Pedagogical and Cost Effectiveness of Computer-Assisted Learning in Control Systems Education

R. Welch, K.R. Goheen
Ingenia Communications Corporation/Ryerson Polytechnic University

Abstract

This study examines the use of courseware to teach feedback control systems material to undergraduate mechanical and aerospace engineering students. Courseware for teaching the frequency response portion of the course was developed using Asymetrix Multimedia Toolbox and subsequently tested on a third year (Junior) level class. Academic results and attitudes of the students were examined and statistically compared to those of a control group. It was found that there was very little difference in the academic results and attitudes of students in the computer group and students in the lecture group. In addition, the effectiveness of the courseware was not related to students' previous computer experience. A cost analysis indicated that in most scenarios the use of courseware to teach control systems would be more costly than traditional lectures.

1.0 Introduction

Like most types of Computer-Assisted Learning (CAL), courseware is rarely used in engineering faculties. Most modern courseware uses multimedia techniques to teach material. Individual users proceed through the material at their own pace, and can often adapt the manner or the order in which topics are learned to suit their learning style. Courseware is used more and more often in education, both at the pre-university level and in higher education. Very little engineering courseware exists, however, which is unusual for a discipline that has otherwise embraced computing technology.

A subject areas best-suited to be a testing ground for CAL in engineering education is control systems. This is in great part due to the highly mathematical nature of the material and the need for numerical graphical representations. Use of computer technology in the practice of control engineering is widespread.

As early as the 1970s, computers were being used to assist teaching control systems. Broome and Woolvet[1] created "[a] program. . . which permits interactive control system design, suitable for use by students either to run tutorial exercises as a back up to lecture material, or to integrate with laboratory work". The program was written in FORTRAN IV for 8K computers such as the Honeywell H3 16. The use of computer for control systems education has since then become widespread. According to a survey of control systems curricula by Feliachi[2], "[software packages of a wide variety are being used by most schools. The most popular packages (in frequency of usage) are MATLAB, MATRIX-X, ACSL, DESIRE, and CC". These packages are typically used in conjunction with the traditional lecture format. Feliachi did not identify any software packages being used to teach the material other than to reinforce the material taught in the lectures.

These programs are part of the third and fourth layers of four described by Schaufelberger[3] as making up the body of control systems teaching software. Each layer represents a different type of computer application, and stacked together they provide the complete teaching/learning environment. The layers are:

(1) "specialized training programs"
(2) "minitools for small tasks such as drawing the graph of a function"
(3) "tools such as MATLAB or simulation environments for solving small to medium problems"
(4) "fill grown commercial tools (professional MATLAB, CTRL_C, ACSL, SIMNON, Mathematical, etc.) "

Some specific-application packages that make up the second layer of Schaufelberger's structure exist, including a circuit design and testing program 4 multimedia development tools[5], a linear control systems design program [1], a package for PID design and tuning[6], and a digital control simulator [7]. What maybe most lacking is the first layer - the development and evaluation of specialized training courseware.

The effects of using CAL to teach control systems material have not been determined. Limited studies, such as those cited above, have been published, but these tend to focus on the development of the CAL tools rather than on their effectiveness. None of these have compared the academic results and attitudes of students using CAL to learn the material to students attending lectures. For example, Wong et al[6] based their results on observation and information contact, rather than on any structured evaluation. Hofer and Roggiero[8] postulate that "evaluationsare more easily controlled in a laboratory environment. This is no doubt why very few performance based, summative training evaluations are being performed in applied domains," such as engineering. Another reason given is that "[e]valuations require considerable discipline and cooperation from all the parties involved, including students, instructors, managers and evaluators."

Baird and Silvern[9] called into question the viability of testing, and decided to assess its effectiveness as an evaluation tool. They found that the mode of testing used has little effect on the results. In other words, a paper test is a reasonable means of testing CAL students, just as a computer test will not skew the results for traditionally-taught students. The researchers concluded that, for the material their non-engineering CAL application taught, "mixing computer study with paper-and-pencil testing does not bias the outcome of the test". Clark and Craig,[10] however, noted that the instructional methods, such as interactivity, might influence evaluation results more than the technology.

A question debated by researchers surrounds the involvement of students in the evaluation process. Reiser and Kegelman, [11] in a comprehensive review of current evaluation methods, found that" [v]ery few models suggest that students serve as evaluators," and recommended that "students should be observed as they use a software program, and conclusions about the quality of the program should be based, in part, upon what the evaluators observe". In addition to recording students reactions and observations while using the program, the researchers recommended that "evaluators should assess how much students have learned as a result of using a particular program".

The actual composition of the evaluation program varies from study to study. To evaluate a non-engineering CAL application, Janniro[12] compared the results of 14 students learning via CAL to a 15-student classroom control group learning the same material. The results were for the two groups were compared statistically using such tools as a two-tail t-test for significance. The focus here was heavily on the academic outcomes rather than the affective outcomes.

Similarly, Brown and Cutlip[13] investigate the results of teaching FORTRAN programming to engineering students by comparing the results of an 18-student courseware group to a 22-student lecture group. Here, both academic and affective outcomes were considered, using both a test and an attitude questionnaire. Hofer and Roggiero[8] also combined questionnaire data with performance measures to evaluate the use of CAL to teach technical material, and found that it was a reasonable means of evaluation while difficult to implement and manage.

According to Outang et al[14], in their study of pre-university CAL use, CAL had a positive effect on the test subjects' academic achievement. The evaluation methods used are not discussed in the published information. Janniro[12] used a combination of pre-testing and post-testing of CAL and control groups, and found that a group of students using CAL "achieved significantly higher post-test scores," and "also required less than half of the scheduled instructional time compared to those students in the classroom group". He concluded that CAL "appears to be most effective when systematically developed and the course content is pedagogically sound," and recommended that developers apply information-processing learning theories and screen design strategies to improve CAL.

Aminrnansour [16] found that CAL contributed to better comprehension and retention in civil engineering students. Here again, the methods used to arrive at this conclusion are not discussed. Hodgson [4] found that it was "evident from the way students take to this self-learning environment. . . that workstation learning will become an ever more important and powerful tool in engineering education". Similarly, Plank et al '[5] found positive results when using CAL in engineering, but base these results on experience rather than on an evaluation.

Brown and Cutlip[13] compared the results of engineering students learning FORTRAN through CAL to those in a classroom control group, and found that the evaluation "failed to yield a significant difference between the two groups". However, they also found that "the experiences of the students in the CBE group resulted in an increasingly negative attitude about the instructional use of computers".

This paper examines the impact of using CAL to teach the frequency response portion of controls systems to undergraduate engineering students. The purpose of the experiment was to determine whether there was any difference in terms of the academic results or the resulting attitudes between students learning material from courseware and those learning from lectures. Differences in learner outcomes can be separated, according to Brown and Cutlip[13], into "academic outcomes" and "affective outcomes". The academic outcomes are those related to the amount learned by thes tudent as a result of using the CAL technologies (e.g., final course grades). The affective outcomes relate to the changes in the students' attitudes and perceptions as a result of participation in CAL projects (e.g., changes in feeling towards computers). The paper concludes with a detailed cost analysis of each mode of instruction.

2.0 Methodology

2.1 The 86.352 Course: Feedback Control Systems

The experimental part of this study was conducted with students enrolled in the course 86.352 Feedback Control of Dynamic Systems [17]. This is a mandatory course for all undergraduates in the Mechanical and Aerospace Engineering programs. It is similar in structure and content to the feedback controls courses offered in most faculties of engineering, covering the basics of control in the time and frequency domains. Most of the students in the course are in their third year of a four-year Bachelor's of Engineering program.

- Early in the course, the students were apprised of the nature of the study and of the process. Most of the students volunteered to participate, and of these, forty were randomly selected. This selection took place in the early part of the term, so that it was not affected by the interim results of the students. The students who would be participating in the experiment were advised, but were not told whether they were in the computer or lecture group until the day before the start of the study.

Throughout the study, steps were taken to ensure that the effect of the study on the students' academic results would be minimized. The students were given an incentive to participate, as those who scored better on the test given at the end of the study period than on their midterm examinations would have the result counted towards their midterm grades. The results of the test and the post-experimental questionnaire were kept confidential and references to the identities of the students were removed before analysis. Students were told that if they felt that their understanding of the course material had been compromised by the experiment, they would be given extra tutorials.

2.2 CAL Software Development

Several development environments were considered. The decision to use Asymetrix Multimedia Toolbook was made because of the features of the software, the relatively low price, the compatibility of the software with other packages, the unlimited run-time license, and the upgrade capability. The additional software packages used included MATLAB, for numerical analysis and simulation, and LView, for image processing.

The development and testing of the software took place over a period of about six months. The content of the courseware was based on the textbook Franklin et al. [18] and the course notes Goheen[19] for 86.352, so that the courseware would be compatible with the material covered in the lectures, facilitating comparisons between the two delivery methods.

The final courseware replaced a two-week section of the course. The module covered frequency response material, including frequency domain analysis and design techniques, Bode plots and stability margins. This portion was chosen because it was well-suited in length and position in the course, and relatively independent from the other course material. The design of the courseware considered the results of previous studies of the design of multimedia for educational use.

2.3 Student Examination and Questionnaire

The twenty students in the courseware group spent two weeks in the computer laboratory instead of in the lectures. During these sessions, they were given assistance with the courseware when required, but not with the course material itself. Most of this group completed their exploration of the courseware in less time than the scheduled lecture time.

Both the computer group and a group composed of twenty students who had attended the lectures wrote a one-hour test on the material. The test consisted of two questions, one on steady-state output in the frequency domain, and the other on Bode plots and stability margins.

Before the start of the test, the students were asked to complete a questionnaire that aimed to assess such factors as familiarity with computers, learning style, and feelings about the courseware. The questionnaire consisted of twent y questions, many of which were multi-part.

3.0 Statistical Results and Student Comments

3.1 Analysis of Data

The data considered in this experiment included academic data such as the test results, the midterm examination and final examination grades, the overall course grade, and the cumulative grade point averages of the students. Additional student information was obtained from the students' responses to the voluntary pre-test questionnaire.

The data was compiled, sorted and referenced to record numbers assigned to each student. The questionnaire responses obtained were coded into discrete values wherever possible to facilitate analysis. Some of the data sets were used to examine the results of the courseware use, whereas others were included to determine that there were no significant differences between the profiles of the courseware and lecture groups. This was done so that there could be reasonable certainty that differences between the results for the two groups were due to the experiment rather than due to pre-existing differences.

Statistical techniques were used to find significant relationships amongst the forty-five sets of data. First, some approximate comparisons were made to see if there were any immediately obvious relations between data sets. Then, statistical analysis was used to compare 165 of the 990 possible pairs of data.

Each of the 165 comparisons was examined for significance, as for any apparent relationship there was always the possibility that it was in fact a chance relationship being observed. Two types of tests were employed for this analysis. The first was the t-test, used when one of the two values being compared was on a continuum and the other was always one of two discrete values. The second type of test used was the correlation test. Here, both data sets compared were continua, and the test was used to find out how the values related to each other.

Tables 1 and 2 list the t-tests and correlations that showed significance. Table 3 lists other comparisons which may be considered interesting because they did not show a significant relationship between variables when one might have been expected.

3.2 Results of Statistical Analysis

From the results of the t-tests, it was found that the aerospace engineering students obtained higher grades on the final exam and the overall course. They also rated the course material as being easier. This was true for both the lecture and courseware groups.

There was a significant difference in results for the rating of the ease of the course material by the computer group and the lecture group. The lecture group rated the frequency response material as more difficult than did the computer group. There was no significant difference in overall grades between the two groups. It maybe that the computer tutorial software made the material easier to learn, or perhaps that it made the students more comfortable with the material, thus bringing them to believe that the material was easier than it was.

The students' years of computer experience, their enjoyment of working with computers and their knowledge of computers were all closely correlated. It seems reasonable to expect this result, that the length of time that students have been using computers would be closely tied to how well they know them and how much they enjoy using them. It is also not surprising that students would be more knowledgeable about computers if they enjoyed working with them, and vice versa.

The ease of the course material as rated by the students was closely related to the amount of time spent studying the frequency response material, and to the midterm grades. Students who found the material more difficult also spent more time learning the material. Also, the more difficult they judged the material to be, the lower were their midterm grades. This is true only of the midterm examination, and it should be noted that students were aware of their midterm grades when they answered the questionnaire.

The amount of frequency response material the computer group students thought they had been taught by the software was inversely related to their preference for future software tutorials over lectures. The more the students felt the tutorials taught them, the more they preferred lectures over future tutorials. This result seems counterintuitive, as one might expect that students who felt they were taught all of the necessary material through the tutorials would lean towards tutorials for future use. It is possible that instead of one of these factors being the result of the other they are both the results of some third factor.

Additionally, some of the results were interesting because no relationship was found where one might have been expected. One of the most interesting of these cases is that, except for the rating of the ease of the course material, there were no significant t-test results for comparisons between the computer and lecture groups.

There were no significant correlations related to students' experience with, knowledge of or enjoyment of computers, except amongst those factors themselves.

The computer group students' stated amount of learning from lectures was not related to their performance on the test or in their course, to their liking of the software or to their lecture/computer preference. Furthermore, the students' rating of the ease of the course material was only related to their midterm examination grade, and not to any other grades.

The amount of time students spent learning the material was not related to their major grades or to their preference for lectures or computer tutorials. Also, the amount of time the computer group students spent learning the material was not related to their liking of the material or the amount they felt it taught.

Overall, most of the comparisons showing significance did not serve to demonstrate differences between the lecture and computer groups. The academic results and resulting attitudes were for the most part independent of the method by which the material was made available to the students.

3.3 Student Comments

The questionnaire offered the participating students several opportunities to provide comments on the software and on CAL use in engineering. Four of these questions were directed to the. computer group only and were intended to offer them an opportunity to elaborate on their impressions of the software and to make suggestions for improvement.

The comments were generally positive. There were a large variety of responses and no remarkable trends. The only comment which occurred in any frequency was that students would like to see the tutorials implemented as part of a package including lectures, instead of replacing them completely.

4.0 Cost Analysis

A cost-comparison of the lecture and CAL methods for teaching the controls course was carried out. The purpose of this comparison was to determine the difference in cost to the university. Only the difference in cost or revenue were considered. Two models were used. The first of these is based on a single year of course delivery. The second assumes a five-year period. The figures used for the lecture model are based on a cost analysis of Carleton University's itv program.[20]

The comparison initially assumes that there are one hundred students in the course. A maximum number of one hundred students per section was assumed. These students pay the same fees whether they are in the lecture group or the computer group, and are also identical where use of university services such as the library are concerned. The printing and publication costs for course handouts are assumed to be the same. Furthermore, outside of the actual course computer laboratory time for the computer model, they use computer facilities equally. The revenue to the university from students' spending on books, food and parking is assumed identical. The course time for both groups is three hours per week. For the computer model, they are assumed to be divided into two equal groups of fifty students.

It is assumed that a professor earns $80,000 per annum, plus fifteen percent benefits. Professors spend fifty percent of their time lecturing. For the lecture model, this represents two credits of courses per annum, or four half-credit courses. For the computer model, there is still a need for the direction of a professor. It is assumed that a professor will supervise six credits of computer-based courses, or twelve half-credit courses. It is assumed that a single teaching assistant works with both halves of the computer group. There is no teaching assistant for the lecture model. The teaching assistant is assumed to cost $3300 per term.

The computer laboratory contains fifty computers. These are assumed to cost one thousand dollars each per annum, including hardware, system software and all peripherals, printers and network servers. Computer management and maintenance personnel are assumed to cost $35,000 per annurn per three hundred computers. The computer software is assumed to take one half-year to program. The programmer is initially assumed to cost the university a total of $25,000 per annum. This is intended to represent an average between a professional programmer and an externally-funded graduate student. For the five-year model, it is assumed that two weeks per year are required for updating the software.

It is assumed that in the lecture room, there is one student per fifteen square feet of space.[20] Computer laboratories in the Faculty of Engineering at Carleton University average approximately twenty square feet of space per student. The rooms are available for use fifty-seven hours per week (twelve hours per day Monday to Thursday, nine hours on Friday), but the lecture room has a sixty percent net utilization rate, meaning that each seat is in use on average for only sixty percent of the available time [20], and the computer lab has an eighty percent net utilization rate. This means that the classroom is 1500 square feet, and is used for this course 8.77 percent of the time. The computer laboratory is 1000 square feet, and is used for the course 13.16 percent of the time. The capital cost of the classroom is assumed to be thirty dollars per square foot per annum, including building and furniture costs.[20] The capital cost of the computer laboratory is assumed equal, but with the addition of the computer capital described above. Maintenance, including cleaning, electricity and heat, is assumed to cost eight dollars per square foot per annum for the classroom[20], and ten dollars per square foot per annum for the computer laboratory due to additional electricity and connectivity costs. The classroom is assumed to have audio and video equipment with a depreciation cost of five thousand dollars per annum.

According to the assumptions above, the computer model would cost $36,245, approximately $15,000 more than the traditional lecture format for the one-year case. The one-year model assumes that all of the development cost is borne in the first year. Because of that assumption, the annual cost differential for the one-year model is higher than for the five-year model or for any other longer term model.

For the five-year case, the development cost is distributed over a five year period. For those five years, the computer model would cost a total of $115,071 or about $30,000 more than the lecture model, or $6,000 per year. The cost differential is smaller than the one-year model because the development cost is only substantial in one of the five years. The differential would be further reduced by considering an even longer model, but based on the current set of assumptions, would never disappear completely.

The variation of cost versus the number of students can be seen in Figure 1. The cost of the computer mode is always above that of the lecture mode. It grows more rapidly as the number of students is increased because the per-student cost increment of the computer model is greater. The step which can be seen in all four curves represents the point where two sections are required. It is notable that the computer cost is closest to the lecture cost when the number of students per section is lowest (at the lowest number of students and immediately after the jump to two sections) because the cost of adding another section to the lecture model is greater than that of adding another section to the computer model.

The cost variation with respect to changes in the development cost can be seen in Figure 2. Here the first year development cost was varied, and the five-year development cost was calculated based on the first year cost with the addition of two weeks per year of the programmer cost. The minimal development cost might represent a case where the software is purchased rather than developed, and the purchase cost is borne by students much as the cost of a textbook. The upper-end cost represents the cost of having the software developed in-house over a full year by a professional programmer. Alternative models in the mid-range include decreased development time, development by graduate students, development with the assistance of external finding, development costs offset by sales to other institutions and external purchase of software. When the development costs are minimized, the cost of the lecture and computer models are very close, although the cost of the computer model is slightly higher. As the development cost is increased, the lecture model cost remains unchanged, and the cost differential grows.

The change in cost relative to the computer use rate is shown in Figure 3. The computer use rate is the fraction of the net use time of the computer facilities for which they are used by the course. As discussed above, the course is assumed to use the computer facilities for six hours per week (two sections of students for three hours each). The net use time is calculated by finding the total number of hours for which the computer facilities are available per week and multiplying it by a net utilization rate. The base assumption was that the computer laboratory was available for fifty-seven hours per week, and that it was used for eighty percent of that time. This meant that the course would be using the computer facilities for 13.16 percent of the total time that they are used. This rate was varied from 8.5 percent to twenty-five percent. The low rate might represents a model where the computer is used on weekends as well as during the week. The high rate might represent a case where the computer lab is underutilised by other groups. When the computer use rate is low, the cost of the computer model is close to that of the lecture model. When the use rate is high, the cost differential is much greater.

5.0 Conclusions

This experimental study of students in a third year undergraduate controls course showed that the use of CAL instead of traditional lectures was equally effective in terms of academic results. As well, the effects on students' attitudes were minimal. The only significant affective result was that the students using CAL instead of attending traditional lectures found the material easier to learn than did the lecture students, although the grades obtained were similar. The results indicate that the effectiveness of CAL is not biased towards those students who are most comfortable with computer use, and that students can learn well from CAL whether or not they are very familiar with computers or enjoy working with them. It is important to note, however, that the students participating in this experiment were all engineering students; as such, the spectrum of computer experience is smaller and weighted towards the high end.

The cost analysis suggested that the computer tutorial model developed would cost the university more to provide than the lecture model. This cost differential was reduced, but not eliminated, by considering a longer period of employment of the model. The resulting cost differential is altered by changing the assumptions used to develop the models (or, the factors under the control of the university administration), but in most cases the computer tutorial model was more expensive.

Acknowledgements

This study was supported by funding from the Natural Sciences and Engineering Research Council of Canada.

REFERENCES

1. Broome, D.R. and Woolvet, G.A. (1976). Mini-Computer Implementation of Interactive Programs for Teaching Linear Control Systems Analysis. The International Journal of Mechanical Engineering Education (Vol. 4, No. 1, pp. 25-36). London, United Kingdom: Institution of Mechanical Engineers, University of Manchester Institute of Science and Technology, Mechanical Engineering Publications Limited.

2. Feliachi,A.(1994). Control Systems Curriculum National Survey. IEEE Transactions on Education (Vol. 37, No. 3, pp. 257-263). New York, New York: IEEE.

3. Schaufelberger, W. (1990). Design and Implementation of Software for Control Education. IEEE Transactions on Education (Vol. 33, No. 3, pp. 291-297). New York, New York: IEEE.

4. Hodgson,J.M.(1992). A Workstation Self-Learning Programme. International Journal of Electrical Engineering Education (Vol. 24, pp. 357-373). Manchester, United Kingdom: Manchester University Press.

5. Abbanat, R., Gramoll, K. and Craig, J. (1994). Use of Multimedia Development Software for Engineering Courseware. In T. Ottmann and I. Tomek (Eds.), Educational Multimedia and Hypermedia, 1994: Proceedings of ED-MEDIA 94- World Conference on Educational Multimedia and Hypermedia (pp. 625). Charlottesville, Virginia: Association for the Advancement of Computing in Education.

6. Wong, Y.-K., Chan, S. P., and Cheung, S.-K.(1 990). Application of Learning Theory to a Computer Assisted Instruction Package. International Journal of Electrical Engineering Education (Vol. 27, pp. 237-246). Manchester, United Kingdom: Manchester University Press.

7. Marcos, M., Wellstead, P.E. and Sandoval, J.M. (1991). A Real-Time Software Package for Teaching Digital Control. International Journal of Electrical Engineering Education (Vol. 28, No. 1, pp. 5-7). Manchester, United Kingdom: Manchester University Press.

8. Hofer, E.F. and Roggiero,F.T.(1990). Computer-Based Flight Training Evaluation. G. Goos, J. Hartmanis, D.H. Norrie and H.-W. Six (Eds.), Lecture Notes in Computer Science: Computer Assisted Learning - 3rd International Conference, ICCAL '90 (pp. 309-319). Berlin, Germany: Springer-Verlag.

9. B~63V.E. and Silvern, S.B. (1992).' Computer Learning and Appropriate Testing: A First Step in Validity A ssessment. D.W. Spuck and W.C. Bozeman (Eds). Journal of Research on Computing in Education (Vol. 25, No. 1,pp. J 8-27). Eugene, Oregon: International Society for Technology in Education.

10. Clark, R.E. and Craig, T-G. (1991). Research and Theory on Multi-Media Learning Effects. M. Giardina (cd.,), interactive Multimedia Learning Environments - Human Factors and Technical Considerations on Design Issues (pp. 19-29) Berlin, Germany: Springer-Verlag.

11. Reiser, R.A. and Kegelman, H.W. (1994). Evaluating Instructional Software: A Review and Critique of Current Methods. Educational Technology Research and Development (Vol. 42, No. 3, pp. 63-69). Washington, District of Columbia: Association for Educational Communications and Technology.

12. Janniro, M.J. (1993). Effects of Computer-Based Instruction on Student Learning of Psychophysiological Detection of Deception Test Question Formulation. Journal of Computer-Based Instruction (pp.. 58-62). Columbus, Ohio: Association for the Development of Computer-Based Instructional Systems.

13. Brown, S.W. and Cutlip, M.B. (1987). Computer-Based Education in College: A Study of Cognitive and Affective Outcomes. Proceedings of the International Conference on Computer-Assisted Learning in Post-Secondary Education (pp. 71-76). Calgary, Alberta: The University of Calgary.

14. Outang, J., Gerlach, G., Bieger, G. and Mikkelsen, V. (1993). Mets-analysis: The Effectiveness of CAI in Elementary Education. In H. Maurer (Ed.), Educational Multimedia and Hypermedia, 1993: Proceedings of ED-MEDIA 93- World Conference on Educational Multimedia and Hypermedia (pp. 619). Charlottesville, Virginia: Association for the Advancement of Computing in Education.

15. Plank, R. J., Burgess, I.W. and Brown, C.J. (1990). Teaching Software For Structural Engineering. G. Goos, J. Hartmanis, D.H. Norrie and H.-W. Six (Eds.), Lecture Notes in Computer Science: Computer Assisted Learning - 3rd International Conference, ICCAL '90 (pp. 300-308). Berlin, Germany: Springer-Verlag.

16. Aminrnansour,A.(1993). Multimedia: A Revolutionary Tool to Enhance Teaching and Learning of Structural Steel Design. In H. Maurer (Ed.), Educational Multimedia and Hypermedia, 1993: Proceedings of ED-MEDIA 93 - World Conference on Educational Multimedia and Hypermedia (pp. 11-16). Charlottesville, Virginia: Association for the Advancement of Computing in Education.

17. Welch, R. (1995). Computer-Assisted Learning in Control Systems Education. M.Eng Thesis, Department of Mechanical & Aerospace Engineering Department, Carleton University.

18. Franklin, G.F, Powell J.D. and Emami-Naeini, A. (1994). Feedback Control of Dynamic Systems, 3rd Ed. Reading, Massachusetts: Addison-Wesley Publishing Company.

19. Goheen, K.R. (1994). 86.352 Course Notes. Carleton University.

20. Goheen, K.R. and Brown, D. (1995). Financing Impact of Increasing itv Operations. Report for the Vice-President (Academic), Carleton University.

Figure 1. Cost Variation with Number of Students

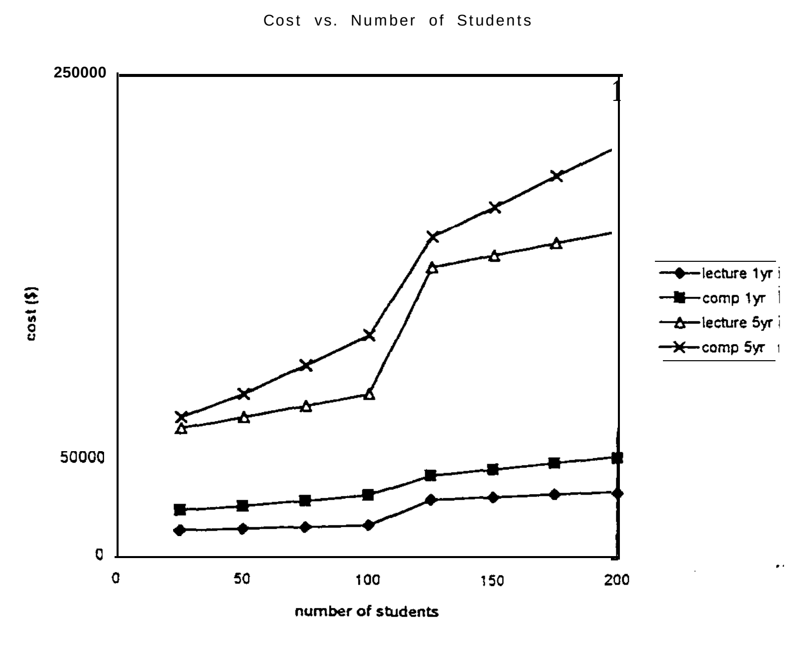

Figure 2. Cost Variation with Development Cost

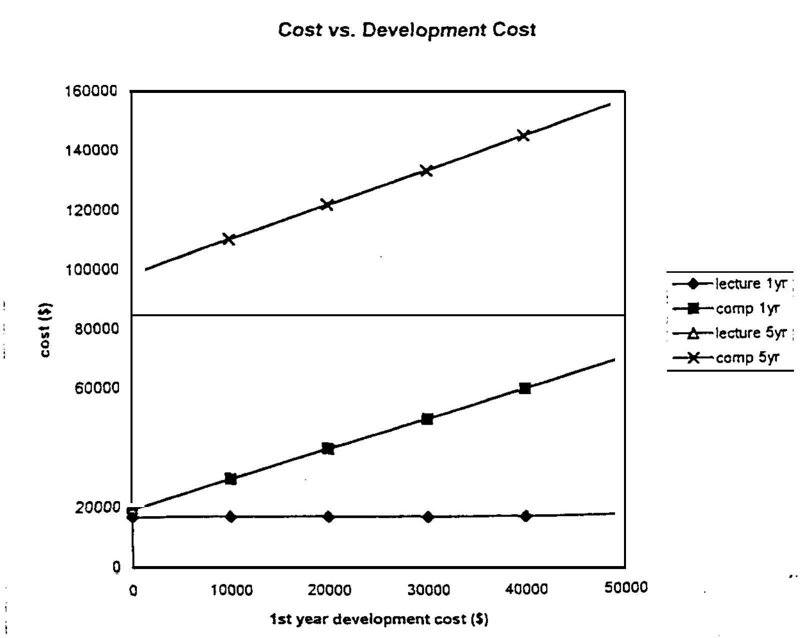

Figure 3. Cost Variation with Computer Use Rate

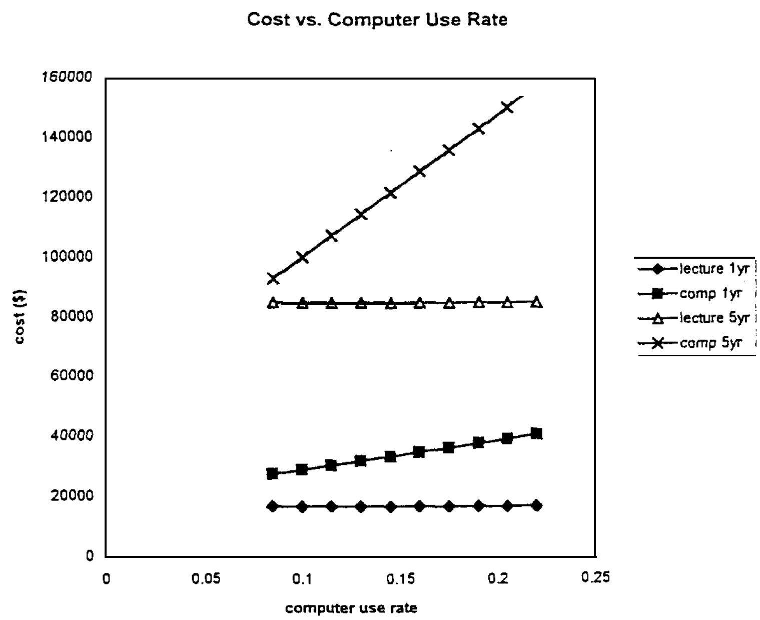

Table 1. Significant t-Test Results

X 1st column	Y 2nd column	t t-test result	v degrees of freedom	p probablility of null hypothesis
degree program	ease of course	2.895	29	0.0071
degree program	exam %	-2.221	28	0.0346
degree program	grade #	-2.401	28	0.0232
group (comp. or lecture)	ease of course	-2.559	30	0.0158

Table 2. Significant Correlations

X 1st column	Y 2nd column	r correlation coefficient	c record count	v degrees of freedom
years using comp.	knowledge of comp.	0.538	32	30
years using comp.	enjoyment of comp.	0.402	32	30
knowledge of comp.	enjoyment of comp.	0.721	32	30
ease of course	additional hours	0.463	27	25
ease of course	midterm %	-0.401	30	28
c-amount of FR	l/c preference	0.584	16	14
c-amount of FR	exam Q2	0.614	13	11
test %	exam Q3	0.432	24	22
midterm %	grade #	0.494	29	27
exam %	grade #	0.952	31	29
total hours	ease of course	0.365	32	30
total hours	test Q1	0.420	32	30
c only -total hours	test %	0.508	16	14

Table 3. Notable Non-Significant Results

1st column	2nd column
group (comp. or lecture)	hours spent
group (comp. or lecture)	additional hours
group (comp. or lecture)	total hours
group (comp. or lecture)	test Q1
group (comp. or lecture)	test Q2
group (comp. or lecture)	test %
group (comp. or lecture)	exam Q2
group (comp. or lecture)	exam Q3
group (comp. or lecture)	exam %
group (comp. or lecture)	grade #
group (comp. or lecture)	l/c preference
enjoyment of comp.	test %
enjoyment of comp.	exam %
enjoyment of comp.	grade #
enjoyment of comp.	c-liking of software
enjoyment of comp.	c-amount of FR
enjoyment of comp.	l/c preference
knowledge of comp.	test %
knowledge of comp.	exam %
knowledge of comp.	grade #
knowledge of comp.	c-liking of software
knowledge of comp.	c-amount of FR
knowledge of comp.	l/c preference
years using comp.	c-liking of software
years using comp.	c-amount of FR
c only -enjoy. of comp.	test %
c only -enjoy. of comp.	exam %
c only -enjoy. of comp.	grade #
c only -enjoy. of comp.	liking of software
c only -enjoy. of comp.	l/c preference
c only -know. of comp.	test %
c only -know. of comp.	exam %
c only -know. of comp.	grade #
c only -know. of comp.	liking of software
c only -know. of comp.	l/c preference
c only -yrs using comp.	test %
c only -yrs using comp.	exam %
c only -yrs using comp.	grade #

Table 3. Notable Non-Significant Results (cont'd)

X 1st column	Y 2nd column
c only -yrs using comp.	liking of software
c only -yrs using comp.	amount of FR
c only -yrs using comp.	l/c preference
c-liking of software	test %
c-liking of software	exam %
c-liking of software	grade #
c-liking of software	c-amount of FR
c-liking of software	l/c preference
c-amount of FR	test %
c-amount of FR	exam %
c-amount of FR	grade #
% from lectures	l/c preference
c only -% lectures	liking of software
c only -% lectures	l/c preference
c only -% lectures	test %
c only -% lectures	exam %
c only -% lectures	grade #
ease of course	test %
ease of course	exam %
ease of course	grade #
total hours	test %
total hours	exam %
total hours	grade #
total hours	l/c preference
hours spent	c-liking of software
hours spent	c-amount of FR