

AC 2008-1408: FRESHMAN ENGINEERING: AN INTRODUCTORY COMPUTER COURSE TEACHING MATLAB AND LABVIEW

John Hrynuik, Clarkson University

John T. Hrynuik is an Undergraduate Teaching Assistant in the Mechanical and Aeronautical Engineering Department.

Matthew Pennington, Clarkson University

Matthew T. Pennington is an Undergraduate Teaching Assistant in the Mechanical and Aeronautical Engineering Department.

David Illig, Clarkson University

David W. Illig is an Undergraduate Teaching Assistant in the Electrical and Computer Engineering Department.

John P. Dempsey, Clarkson University

John P. Dempsey is a Professor in the Civil and Environmental Engineering Department.

Freshman Engineering: An Introductory Computer Course Teaching MATLAB and LabVIEW

Abstract

This paper will describe the adaptation and implementation of a revised curriculum for Clarkson University's freshman engineering course ES100: Introduction to Engineering Use of the Computer. The objective in ES100 is to teach engineering problem solving using both MATLAB and LabVIEW. The course was previously taught by Clarkson University's four engineering departments without the implementation of a common curriculum. Through the award of a Course, Curriculum and Laboratory Improvement, Adaptation and Implementation (CCLI A&I) grant by the National Science Foundation, ES100 was consolidated into an interdepartmental course in spring 2005, in which all freshmen engineering students were presented with common course content. The teaching environment was restructured to place students in sections of around 30 students, each with their own computer workstation. A series of interactive, PowerPoint-oriented, hands-on lectures were created as part of the course restructuring. Hands-on laboratory experiments designed to reinforce lecture material and introduce experimental methods were also created. Post-course evaluation surveys were administered in 2005 and 2006. Survey results lead to the creation of a panel of students in summer 2006 that had just recently completed ES100. The advice of this student panel led course coordinators to continue revisions, including suggested lecture and lab revisions, a modified grading system, and the introduction of Undergraduate Teaching Assistants (UTAs). The spring 2007 course included these UTAs in the classroom. An additional evaluation survey was administered at the end of the course in spring 2007. This paper will report the findings of these surveys, the effectiveness of course revisions, and discuss the general impact of departmental course integration.

Introduction

The revisions to Clarkson's ES100 course were made possible by a CCLI A&I grant¹ from NSF. This proposal's intent was to design a hands-on laboratory setup for all engineering undergraduates at Clarkson University.¹ The Principal Investigators of this "Hands-On Learning in Engineering" project were Professors J. Dempsey, J. Carroll, J. Taylor, W. Wilcox, and A. Zander. The teaching methodology for the revised ES100 course adapted the 'integrated teaching and learning' paradigm pioneered and developed by Drs L.E. Carlson and J.F. Sullivan at the University of Colorado at Boulder.² The adaptation at Clarkson is a combination of laboratory experience woven within an introductory computer course teaching both MATLAB and LabVIEW. The goals and objectives of this original proposal are listed below. These goals have guided the ES100 course revisions throughout all of the twists and turns discussed in this paper. Significantly, note that just recently (February, 2008), Drs Sullivan and Carlson were awarded the prestigious 2008 Bernard M. Gordon Prize by the National Academy of Engineering for their ITL Program at Boulder.³

Goals and Objectives

The list of goals and objectives presented in the original project proposal are listed below:

- a) Create a hands-on learning environment that enhances the laboratory experience across the engineering curriculum.
- b) Promote collaborative learning and team-oriented problem-solving skills.
- c) Build self-confidence.
- d) Promote the ability to design and conduct experiments.
- e) Augment theoretical engineering courses with related hands-on interactive experimental modules.
- f) Integrate research methodologies and advances into the undergraduate curriculum.
- g) Broaden the range of teaching styles to meet the needs of a diverse student population who learn in different ways, thereby increasing the retention of all students.
- h) Increase the recruitment and participation of under-represented groups.
- i) Coordinate throughout the engineering curriculum, via repeated exposure, diverse programming opportunities in both LabVIEW and MATLAB.
- j) Produce a more knowledgeable and experienced engineer better suited to handling real world problems.

Course Origin and Format

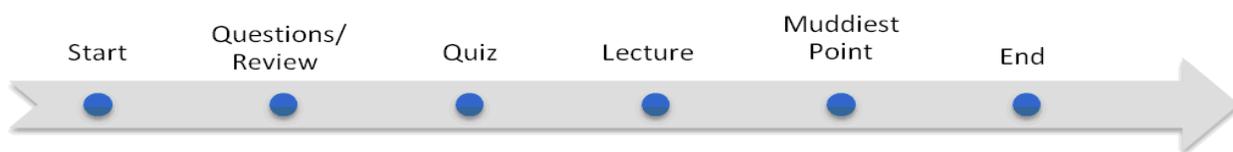
Prior to the spring of 2005, computer-based courses at Clarkson University were taught by the four engineering departments using different syllabi, software, and grading methods. The four engineering departments at Clarkson are Civil and Environmental Engineering (CEE), Chemical and Biomolecular Engineering (CBE), Electrical and Computer Engineering (ECE), and Mechanical and Aeronautical Engineering (MAE). The separate courses taught by these departments were almost completely independent in teaching style, textbook, programs taught, and examples used. Each course revolved around concepts applied in later courses taught by that professor and/or his respective department. For example, a CEE professor might use an example based on beam deflections, while a professor from the MAE department might use an example based around aerodynamics. Faculty also taught using different programming languages. MAPLE, MATHCAD, EXCEL, MATLAB, and other program packages were used at different times. In 2002, a CCLI A&I proposal was submitted to NSF proposing that a MATLAB and LabVIEW course be taught in a hands-on style to all engineering disciplines.¹ The fact that ES100 would be the most suitable course was not anticipated at that time. The proposal was funded in 2003, and, after many interdepartmental discussions, a method of combining ES100 into a jointly run, hands-on course was adopted by all four departments. MATLAB and LabVIEW were selected in the proposal to be the only programming languages taught in the combined ES100. MATLAB is a modern text-based programming language and problem solving environment with sophisticated data structures, built-in debugging and profiling tools, and support for object-oriented programming. MATLAB syntax also closely parallels FORTRAN and C based languages, allowing students to more readily learn these languages in the future. MATLAB was also in reasonably common use at Clarkson University in higher level courses before the introduction of ES100. At the time, a new program 'Computational Science and Engineering' was also being set up by the Clarkson School of Science with MATLAB as its core programming language. LabVIEW was chosen to be used in the laboratory experiments

associated with the course and was primarily used as a data acquisition tool. LabVIEW is an industry-standard graphical development environment for automated testing and measurement. The LabVIEW working environment is similar to the construction of flow charts. A set of standard National Instruments (NI) components are available for use in the experiments proposed and are readily available from NI.⁴ At the time of submission, this proposal had strong commitment from the Dean of Engineering and University Provost. A unified commitment from the chairs of all four engineering departments was the final step to bringing ES100 into its combined status. Two teams of faculty and graduate students were formed: one to work on the overall curriculum issues and one to develop multi-disciplinary laboratory experiments.

The combined sections of ES100 began in the spring of 2005. This course became a graduation requirement for all engineering students. The graduation requirement created a class size of over 300 mostly freshman level students. To accommodate the hands-on nature of the course students were separated into class sizes between 20 and 40 students, requiring around 12 class sections. These relatively small groups of students could attend class in computer labs designed as lecture halls. Each student had access to a computer at all times during class and was able to follow along with the lectures presented. They were also able to work out examples in MATLAB during the class with the help of the professor. A set of 28 PowerPoint lectures were created for the course. These lectures were used in all 12 sections and by all five instructors. The central PowerPoint lectures were seen by all students, but the instructors were given some time to discuss examples and issues of their own interest. An overall lecture timeline was created to work around these PowerPoint presentations.

Lecture Structure

Figure 1: Timeline of Typical Classroom Session - 2006



As the timeline in Figure 1 shows, time was allotted at the beginning of each lecture session for students to ask questions on confusing topics. A daily quiz was also given and was expected to run for around 15 minutes. These quizzes were primarily multiple choice and fill in the blank style questions. The PowerPoint lecture would be presented following the quiz and would take the majority of the lecture time. A “Muddiest Point”⁵ was given at the end of the lecture for students to comment on the subjects they had the most trouble with, as well as comment on instructor teaching methods and course materials. Students were required to submit a Muddiest Point as it would also count for attending the lecture and would account for a small percent of the students’ grade. This small grade percentage was created as incentive for students to provide feedback on each and every lecture.

The lecture topics covered for this running of ES100 is shown in Table 1 below. The lecture topics for each subsequent revision will also be shown and compared to the Spring 2006 case.

Table 1: Lecture Topics Spring - 2006

Lecture Session	Lecture Topics
1	Introduction to the Course and MATLAB
2	Variables and Arrays
3	Array Operations, Displaying Data
4	Built-in Functions, Plotting, Debugging
5	Program Design, Flowcharts and Pseudo-Code
6	The Logical Data Type
7	Branches: if and switch
8	Plotting
9	Lab I Preview/ Exam 1
10	Looping: for and while
11	Logical Arrays
12	Functions
13	Lab 1 Review/ Lab 2 Preview/ Exam 2
14	Variable Passing and Optional Arguments
15	Global and Local Variables
16	Complex Data
17	Lab 2 Review/ Lab 3 Preview
18	String Functions Part I
19	String Functions Part II
20	Additional 2D Plotting
21	Lab 3 Review/ Lab 4 Preview/Exam 3
22	3D Plotting
23	Cell Arrays Part I
24	Cell Arrays Part II
25	Solving Linear Simultaneous Equations
26	Symbolic Math
27	Review
28	Exam 4

To facilitate the large quantity of lecture materials, quizzes, and Muddiest Points for 300 plus students, the BlackBoard Academic Suite⁶ of online resources was utilized for the course. This was also a readily available application on Clarkson's campus and was already in place for other courses. All PowerPoint lectures were posted on BlackBoard for students to access at any time. Quizzes and Muddiest Points were programmed to be accessible to students only during class

times, and passwords were implemented so students not attending class would be unable to take the quizzes. Homework assignments were a combination of textbook examples from Stephen J. Chapman's *MATLAB Programming for Engineers* (1st Edition).⁷ Homework submission was also handled by the BlackBoard system through the "digital drop box" feature, which allowed students to submit homework in digital form, such as the M-file format. These submissions could be accessed by the instructor and recorded the time of submission from the students.

A series of four laboratory experiments were also designed to supplement the in-class activities with a hands-on learning experience in a lab setting.^{8,9,10} The lab setups were designed around a proton exchange membrane hydrogen fuel cell. This device was supplemented with solar panel cells, water electrolyzers, light sensors, and various light sources. Each of the four labs was designed around a concept the students were tasked to analyze, such as the efficiency of the fuel cell-electrolysis system². Data was recorded using a LabVIEW Virtual Instrument (VI) provided to the students for the lab⁸. A formal lab report was collected from each lab group.

The first iteration of ES100 proved relatively successful, but the course still had many areas in need of improvement. Surveys were taken at the end of the course. Few changes were made to laboratory experiments. The course was run in the spring of 2006 under the same conditions with a new freshman class. During spring 2005 and spring 2006, considerable effort was expended on the design of suitable hands-on PowerPoint lectures for BlackBoard and on a novel fuel cell experiment.^{11,12} It was during this semester that the need to thoroughly re-evaluate the course was realized. A second round of evaluations was given at the end of spring 2006. This led Professor John P. Dempsey, the course coordinator, to invite students that had recently completed ES100 to return to campus for a late summer workshop to evaluate the course.

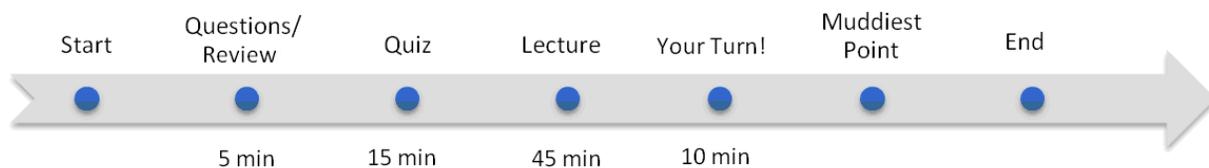
Course Revisions - 2006

At the summer workshop, six students that had taken the course in the spring returned to Clarkson one week in advance of classes to discuss ES100.^{8,13} These students' concerns initially focused on the laboratory experiments. They soon also focused on the lectures, homework, and quizzes. The grading of homework had previously been done on the basis of giving students full credit for submitting a file. The grading of labs was also seen as being deficient. Following the first three days of the workshop, the group of students and instructors decided that the entire course would need a general revision. A new textbook, Holly Moore's *MATLAB for Engineers* (1st edition),¹⁴ would be used as a tool to organize the class timeline. To facilitate this, a team of undergraduates and instructors were tasked with reworking all aspects of the course during the fall semester. These students also took up the newly created position of Undergraduate Teaching Assistant (UTA). They attended all lectures to assist students during hands-on portions of lecture. They also graded each homework assignment for completion and were in charge of handling quiz and exam grading. The UTAs also created AOL Instant Messenger (AIM) screen names, to communicate with students in a means familiar to them, and promoted the use of a discussion board on Blackboard.

During the fall of 2006, seven UTAs worked to revise the course materials. PowerPoint lectures were revised to incorporate directly indicated hands-on examples. To encourage students to participate in these examples, as well as to organize the lectures more effectively, a color code

was applied to the lecture set. The major purpose of this color code was to show students when they should be typing an example of MATLAB code. This was indicated with a yellow font, whose style was specifically chosen to match the font used by MATLAB. New concepts appeared in blue text, while syntax appeared in green. Orange warning messages appeared where the UTAs felt common errors might occur. The UTAs also created a new set of quizzes and homework based around the lecture PowerPoint's and *MATLAB for Engineers* by Holly Moore, the required textbook for the class. Page numbers referencing this text book were added to select PowerPoint slides to reference material being lectured on. A new hands-on section of the lecture was also created, called the "Your Turn!" exercise, and was placed at the end of each lecture. With a new lecture timeline, 15 minutes was allotted for the Your Turn! This was a time where students would work out a few problems based on the material presented in that day's lecture. When a student had completed these problems, the UTA or instructor would give the student the Muddiest Point password for the day and the student could then leave. This allowed the UTAs and instructors to gauge student understanding of material in lecture and help students one-on-one in class. A new lecture timeline, similar to the one shown in Figure 1, was designed around times and a review slide added to each lecture. This new timeline is shown below in Figure 2.

Figure 2: Timeline of Typical Classroom Session - 2007



The timeline for spring of 2007 has been restructured showing the inclusion of the Your Turn! activity at the end of each lecture. A rough time estimate was placed on sections of the lecture. No time estimate is made for the Muddiest Point as the time taken here is included in the time estimated for the Your Turn! Lecture revisions during fall of 2006 were worked around these time estimates. A reevaluation of concepts taught in the course was also part of the fall of 2006 revisions. The new set of lecture topics is shown in Table 2.

The spring 2007 lecture topics shown in Table 2 closely reflect the order of the Holly Moore textbook used for ES100. One notable change in the material covered is the addition of LabVIEW to spring of 2007 as a programming language rather than just a laboratory tool. Lecture time has been reduced in spring of 2007 due to the laboratory experiments. In spring of 2007, when students were required to perform a laboratory experiment, no lecture was held. This adjustment was made based off student comments on the course work load during weeks in which laboratory experiments were held.

The laboratory experiments were also reworked during the fall of 2006. The original set of fuel cell-based experiments drew heavy criticism from freshmen students due to the advanced nature of some concepts presented in these experiments. The four fuel cell-based labs were consolidated into one lab experiment covering only select concepts from those original labs. Two new

experiments were also created. A global positioning system (GPS) lab was created, where students would take data on the Clarkson campus and analyze this data using MATLAB.⁹ A

Table 2: Lecture Topics Spring - 2007

Lecture Session	Lecture Topic
1	Introduction
2	Variables and Arrays
3	Matrices
4	Built-in Functions
5	Importing/Exporting Data, M-files
6	Plotting in 2-D
7	More Plotting in 2-D, Curve Fitting
8	Lab 1
9	Symbolic Mathematics
10	User-Defined Functions
11	Lab 2
12	Exam 1
13	User-Controlled Input and Output
14	Logical Functions
15	Selection Structures
16	For Loops
17	While Loops
18	Looping Practice
19	Function Practice
20	Exam 2
21	Lab 3 lecture
22	Lab 3 Part A
23	Lab 3 Part B
24	LabVIEW Introduction
25	LabVIEW Programming
26	LabVIEW Simulation
27	Exam 3
28	Course Surveys

traffic light setup at a small model intersection was designed for the third lab experiment, where students were required to write a program to control the traffic lights at a theoretical intersection.¹⁵ A small Measurement Computing Corporation data acquisition device was used for that lab.

With course materials approaching completion, a presentation was made to the course instructors in October 2006. Early mixed reviews of the process eventually led to a consensus to run the course with the new material and the new UTAs. The instructors were presented at this time with the first ten course lectures to review and were asked to suggest changes and edits. The full set of lectures was provided to instructors over the winter break before the spring semester. On January 6th, 2007 the group of UTAs, now totaling 8 undergraduates, met to set up the ES100 course on the BlackBoard system. A lack of knowledge about this system led to a 3 day session of course setup. The UTAs uploaded and programmed the quizzes, homework's, Muddiest Points, and general course postings such as the syllabus. Full course setup took the group of 8 UTAs approximately 12 hours to complete.

ES100 Spring 2007

With the course fully setup on the BlackBoard system ES100 classes began. A wide range of problems appeared through the course of the semester. While the course started with a full set of quizzes and homework assignments, the exams had yet to be created and the lab experiments had yet to be finalized. The group dynamic between instructors and UTAs also proved a challenging hurdle. Some instructors felt infringed upon by having an undergraduate in their classroom for assistance. Other issues found were occasional typing and factual mistakes in the lecture PowerPoint's, which drew harsh criticism from instructors. Many of these errors had to be changed in between classes. A set of guidelines was also posted for students regarding emailing UTAs and instructors. These guidelines let students know how best to inform instructors and UTAs what their issues were in a professional manner. Numerous announcements were made using the BlackBoard system with a wide range of purposes. ES100 UTA AIM screen names were announced to the students early in the course, as was a homework submission clarifying procedure. Quiz and Summary Quiz (Exam) re-grading occurred throughout the semester as occasional ambiguous wording or errors were found in examination questions. Re-grading of these errors was another task taken on by the UTAs during the semester. A "Uses of" slide was added to certain PowerPoint Lectures through the semester. This slide describes the later uses of the material discussed in lecture. The UTAs handled all homework and required quiz grading through the semester. Lab reports for laboratory experiments one and two were graded by a graduate teaching assistant, while the third lab report was graded by a subgroup of the UTAs.

Surveys were given to all students at the end of the semester.¹⁵ An entire lecture session was devoted to a 70 question survey. A total of 234 students took the survey. The survey questions focused primarily on student opinions of course material. Some specific questions were analyzed for all 234 participating students. Approximately 36% of students taking ES100 had previous programming experience. The number of students that replied "often" or better to a question about how often they typed in yellow code examples during class was 85%. When asked if the third lab experiment improved their understanding of MATLAB, 77% of students responded yes. A slightly lower number of students (65%) felt that homework helped prepare them for quizzes and summary quizzes. Data from these surveys helped reveal the strong and weak points of the course revisions. Yellow code examples given in lectures seemed to be accepted by the majority of the students taking ES100. Far fewer students felt the homework problems connected well with the course, as they were not considered as helpful. These data show that the third laboratory

experiment, based on a traffic light setup, was found beneficial to students as over three quarters of students felt it helped them improve their knowledge of MATLAB.

Analysis of Muddiest Point Data

Muddiest Points throughout the semester were categorized for analysis and possible course revisions. Student responses on a topic over 1 percent of the total set of Muddiest Points are described below and analyzed. Looping was found to be a troublesome area of the course with more than 3% of all Muddiest Points having negative feedback towards the subject. Plotting in MATLAB and user-defined functions also drew notable amounts of student criticism with over 2% of responses focused on each of those subjects. The difficulty level of the course was commented on negatively by 8% of student responses to Muddiest Points. Analysis of future and past courses is needed to fully understand what parts of the course resulted in students finding the class difficult. Over 57% of all Muddiest Point comments were found to be positive, such as the students finding the lecture material clear or commenting that they had understood the material presented in that lecture.

At the conclusion of the spring 2007 lectures, it was found that 3 examples came from freshman level chemistry, 9 examples from freshman level physics, 3 examples from freshman calculus, 43 general math based examples, and 95 examples testing MATLAB and programming principles. Questions about solutions to problems presented in the PowerPoint lectures were also posed and it was found that only about 56% of the examples in the lectures had solutions shown. Analysis of the use of the discussion board was also done on the course. It was found that the discussion board had been viewed 52,423 times by students, instructors and UTAs. Homework assignments were posted on the discussion board and were viewed 16,702 times bringing the total views of ES100 and MATLAB related topics in the discussion board to 35,721. With 338 users enrolled in BlackBoard, including 6 instructors and 8 UTAs, the average number of views of the discussion board per individual was 110 during the course of the semester, or about once per day over a 15 week semester.

Summer 2007 Revisions

Summer of 2007 brought a full course revision again, this time with only 3 of the UTAs from the original 8. All 20 PowerPoint lectures were revised to account for student criticism. Three dimensional plotting and interpolation features have been added into a newly created lecture. These new lecture subjects are detailed in Table 3. Quiz questions were scrutinized for possible errors found during the spring 2007 semester and also had a complete revision. Homework problems were reselected to be taken almost entirely from the Holly Moore textbook to improve the connection between homework and course materials. Some homework problems considered too difficult according to Muddiest Point data were changed and reincorporated in a more appropriate location. For example, a looping homework that required students to write a program developing the Fibonacci sequence was moved from the introduction to looping lecture, to be connected to the new Additional Looping Concepts lecture. The majority of solutions to all course materials, including Your Turn! activities were also created in this revision session. Page numbers and Uses of slides were added to all PowerPoint slides. *Essential MATLAB for Engineers and Scientists* (3rd edition)¹⁶ by Brian Hahn and Daniel Valentine has been added as a

required book to the course. This book will serve primarily as a second reference for students and page numbers for this book have also been incorporated into the PowerPoint lecture slides.

Table 3: Lecture Topics Spring - 2008

Lecture Session	Lecture Topic
1	Introduction
2	Variables and Arrays
3	Matrices
4	Built-in Functions
5	Error Checking and Transferring data
6	Plotting in 2-D
7	More Plotting in 2-D, Curve Fitting
8	Lab 1
9	Symbolic Mathematics
10	Exam 1
11	User Defined Functions
12	Input and Output
13	Logical Operators and Functions
14	Planning Tools and Selection Structures
15	Lab 2
16	For Loops
17	While Loops
18	Exam 2
19	Additional Looping Concepts
20	Lab 3 Lecture
21	Lab 3 Week 1
22	3-D Plotting and Interpolation
23	Lab 3 Week 2
24	LabVIEW Introduction
25	LabVIEW Programming
26	LabVIEW Simulation
27	Exam 3
28	Evaluations

Laboratory experiments were also reevaluated at this time. Based on student comments, an aging laboratory setup, and UTA and instructor experiences, the fuel cell experiment was discontinued. The purpose of this lab was for the students to be able to gather experimental data for analysis with MATLAB. A primary issue with this specific lab setup was the complexity of the device and the frequent malfunctions during experiments. UTAs and instructors present in lab sessions struggled to keep up with and troubleshoot problems students encountered while performing the experiments. The lab will be replaced in spring 2008 with a similar front end experience for the

students, with a lab based around projectile motion. A LabVIEW VI will be created in a manner similar to the fuel cell VI, but a physical apparatus will not be used for this new projectile motion experiment.² Use of functions built into LabVIEW will allow for the creation of data with simulated experimental errors. Simple random number generation will be used to take standard known equations involved in projectile motion and produce small errors designed to mimic the errors that might be found in experimental data. This data however will not be subject to malfunction of a physical apparatus or true experimental error. While this experiment may not precisely match the intents of experimental design, it does allow for large numbers of students to gather data that can be analyzed without having to repeat the experiment. The choice to use this new lab design is based on UTA feedback about in lab equipment situations and student frustrations with malfunctioning equipment and poor quality data. The GPS lab will be reworked to include higher level uses of MATLAB to analyze data. The lab will be repositioned in the semester timeline, which will allow for the use of looping techniques in the analysis. A more complex problem, in which students were asked to find the area inside a GPS path, was introduced by John E. Bean, a graduate student working on the course, in 2006. The way this problem was originally posed, it proved too confusing for a homework assignment, but will be reworked into the laboratory experiment where students will have more time to work on the problem, as well as group help to solve the more challenging question. The third laboratory experiment will be revised primarily by rewording and general editing. This is based on student responses to that experiment, where some students found the instructions confusing or convoluted. Other students found that the logic was not explained in a clear enough manner. These two problems are the primary reason for revisions to Lab 3.

Conclusions

Ongoing revisions to the ITL concept created at UC Boulder that has been adapted and implemented at Clarkson University in ES100 has developed a unique program and experience for both students and instructors. The high level of student feedback in the interactive, hands-on environment in ES100 allows students to influence their learning experience.

Acknowledgements

The authors wish to acknowledge the dedicated efforts of James Carroll (ECE), Jubum Kim (CEE), Daniel Valentine (MAE), and William Wilcox (CBE). The authors also wish to acknowledge PhD student John Bean for his contributions to the instruction and revision of ES100. In addition, the authors wish to acknowledge the contributions of the Clarkson undergraduate students who participated in the August 2006 workshop and/or served as UTAs: Kyle Ashley (CEE), Sarah French (CEE), Kayla Kuzmich (MAE), Nora O'Loughlin (MAE), Michelle Turk (ECE), and Greg Zenger (ECE). This CCLI A&I project is being supported in part by the National Science Foundation through the Department of Undergraduate Education Grant 0311075 and in part by the Coulter School of Engineering and the President's Office at Clarkson University.

Bibliography

1. Dempsey, J. P., Carroll, J. J., Taylor, J., Wilcox, W. R., and Zander, A. K. "Hands-On Learning in Engineering." NSF CCLI A&I Grant DUE-0311075.
2. Carlson, L.E. and Sullivan, J.F. "Hands-On Engineering: Learning by Doing in the Integrated Teaching and Learning Program," *International Journal of Engineering Education* 15 (1999) 20-31.
3. <http://www.nae.edu/nae/awardscom.nsf/JMAN-7A4L7N?OpenDocument>
4. National Instruments, Austin, Texas: National Instruments Corporation, 2000.
5. Angelo, T.A. and Cross, K.P. *Classroom Assessment Techniques*. 2nd edition. San Francisco: Jossey-Bass 1993 pp. 154-8.
6. *Blackboard Academic Suite*. Blackboard Inc., 2008.
7. Chapman, S. J. *Essentials of MATLAB Programming*. 1st ed. Ontario: Thompson Publishing, 2005.
8. Schwartz, T.L. and Dunkin, B.M. "Facilitating Interdisciplinary Hands-on Learning using LabVIEW," *International Journal of Engineering Education* 16 (2000) 218-227.
9. Elshorbagy, A. and Schonwetter, D.J. "Engineer Morphing: Bridging the Gap Between Classroom Teaching and the Engineering Profession," *International Journal of Engineering Education* 18 (2002) 295-300.
10. Ferguson, E.L. "Learning With Real Machines or Diagrams: Applications of Knowledge to Real-World Problems" *Cognition and Instruction* 13, (1995) 129-160.
11. Bean, J., Carroll, J., Dempsey, J.P., Strong, A.H. and Wilcox, W.R. "Adding a Hands-On Laboratory Experience to the Freshman Engineering Programming Class at Clarkson University." *Proceedings 2006 Annual Conference*, Paper #2251, 9p.
12. Bean, J., Carroll, J., Dempsey, J.P., Valentine, D.T. and Wilcox, W.R. "Collaboration between Engineering Departments at Clarkson University for a Freshman-Level Engineering Programming Course Including an Experimental Lab Experience." *CIEC 2007 Conference*, 4p.
13. Busby, J.S. "An Evaluation of an Instructional System for Engineering Task Estimation," *IEEE Transactions on Education* 43 (2000) 30-35.
14. Moore, H. *MATLAB for Engineers*. 1st ed. Upper Saddle River, New Jersey: Pearson Prentice Hall, 2006.
15. Chang, J.L. , Picket-May, M.J. , Avery, J.P., "Using Active Student Feedback in the Learning Environment", *Frontiers in Education Conference*, 1998. FIE '98. 28th Annual Vol. 2 pp. 643-646.
16. Hahn, B. and Valentine, D. *Essential MATLAB for Engineers and Scientists*. 3rd ed. Italy: B-H Elsevier, 2007.