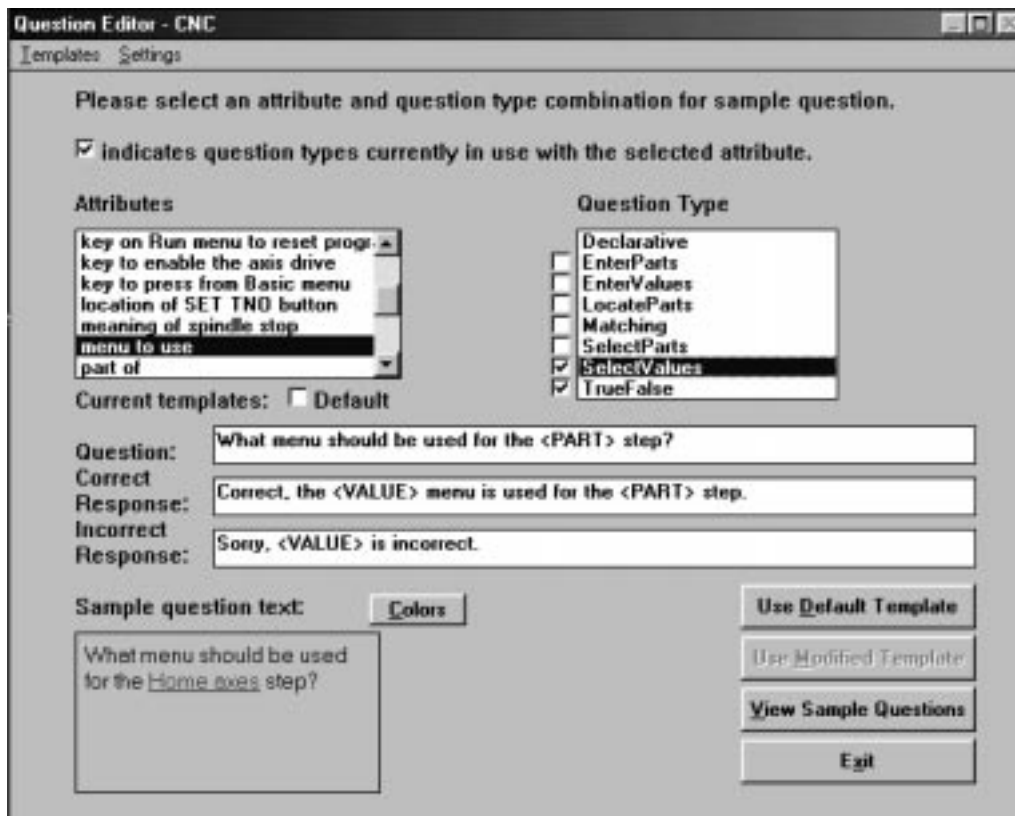


Fig. 4. Question editor in XAIDA Develop.

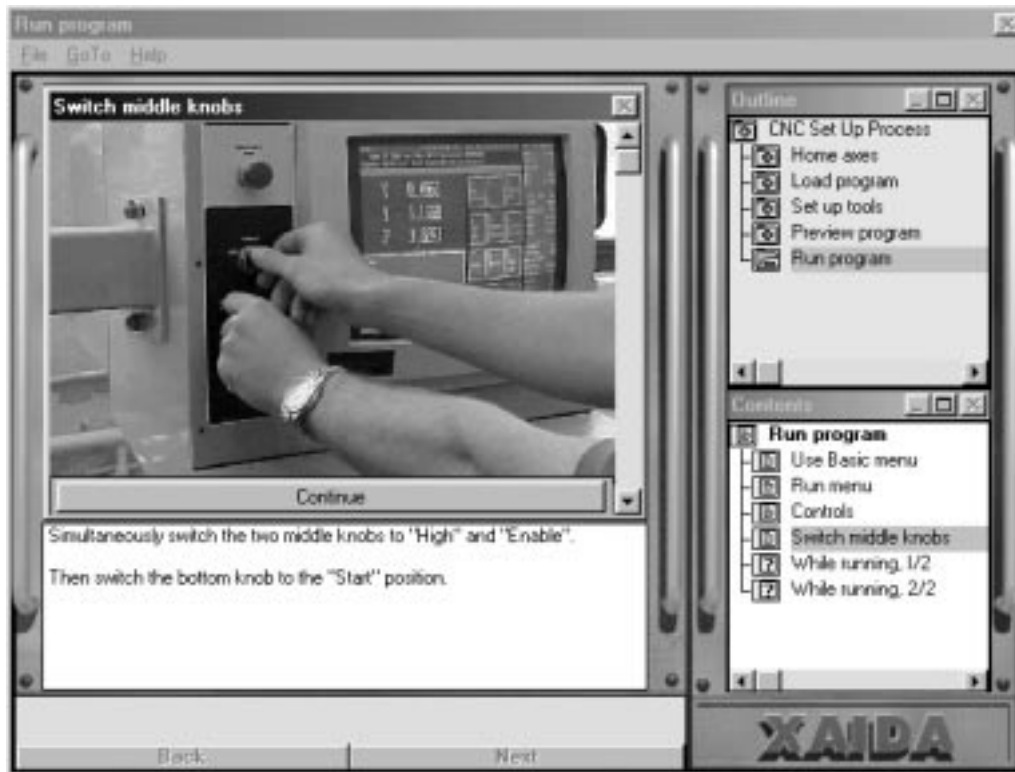


Fig. 5. Sample presentation screen in XAIDA Deliver.

information into an XAIDA knowledge base from an application called 'Develop':

1. The major steps were entered using the Outline Editor (see Fig. 1).
2. Graphics were linked in using the Resource Editor (see Fig. 2).
3. Information to be used for questioning was entered using the Fact Editor (see Fig. 3).
4. XAIDA's built-in templates, which are used for



Fig. 6. Sample practice question screen in XAIDA Deliver.

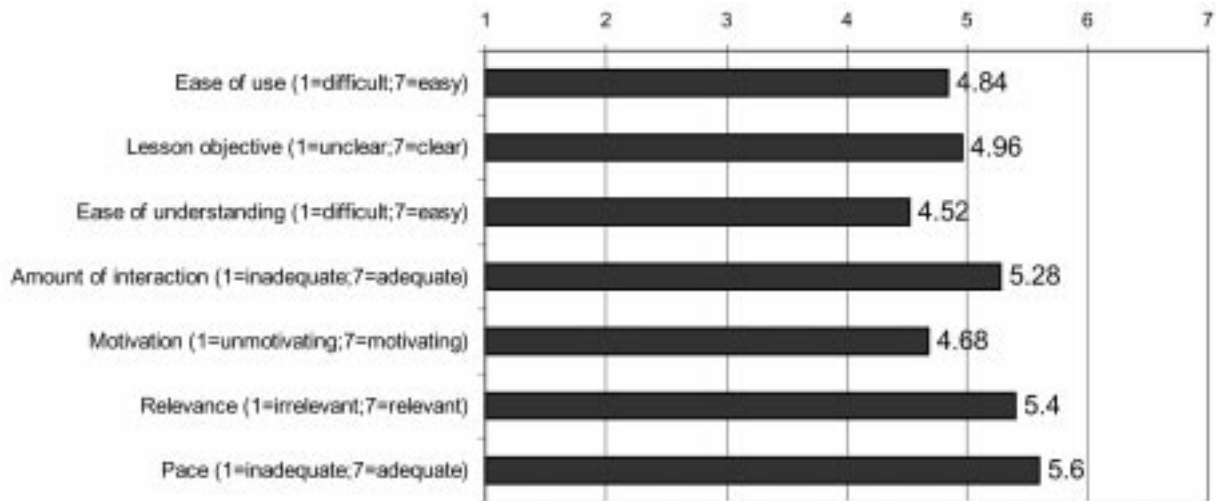


Fig. 7. Average ratings of various aspects of CNC lesson.

presenting facts, asking questions, and offering feedback were modified using the Question Editor (see Fig. 4).

Note that courseware development process using a knowledge-based authoring tool, such as XAIDA, is different from traditional courseware authoring tools, such as Macromedia's Authorware. With traditional tools, the author's main task is to develop an instructional presentation. With knowledge-based tools, the main task is to describe knowledge; the system then generates the presentation. In XAIDA, a program called Deliver generates the instructional presentation. (Figures 5 and 6 show presentation and practice question screens taken from the developed tutorial.)

The tutorial took about 80 person hours to complete including development of multimedia, (primarily digital images). Because the authors were primarily interested in defining and representing procedural information, XAIDA's ability to link to digital audio or video files and to other documents or applications was not utilized.

Tutorial evaluation

In evaluating how the tutorial progressed with students, we were particularly interested in the following issues:

- How long did students take to complete the lesson?
- Did the lesson succeed in its goal of teaching students more about the procedure for setting up the CNC machine?
- What did students think of various aspects of the lesson, such as:
 - * features,
 - * objective,
 - * instructional,
 - * sequence,
 - * difficulty,
 - * repetition,
 - * emphasis on important information,

- * question-and-answer sessions,
- * use of multimedia,
- * interactivity,
- * pace,
- * quality,
- * relevance to their education?

Participants, materials, and experimental procedure

Evaluation of the tutorial involved twenty-five students from a Numerical Control and Computer-Aided Manufacturing class. The students saw a demonstration showing how to set up the CNC machine a few weeks before this study, but were not proficient at the task.

Students received hard copy manual entry forms designed to measure the effects of the computer-based tutorials on subject knowledge and attitudes toward instructional methods and computers. Pretest and posttest forms consisted of a 3-item survey and a subject test. The survey asked students to:

- Choose their preferred method of instruction for that subject,
- Rate their degree of comfort in using computers,
- Rate their degree of familiarity with the assigned subject.

The subject test consisted of multiple choice questions on the lesson content. For example, a question on the subject test was:

When homing the axes, which menu do you use?

- a) setup menu
- b) opening screen menu
- c) basic menu
- d) jog menu

Students were asked to complete a 12-item opinion survey after the posttest. For this survey, students used Likert scales (with anchors from 1 to 7) to rate their opinions about several characteristics of the lesson, including features (e.g. buttons, hot

boxes), question-and-answer sessions, objective, use of multimedia, instructional sequence, interaction with computer, difficulty, pace, content repetition, emphasis on important information, relevance to education, and overall quality.

Students first completed the pretest materials and then began the computer-based lesson, working at their own pace until they completed the entire lesson, including the practice questions, followed by the posttest and the opinion survey.

RESULTS AND DISCUSSION

Tutorial development observations

Use of XAIDA-PC to teach equipment procedure is both feasible and quick. Of particular benefit is XAIDA's question generator feature, which not only automatically generates questions, but also adjusts the content and sequence of practice questions to the needs of individual students (based on their performance). To do this, the system maintains a simple student model for each student and uses it to determine what (if any) question to ask next. Consequently, students who have mastered the material answer fewer questions than students who have not. In addition, for students who require more practice, the algorithms that drive the question generator tend to extend questioning in areas where the student might be having difficulty.

However, using XAIDA-PC to teach about steps of a procedure, rather than parts of a device, limits benefits normally gained from use of the system, particularly concerning the types of questions that the students can be asked.

XAIDA-PC stores factual information in the format '{part, attribute, value}' e.g. {emergency stop button, has function, halt the system}, and the developer defines the parts, attributes, and values for each tutorial. The design of XAIDA-PC created the expectation that a part would be a physical component of a device, and the types of questions it generates maintain this assumption. For example, one question type asks students to locate parts in a diagram. Another requires students to match parts to values according to a specific attribute, such as matching a list of parts to their functions. The result of this is to cause certain question types to become awkward or even unusable if the 'part' slot is repurposed for some other type of entity such as a 'step', e.g. '{preview program, menu to use, basic menu}'. Fortunately XAIDA's Question Editor allows developers to suppress question types by attribute and/or modify system templates to make questions more readable.

In addition, some question types that would have been appropriate for teaching a procedure were not readily available. For example, it would have been useful to have a question that required students to sequence steps. We were able to achieve

this partially by creating an attribute called position, with values first, second, third, and so forth, thus allowing the system to ask multiple choice questions such as 'What is the position of the Preview Program step in the procedure sequence?' However, this format is less satisfying for this application than a drag-and-drop sequencing question format.

In short, what would have been best was a shell designed specifically to represent and teach equipment operation procedures. (The designers of XAIDA developed a Procedures shell to do this, but the software for the development interface was not complete at the time of this investigation.) However, in the absence of a Procedures shell, the Physical Characteristics shell was certainly usable and offered the advantage of adaptive practice.

Student evaluations

The mean time for tutorial completion for the twenty-five students was 27.3 minutes. In addition, an ANOVA test with $\alpha \pm 0.05$ found a significant difference between the pre- and post-lesson test scores.

- Mean pretest score—33.3%
- Mean posttest score—87.9% ($f = 271.37$, $F(0.0500, 1, 46) = 4.05$, $p < .001$).

Figure 7 summarizes student ratings of their learning experience from the attitudinal survey. On average, the student ratings were more positive than not for all measured aspects. That is, the students felt that the lesson was interactive, relevant, adequately paced, and easy to use and understand. However, the ratings were not as high as we would have liked.

The finding of instructional effectiveness in the student evaluations is quite strong, and consistent with results from previous evaluations of XAIDA's instructional effectiveness in other domains [29, 30, 31], although we would have liked more student data upon which to base our conclusions. Also, students' subjective impressions of the instruction were positive overall.

One factor that may have influenced the motivation rating was that students had previously viewed a demonstration on how to operate a computer numerical control machine. Thus, they may have been less enthusiastic about taking the lesson than they might have been otherwise. However, we felt that the lessons were still appropriate for use as a review. As it turned out, based on the results on the pre- and post-tests, this review appears to have been not only appropriate, but also efficacious.

A factor that may have influenced both motivation and ease of understanding the lesson is that the XAIDA software adapts instruction to students by automatically generating practice questions based on individual student performance. It tends to ask about items the student

Table 2. Examples of Types of Manufacturing Engineering Knowledge

Type of knowledge	Example	Possible knowledge representation
Physical characteristics of equipment	Parts of a CNC machine	Semantic net
Equipment theory of operation	Theory of operation of a machine vision system	Rules, cases
Abstract theories	When to use various types of production systems	Rules, cases
Equipment problem-solving procedures	Use of a fault tree for a particular piece of equipment	Tree
Cognitive procedures	Statistical methods, such as computing a standard deviation	Rules, semantic net
Physical procedures	Operation of a particular model of a machine vision system	Semantic net
Application of methods	Deciding which statistical methods to use and how best to employ them for a particular problem; developing ladder logic for a particular programmable logic controller application	Cases
Pattern recognition and classification	Deciding which type(s) of sensors are most suited for a particular application	Cases

does not know about, and it keeps asking until it believes the student has mastered every item of instructional content. (If it was a human instructor, it would come across as being thorough to the point of obsession.) In addition, because the phrasing for the questions and feedback is template-based, the grammar sometimes seems awkward, even when it is technically correct.

CONCLUSIONS AND FUTURE DIRECTIONS

A typical manufacturing engineering curriculum includes courses such as:

- Introduction to Quality Assurance
- Manufacturing and Assembly Processes I
- Introduction to Manufacturing Processes,
- Nonmetallic Materials Manufacturing and Assembly Processes II
- Quality Assurance
- Numerical Control and Computer-Aided Manufacturing
- Electronics Manufacturing
- Computer-Aided Manufacturing
- Manufacturing Automation and Robotics Production and Inventory Planning.

However, it is more useful to view manufacturing engineering knowledge not in terms of courses, but rather, in terms of the types of knowledge and skills taught when developing and using authoring tools for knowledge-based instruction. Table 2 is an example of how one might view the classes of knowledge taught in the list above. It also lists how a knowledge-based authoring tool might represent each type of knowledge. Using this type of analysis, it is easier to see why the design of a particular authoring tool might be more appropriate for some

educational applications than others. Judged on results achieved, in the realm of manufacturing engineering, XAIDA-PC is:

- well-suited for teaching physical characteristics of equipment,
- marginal-but-workable for teaching equipment operation procedures,
- probably inappropriate for teaching theory of operation or troubleshooting.

Our plan is to continue to use XAIDA-PC to teach subjects where it is appropriate, and to explore use of other tools, such as RIDES, to teach other types of manufacturing engineering knowledge. In addition, XAIDA is a prototype, and is not web-based. We would like to locate, or possibly even develop, web-based ITS authoring tools for the types of knowledge needed for manufacturing engineering education.

In addition, we plan to continue to evaluate the developed CNC tutorial, preferably with students who have not already been exposed to the CNC machine. We have a particular interest in assessing to what extent taking the tutorial reduces or improves the quality of time spent with the actual equipment.

Acknowledgements— The Center of Teaching Excellence at Texas A&M University, College Station supported this work through an incentive grant. The project titled 'Computer-based Manufacturing Engineering Technology Education,' aims to improve the student/equipment ratio by integrating computer-based training with laboratory exercises. 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. It is currently available free for educational use. Contact Mr. Bill Walsh, 8930 Fourwinds Drive, Suite 450, San Antonio, TX 78239, USA to obtain a copy of XAIDA.

REFERENCES

1. G. Bengu, Interactive multimedia courseware on manufacturing processes and systems, *Int. J. Eng. Educ.*, **11** (1), pp. 46–57
2. Y. Shiue, B. B. Beard, M. L. Santi, and J. E. Beaini, Integrated laboratory for manufacturing education, *Int. J. Eng. Educ.*, Vol. 15, No. (1), pp. 51–57.
3. D. Byrckett, and O. Ettouney. A model to develop and incorporate a computer integrated manufacturing laboratory into an engineering curriculum, *Int. J. Eng. Educ.*, **12**(4), pp. 272–282.
4. R. Choren, C. Laufer, M. Blois, V. Torres, F. Ferraz, G. Robichez, L. Dafflon, C. J. M. P. de Lucena, and H. Fuks, Using AulaNet for web-based course development, String Processing and Information Retrieval Symposium, *IEEE International Workshop on Groupware*, pp. 322–327.
5. W. Doube, (1995). A browser-based system to support and deliver DE, *28th Annual Frontiers in Education Conference*, (FIE '98), Vol. 1, pp. 479–484.
6. G. D. Magoulas, K. A. Papanikolaou, and M. Grigoriadou, Towards a computationally intelligent lesson adaptation for a distance learning course, *Proceedings, 11th IEEE International Conference on Tools with Artificial Intelligence*, 1999, pp. 5–12.
7. K. A. Papanikolaou, G. D. Magoulas, and M. Grigoriadou, A connectionist approach for adaptive lesson presentation in a distance learning course, *International Joint Conference on Neural Networks*, 1999. (IJCNN '99) Vol. 5, pp. 3522–3526.
8. E. Wenger, *Artificial Intelligence and Tutoring Systems: Computational and cognitive approaches to the communication of knowledge*, Morgan Kaufmann Publishers, Inc, Los Altos, CA.
9. D. Sleeman, and J. S. Brown, (Eds.), *Intelligent Tutoring Systems*, Academic Press, Inc., London, 1982.
10. J. R. Fredericksen, B. Y. White, A. Collins, and G. Eggan, Intelligent tutoring systems for electronic troubleshooting. In J. Psotka; L. D. Massey, and S. A. Mutter (Eds.), *Intelligent Tutoring Systems: Lessons Learned*. Erlbaum, Hilldale, NJ, 1988.
11. D. M. Towne, and A. Munro, The intelligent maintenance training system, In J. Psotka; L. D. Massey, and S. A. Mutter (Eds.), *Intelligent Tutoring Systems: Lessons Learned*. Erlbaum, Hilldale, NJ: 1988.
12. F. Ritter, and W. Feurzeig, Teaching real-time tactical thinking, In J. Psotka; L. D. Massey, and S. A. Mutter (Eds.), *Intelligent Tutoring Systems: Lessons Learned*. Erlbaum, Hilldale, NJ, 1988.
13. W. Regian, Representing and teaching high-performance tasks within intelligent tutoring systems. In H.L. Burns, J.W. Parlett, and C.A. Luckhardt (Eds.), *Intelligent Tutoring Systems: Evolutions in Design*. Erlbaum, Hillsdale, NJ, 1991.
14. K. R. Koedinger, J. R. Anderson, W. H. Hadley, and M. A. Mark, Intelligent tutoring goes to school in the big city, *International Journal of Artificial Intelligence in Education*, **8**, pp. 30–43.
15. D. McArthur, M. W. Lewis, and M. Bishay, *The Roles of Artificial Intelligence in Education: Current Progress and Future Prospects*, RAND Technical Report No. DRU-472-NSF, Santa Monica, CA, November, 1993
16. V. J. Shute, and J. Psotka, Intelligent Tutoring Systems: Past, Present, and Future, In D. Jonassen (Ed.), *Handbook of Research for Educational Communications and Technology*, (pp. 570–600). Macmillan, New York, NY, 1996.
17. M. Urban-Lurain, Intelligent Tutoring Systems: An historic review in the context of the development of artificial intelligence and educational psychology. Available online at <http://aral.cse.msu.edu/Publications/ITS/its.htm>
18. A. Gramopadhye, S. Bhagwat, D. Kimbler, and J. Greenstein, Use of advanced technology for visual inspection training, *Applied Ergonomics*, **29**(5) pp. 361–375.
19. E. Rogers, Y. Kennedy, T. Walton, P. Nelms, and I. Sherry, Intelligent multimedia tutoring for manufacturing education. In D. Budny and B. Herrick (Eds.), *Proceedings of Frontiers in Education Conference*, 1-4 Nov 1995, Vol. 2, pp. 4d1.5–4d1.8
20. D. Shin, E. S. Yoon, S. J. Park, and E. S. Lee, Web-based interactive virtual laboratory system for unit operations and process systems engineering. In *Computers and Chemical Engineering (7th International Symposium on Process Systems Engineering, July 16–21 2000)*, **24**(2), pp. 1381–1385
21. P. Woolf, C. Poli, and I. Grosse, Multimedia systems and intelligent tutors for teaching Design for Manufacturing, In *Proceedings of the 1997 27th Annual Conference on Frontiers in Education*, Nov 5–8 1997, Part 2 (of 3), pp. 711–716.
22. B. T. Check, and A. Y. C. Nee, Developing a design system into an intelligent tutoring system. To appear in the *Int. J. Eng. Educ.* Unpublished abstract available at <http://www.ijee.dit.ie/articles/tobepub07.html>
23. M. M. Dessouky, S. Verma, D. E. Bailey, and J. Rickel, A methodology for developing a web-based factory simulator for manufacturing education. *IIE Transactions*, **33**, 2001.
24. J. Gecsei, and C. Frasson, SAFARI: An environment for creating tutoring systems in industrial training. In *Proceedings of ED-MEDIA 94-World Conference on Educational Multimedia and Hypermedia* (Vancouver, BC, Canada, June 25-30, 1994). ERIC Document No. ED388213.
25. A. Munro, Q. A. Pizzini, D. M. Towne, J. L. Wogulis and L. D. Collier, *Authoring procedural training by direct manipulation*, Working Paper WP94-3. University of Southern California—Behavioral Technology Laboratories, Redondo Beach, CA.
26. P. Y. Hsieh, H. M. Halff and C. L. Redfield, Four easy pieces: development systems for knowledge-based generative instruction, *International Journal of Artificial Intelligence in Education*, vol. 10, pp. 1–45.
27. B. Bell and S. Kedar, When less is more: Supporting authoring and interface building via special purpose task models. In J. Greer, Ed., *Artificial Intelligence in Education*, 1995: Proceedings of AI-ED 95- 7th World Conference on Artificial Intelligence in Education.
28. B. M. Wenzel, M. T. Dirnberger, P. Y. Hsieh, T. J. Chudanov and H. M. Halff, Evaluating subject matter experts' learning and use of an ITS authoring tool. In B. P. Goettl, H. M. Halff,

- C. L. Redfield, and V. J. Shute (Eds.), *Intelligent Tutoring Systems: 4th International Conference, ITS '98*, San Antonio, Texas, U.S.A., August 1998 Proceedings (pp. 156–165). Berlin: Springer-Verlag (Lecture Notes in Computer Science, Vol. 1452).
29. M. G. Casaus, E. G. Gibson, B. M. Wenzel and H. M. Half, Effectiveness of adaptive practice in the Experimental Advanced Instructional Design Advisor (XAIDA). Unpublished manuscript.
 30. B. M. Wenzel and R. Ellis, Evaluation of the instructional effectiveness of interactive multimedia distributed courseware for emergency medical technician training. Unpublished manuscript.
 31. B. M. Wenzel, K. H. Richardson and E. G. Gibson. Assessing the instructional effectiveness of the Experimental Advanced Instructional Design Advisor (XAIDA). Unpublished manuscript.

Sheng-Jen (Tony) Hsieh received his Ph.D. in Industrial Engineering from Texas Tech University, Lubbock, TX. He is currently an Assistant Professor in the Department of Engineering Technology and Industrial Distribution and a member of the Graduate Faculty at Texas A&M University. His research interests include intelligent manufacturing system design, thermal signatures for process and product characterization, and simulation and modeling. He is also the Technical Director for an industry grant to establish the Rockwell Automation Center at Texas A&M University. The Center will support courses related to automation, control, and production systems, and provide a vehicle for education, outreach, and research.

Patricia Yee Hsieh is currently a Senior Training Specialist and web-based training designer in the System Human Resources Office of the Texas A&M University System. This research took place while she was a Senior Instructional Designer at MATCOM. She has 12 years of instructional technology R&D and multimedia courseware development experience, including seven years of experience in designing and developing authoring tools for intelligent tutoring systems. Ms. Hsieh received her MA in Curriculum and Instruction from the University of Texas at Austin and her undergraduate degree with honors from Harvard University.