In the twenty-odd years since educators first began experimenting with computer-assisted instruction (CAI), its potential benefits have been well established by a variety of efforts [1]-[8]. However, among the major factors limiting the application of CAI has been the great investment of faculty time necessary to develop instructional programs or "courseware" for the computer. This has been particularly true of CAI for engineering education. Where CAI has been used, the focus has been more on teaching facts than on developing complex skills such as engineering problem solving. We report here on a CAI application developed at the U.S. Military Academy, West Point, NY, which is an exception to both of these generalities. Our CAI system requires only a modest effort by the faculty, and supports teaching of electrical engineering problem solving methods.

Our CAI system, known as TUTOR301, grew out of a desire to automate the process of "additional instruction" which is traditional at the Military Academy. The West Point cadet's workload (both academic and otherwise) is highly demanding and time driven. A cadet experiencing difficulty in any subject is expected to contact the course instructor for what amounts to a private tutoring session. This is a natural extension of the small-group method of instruction at the Academy. Over the years, it has served well in helping students to master difficult subject material in the shortest possible time.

Every cadet at West Point takes at least one course in electrical engineering. The course taken by those concentrating in the humanities and public affairs disciplines (about 250 per semester) is known as EE301, Basic Electrical Systems. It is understandable that students in this course often experience difficulty with engineering material. Although problem solutions were posted in classrooms, often the students needed further live demonstrations of problem solving methods. As a result, EE301 instructors were spending many long hours providing needed additional instruction.

We noticed that the process of tutoring soon became repetitive; the instructor would lead the student through a problem solution, encouraging as much unassisted effort as possible, and correcting whenever a false step was taken in the solution. It occurred to us that, in many cases, the tutoring could be performed by an interactive computer just as effectively. This would free the instructor to concentrate on students having unusual difficulties, while making the assistance available at all hours right in our dormitories (barracks). It might also overcome the natural reluctance of students to approach an instructor for assistance.

TUTOR301—A Successful Application of CAI for Electrical Engineering Problem Solving

J. M. PULLEN AND R. T. MERCER

Abstract—A computer-assisted instruction (CAI) system has been developed at West Point. The system is used to teach electrical engineering problem solving in an introductory course for nonengineers. The system, known as TUTOR301, takes the place of an individual tutor or coach in problem solving methods. The simple form of solution text employed minimizes faculty time required to prepare solution scripts for the computer. It has been accepted well by students, and appears to improve the academic performance of those students who use it.

Manuscript received January 30, 1987; revised July 20, 1987.

The authors are with the United States Military Academy, West Point, NY 10996.

IEEE Log Number 8717787.

U.S. Government work not protected by U.S. Copyright
swers the question successfully, another question is asked. The process continues until a wrong answer is given; at that point, the instructor explains the concept involved and asks the question again, if necessary demonstrating the method of solution. This continues to the end of the solution, with the result that the student is reinforced in the concepts already understood, retaught those not understood, and exposed to the general method of solution for the problem at hand. This stepwise solution paradigm of tutoring was, again, if necessary demonstrating the method of solution. This continues to the end of the solution, with the result that the student is understood, and exposed to the general method of solution for the reinforced in the concepts already understood, retaught those not program at all!

Making regular use of TUTOR301 had a grade below C relieved the source of difficulty. At the end of the term, no student was available at their own convenience was sufficient incentive to study. It was very gratifying to observe that it also simplified the process of preparing solutions. We were lucky by incorporating any required diagrams in the problem statements. While this limited interaction with the student somewhat, it also simplified the process of preparing solutions. We were lucky in that our notation required only minimal use of Greek letters and subscripts. In these cases we substituted a similar notation such as "BETA" for $\beta$ or "hfe" for $h_{fe}$. We feel that another key aspect of TUTOR301's success is its very simple logic structure. Rather than the elaborate paths through the terminal session commonly available with CAI packages, our program allows only the simple structure shown in Fig. 1: Any amount of text is followed by a single question, which may require either a character or numeric response. Character responses must match exactly (most are single letters, often Y or N). Numeric responses match if within 1 percent of the nominal answer. Whenever the correct answer is not given, an explanation of the correct method (but not the actual answer) is provided. After three reasonable tries are made, the program gives the answer and proceeds to the next text section. (The definition of a "reasonable" numeric answer is one which is within an order of magnitude of the correct value. Unreasonable answers result in messages indicating that the answer is far too large or small, a sense which we feel is important for the student to develop.)

Problem solutions for TUTOR301 are generated using a standard text editor program. The text of the solution is printed out for the student exactly as entered, except for the first character of each line. The first character, which is not displayed to the student, indicates the purpose of the line:

- $T$ solution text
- $Q$ question having numeric solution
- $N$ numeric solution value
- $C$ question having character solution
- $A$ character solution value
- $P$ page eject/clear screen
- $*$ author comment (not shown to student)
- $E$ explanation of correct method (follows each answer but is only printed when response does not match solution value).

The TUTOR301 program itself is quite simple. Although made more elaborate and "user friendly" since its inception, it still consists of less than ten pages of Fortran code, more than half of which is documentation. The general structure of the program is shown in Fig. 2. A sample session and the solution file from which it was generated are given in the Appendix.

Enhancements to the program since the original version have had two purposes: to improve data collection about usage, and to encourage cadets to use the system. For the former purpose, features were added to capture the user's account number, time and data of usage, and individual problems attempted, for statistical analysis (the log of "successful completions" is still available for classroom instructors). For the latter purpose, the system now includes a "bulletin board" for course memos, allows the user to copy the session to a printer if a nonprinting terminal is used, makes "escape" from frustrating problems easy, and ends the terminal session with a "fortune" quip for the day.

III. The TUTOR301 Program

The success of our CAI system is undoubtedly due in part to the computer support provided at the Military Academy. The academic area and barracks at West Point are served by a local-area network which supports terminals in all academic buildings, faculty offices, and barracks. (Eventually each cadet room will have access to the network.) The instructional support computers on this network are PRIME 850 superminis with shared disk storage. System reliability is very high due to the ability to shift users to another processor if any unit fails. Response time is typically less than a second, although it occasionally degrades under peak usage.

The only limitation posed by our computer support is that graphic output is not available on most terminals. This was easily overcome by incorporating any required diagrams in the problem statements. While this limited interaction with the student somewhat, it also simplified the process of preparing solutions. We were lucky in that our notation required only minimal use of Greek letters and subscripts. In these cases we substituted a similar notation such as "BETA" for $\beta$ or "hfe" for $h_{fe}$. We feel that another key aspect of TUTOR301's success is its very simple logic structure. Rather than the elaborate paths through the terminal session commonly available with CAI packages, our program allows only the simple structure shown in Fig. 1: Any amount of text is followed by a single question, which may require either a character or numeric response. Character responses must match exactly (most are single letters, often Y or N). Numeric responses match if within 1 percent of the nominal answer. Whenever the correct answer is not given, an explanation of the correct method (but not the actual answer) is provided. After three reasonable tries are made, the program gives the answer and proceeds to the next text section. (The definition of a "reasonable" numeric answer is one which is within an order of magnitude of the correct value. Unreasonable answers result in messages indicating that the answer is far too large or small, a sense which we feel is important for the student to develop.)

Problem solutions for TUTOR301 are generated using a standard text editor program. The text of the solution is printed out for the student exactly as entered, except for the first character of each line. The first character, which is not displayed to the student, indicates the purpose of the line:

- $T$ solution text
- $Q$ question having numeric solution
- $N$ numeric solution value
- $C$ question having character solution
- $A$ character solution value
- $P$ page eject/clear screen
- $*$ author comment (not shown to student)
- $E$ explanation of correct method (follows each answer but is only printed when response does not match solution value).

The TUTOR301 program itself is quite simple. Although made more elaborate and "user friendly" since its inception, it still consists of less than ten pages of Fortran code, more than half of which is documentation. The general structure of the program is shown in Fig. 2. A sample session and the solution file from which it was generated are given in the Appendix.

Enhancements to the program since the original version have had two purposes: to improve data collection about usage, and to encourage cadets to use the system. For the former purpose, features were added to capture the user's account number, time and data of usage, and individual problems attempted, for statistical analysis (the log of "successful completions" is still available for classroom instructors). For the latter purpose, the system now includes a "bulletin board" for course memos, allows the user to copy the session to a printer if a nonprinting terminal is used, makes "escape" from frustrating problems easy, and ends the terminal session with a "fortune" quip for the day.

IV. Results

As of this writing, TUTOR301 has been in use for five semesters, and as far as the faculty and students are concerned, it is an unqualified success. Several problem solutions have been prepared for each lecture, with more added each semester, so that there are now several hundred problems available. After preparing two or three problems, a faculty member is sufficiently experienced that each new problem takes about an hour to code, enter, and test. It is thus possible to add new problems each semester with a modest effort, divided among the instructors teaching EE301. With a student population between two and three hundred, we have recorded over 6000 completed problems per semester. Clearly, the program is successful beyond our expectations.

As described above, the overall improvement in test scores after TUTOR301 became available was clear. We thus are certain that adding the "computer tutor" to the general course environment enhanced learning of the entire group. We are also certain that the system, to a large extent, takes the place of a faculty tutor for students having difficulties with the course material.

An open question is whether all students who make frequent use of TUTOR301 perform better, on the average, than those who do
not. It is our impression that this is the case, so we have used the program’s data collection facilities to analyze grades for 120 students taken from the middle of the class (cumulative averages between “C” and “B”). We found that those cadets (about half of the group) who used the program more than 20 times during the term did in fact earn better grades than would be predicted by their grade-point averages, although the difference (a little over a tenth of a grade point) was not statistically significant for the sample size. Our results here may be skewed by the fact that when students work in groups on the system, only one account is recorded. Also the printed solutions are at times shared among students, which amounts to noninteractive use of the system without any record of its use.

V. CONCLUSIONS

We enthusiastically advocate the “keep it simple” approach embodied in TUTOR301. Although there are undoubtedly cases where a more elaborate control structure could be advantageous, the overall benefits of simplicity in preparing solutions are great. Clearly our method is not applicable to all learning environments, but within the scope of basic electrical engineering problem solving we have yet to find a case where the strategy embodied in Fig. 1 is less than adequate. We believe this is because that strategy reflects an excellent approach to coaching problem solving, with or without CAI. The strategy is one which could and should be expanded to a wide range of scientific and engineering problems. (To do this, a graphics capability would be useful but not necessary. It is, however, important that the computer terminals used support a character set adequate for the symbology of the disciplines involved.) We are willing to make copies of our program available upon request.

We also believe that the success of TUTOR301 lies at least as much in its compatibility with human nature as in its approach to problem solving. The combination of immediate feedback with incentives to practice problem solving is undoubtedly a powerful aid to learning (and, thus, to teaching). We are convinced that our method is applicable to a broad class of introductory engineering topics. While there is surely a limit to the level of abstraction which can be dealt with successfully in this way, we feel that the approach we described in this paper will be found to have merit in any course where concrete principles and structured problems with numeric solutions are included.
**REFERENCES**