The Scholarship of Teaching and Learning at EMU

Volume 1 *Toward Transformation: EMU Faculty Journey into the Scholarship of Teaching and Learning* Article 5

9-28-2009

Matlab as a Tool to Increase the Math Self-Confidence and the Math Ability of First-Year Engineering Technology Students

Mary L. Brake
Eastern Michigan University, mbrake@emich.edu

3

Matlab as a Tool to Increase the Math Self-Confidence and the Math Ability of First-Year Engineering Technology Students

Mary L. Brake
School of Engineering Technology

Mary Brake's chapter details her efforts to introduce a computational software package (Matlab) in her Introduction to Engineering Technology course. Mary started off her scholarship of teaching and learning project interested in why women and minorities tend to drop out of engineering programs nationwide; she anticipated it is caused by lack of confidence these students have that they can "make it" in such a program. Along the way, Mary's focus widened as she discovered that the problem may not exactly be lack of confidence; her students continued to feel confident in their ability to solve complex problems even in the face of clear evidence that they could not.

Mary's big goal has become to help students learn to think like "experts." She wants her students to excel at using information they have previously learned in solving complex problems. This investigation speaks to how Matlab can help students learn to do this; ultimately, the goal is to help students feel (deservedly) more self-confident in their ability to do the math necessary for an engineering technology program. From a practical standpoint, this study suggests that Matlab would be most valuable to students if it were introduced early in the program. Mary also suggests it might prove more valuable when not crammed into an already full class such as hers.

Framing the Question

I have now taught three semesters of Introduction to Engineering Technology, as well as upper level courses in mechanical engineering technology. This experience comes on the heels of three years of high school teaching and sixteen years of teaching nuclear engineering. Right away I noticed that freshmen engineering technology (ET) students tend to enter college with less math preparation than students who study engineering, and that many of them really struggle with the simplest of concepts. In contrast, my upper-level students are able to approach each problem asking, "What concepts have I learned that could be used in this situation?" There is a big disconnect between the problem-solving approaches of freshmen and more experienced students.

In my experience, one of the most difficult parts of teaching technical subjects like ET, engineering, or even high school physics, is to move the students to the point where they can take the basic concepts that they have practiced on homework and exams and apply them to solving new problems they have never encountered. This is typically the difference between a novice learner, who typically does not know how to accomplish this transference of knowledge, and an expert learner, who typically does (Wineburg 2001).

At one point in a student's academic career they usually "hit the brick wall" as I call it. They discover that they cannot memorize their way to success in learning difficult material. This realization is painful for students; often, they abandon technical careers because they can't get past "the brick wall." More self-confident students find a way to become expert learners and less confident ones struggle, sometimes making it, sometimes not. One of the biggest challenges educators face is training students to analyze new situations by transferring the basic concepts, even if they have not seen the specific problem before. In other words, it is the challenge of training novice learners to be expert learners.

I decided to investigate if introducing a computational software package like Matlab into ET 100 (Introduction to Engineering Technology), the first engineering technology course in the ET program, would improve student learning. I wondered if Matlab would increase student's self-confidence in solving technical problems, particularly in analyzing problems using mathematics. I also wanted to see if I could improve students' math skills. I was hoping that if students didn't struggle so much with the mathematics, then they would be able to analyze a problem as a real-world problem rather than only a math problem. While beyond the specific scope of this study, my long-term hope is that if I can get students over "the brick wall" by helping them become expert learners, more of them would finish their degrees in technology.

The Context: Introducing Engineering Technology to Freshmen

In the 1960s, engineers were taught using a theoretical approach. There was a need, however, for practical engineers whose training was more applied. Engineering technology was developed to bridge the gap between theoretical engineering and applied engineering. Until recently, engineering technologists often worked under the direction of an engineer to help communicate information to workers (e.g., in a manufacturing setting). They performed calibrations and tests on equipment. ETs also supervised skilled and semi-skilled workers (Pond 2005). But engineering and technology in general has become so interdisciplinary that many students with engineering technology degrees are being hired as "engineers." This means that in addition to the duties mentioned above, they also design products, manage manufacturing systems and work all of the jobs done by engineers in industry. They do not tend to get jobs in research, but that is not the intent of the major.

Due to the changing workplace and the change in expectations of engineering technologists, it is important that ETs learn how to perform the same types of calculations as traditional engineering students. This requires a change in curriculum. About ten years ago, engineering schools started to adopt computational software packages such as Matlab, Maple, or Mathematica into their curriculum (Hodge and Steele 2002; Hornaes and Royrvik 2000). Some dropped the traditionally required computer courses in C++ (or FORTRAN) because of this change. Students were then taught the basics of computational software packages as freshmen and were expected to pick up more complex commands when they encounter difficult problems in their upper level courses.

My anecdotal sense is that most engineering students nationwide take calculus as freshmen whereas only a few ET freshmen do so; ET students usually start with college algebra or college trigonometry. ET students are technically very competent by the time they graduate; however, the first couple of years of an ET curriculum can seem daunting for a student who starts college less mathematically prepared than one who can dive into calculus.

An engineering technology major is difficult compared to many college majors. Students need to be empowered with a feeling of confidence in the areas in which they feel most vulnerable, like math. In comparing my upper level classes to my freshmen classes, it is apparent that women and minorities seem to fall by the wayside. They seem to be particularly at-risk groups for not finishing, both at EMU and nationwide. There are more studies of the retention problems in engineering (e.g., Wolcott 2007) than in ET because of the large number of engineering students and the fact that there are more women enrolled as undergraduates, compared to men. For example, 74,186 students graduated with a bachelor's degree in engineering in 2005-2006 (Gibbons 2006), with 19.3% awarded to women (the lowest representation since 1998). African-American and Hispanic students made up 11%. In engineering technology, however, only 5,221 students graduated with bachelor's degrees in 2005-2006 (Gibbons 2006) and only 10.4% were women. African-American and Hispanic students made up 13%. By introducing a tool to help students improve their math skills and their self confidence, I hoped that more students, particularly women and minorities, would remain in ET. This would require a larger study than the one discussed here, but introducing Matlab into the curriculum and studying how students respond to it is a start.

I studied two sections of ET 100, Introduction to Engineering Technology, during the 2006-2007 academic year. This course has no prerequisites and is intended to introduce the various fields of engineering technology to interested freshmen. It is a required course for our three ET majors: Mechanical Engineering Technology, Electronics Engineering Technology, and Computer Engineering Technology. In this class the students are introduced to engineering technology careers, ethics, two projects, team work, project management and a review of conversion of units, particularly the confusing English units, as well as a review of early high school math that they will need to know to be successful in college math and science.

Last year the industrial advisory board of Eastern Michigan University's electronics engineering technology program suggested that we introduce computational software like Matlab into our curriculum. In recent conversations with the mechanical engineering technology industrial board, the members were not concerned with the ability of students to solve problems using structured program languages such as C++ but they said that they expect potential employees to solve real world problems that involve mainly arithmetic and algebra. Matlab therefore is an important job skill, not just a tool to be able to solve engineering problems in college.

When I taught Introduction to Engineering Technology for the first time I noticed that the students who struggled with simple algebra and simple unit conversions appeared to have the least amount of self confidence; they appeared to be less certain of remaining in ET compared to better-prepared students. They did not do as well as the students who had stronger math skills *even though many of our projects did not require mathematics*. There is some evidence to suggest that students' confidence in their ability to solve complex math problems can not only determine their success but also how likely they are to finish their challenging major (Gallaher and Pearson 2000; Yokomoto et al. 1999). To address the problem of retention, self confidence, and skills expected by industry, the School of Engineering Technology invested in 50 licenses for Matlab, which were installed on the server to two key computer labs used by ET students. Since all ET majors are required to take this introductory class, I believed, based on best practices at other schools, that it was most expeditious to teach the fundamentals of Matlab in Introduction to Engineering Technology.

Gathering the Evidence

I am used to making measurements in a laboratory where I know the controls and variables. I could have taught one section with my usual curriculum (the control group) and then taught the second section using Matlab. But each section was limited to twenty students and I felt that I would not gather as much information compared to studying both groups together. Also, since I believed that Matlab would make a difference, it seemed only fair to use it to teach all the students. I had a sense of how students did in the course from teaching it the two previous semesters. I felt that these prior classes could serve as a qualitative control group and this semester's sections could be the experimental group. The total number of students in both of my sections (38) wasn't large enough to gather statistically significant data. However, it is one of the first attempts to understand freshmen ET students and how well computational software packages affect their self esteem as well as how well they perform in class.

I surveyed the two sections to obtain demographic information, as well as data on their past mathematical training and how confident they felt in attempting to solve two rather difficult problems. Many of the demographic questions were based on the work of Gallaher and Pearson (2000) who studied women's perception of the climate in engineering technology. I noticed that their results (to be discussed later) appeared to describe all my students, regardless of gender. The survey also contained twelve loci of control questions that indicated their level of overall self-confidence by asking questions regarding how much control they perceived to have over their life (Rotter 1966). I also kept track of quiz, midterm and final grades on specific questions solved using Matlab. At the end of the semester I surveyed them again with the same questions as on the first survey but without the demographic questions (since I already had that data).

In addition to the quantitative data, I kept track of how much material I had covered by certain points in the semester. The curriculum is quite full and I had to either eliminate or shorten the time spent on particular topics, specifically graphing and demonstrating how to make a good oral presentation. I was not able to study the retention of students because this would require a more in depth study that was beyond the scope of this year-long study (Al-Holou et al. 1999; Knight, Carlson and Sullivan 2003). For example, Gallaher and Pearson (2000) found that women often leave ET due to lack of peer support, lost confidence and the competitive culture, all important topics but difficult to study over one course.

I used two questions that can be found in the first chapter of "Introduction to Matlab 7" by Palm (2005) as my benchmark. This book was one of three that I recommended to the students to consider buying as a reference. The questions were as follows:

> Question 1: $e^{(-4.1)^3} + 3.57\log(36) + \sqrt[4]{142}$
>
> Question 2: The amount of energy released by an earthquake can be described by the following: E = $10^{4.4}10^{1.5M}$ where *M* is the magnitude of the earthquake.

I chose these questions because I had found that students really struggled with taking roots of numbers, including the square root of a number (much less the fourth root) and that they had problems with exponents in general. I chose the second question because it not only involved exponents but also required students to apply the math to a real world problem. For example, students were asked to compare the amount of energy released from an earthquake of one Richter number (for example, a "7") to that of another Richter number (for example, a "5"). Even when I specifically told them to find the energy for two different Richter numbers separately and then divide the results to determine how much more energy was released in the large earthquake versus the smaller earthquake, many could not grasp this concept of comparison.

Emerging Findings and Broader Significance

It was important to learn the demographics of my students because they tended to be very different from students I had previously taught at a large engineering school. As mentioned, I took many of my questions from Gallaher and Pearson (2000). They found that women ET majors were different compared to women engineering majors. They were more likely to be middle children, rank in the upper 10 – 25% rather than 2 -5% of their high school graduating class and come from less affluent circumstances. I found that the majority of most of my students fit this profile regardless of gender.

Out of the 35 students who filled out the consent form and then took the initial survey, 28 were male and 7 were female. Even though the two classes contained a fairly large number of minorities (mainly African American), I did not ask ethnic background because I felt the numbers would be too small to make any significant conclusions regarding gender or ethnicity. This is certainly an important topic because more women and minorities drop out of ET compared to white males. This issue would require a longitudinal study; moreover, the reasons for retention are likely to be far more complicated than self confidence in solving math-based problems. This remains an important issue for study by the engineering technology education profession.

All of the students save one were full time students (taking more than 12 credits). Seventy five percent of the students were in the traditional college age group (18 to 22 years old) despite the fact that our university attracts a lot of non-traditional students. Over half of the students said that going to college was their decision, which implied intrinsic self motivation. But over one-third reported that either their mother or father was most influential in making their decision to attend college. This may indicate that for them, a technical major was not the student's first choice and their motivation to attend college was influenced by their parents rather than by a desire to learn.

A key question in this study was how much mathematics the student had in the past or was currently taking. Twenty percent of the students had *not taken any college math* even though this was their second semester. To successfully complete an ET degree, students really need to start college with the skills to take college algebra or, better yet, calculus. I learned from the Math Department at EMU that 22 out of 38 students *did not* meet their criteria to start college algebra. So only 16 students (42%) had a strong enough background to start college algebra or, in a few cases, calculus. An advisor for the College of Technology told me that students who start with remedial math (that is, courses below college algebra) rarely, if ever, end up graduating with an engineering technology degree. Despite this lack of math background, these students, who comprised a majority of the study, seemed to think that they would do well in engineering technology and one of its three majors.

Interestingly, all of the students scored very high on the loci of control questions that can indicate self-confidence and control over one's destiny. On a scale of 1 to 5 with 5 being the most self confident, the mean score at the beginning of the semester was 4.15. For example, students were asked to indicate if they felt that they were in control of/in charge of their career goals and choices. They picked from strongly agree, somewhat agree, neutral, somewhat disagree, and strongly disagree. Another statement they had to agree or disagree with was "Becoming a success is a matter of hard work; luck has little or nothing to do with it." This mean of 4.15 can be compared to another research project of mine where high school students are found to typically average between 3 and 3.5. The fact that my students were attending college and making decisions for themselves seemed to give them a sense of control over their lives; all of the freshmen felt in control.

There was no significant difference in their self confidence at the end of the semester despite struggling with the mathematics portion of the class and the fact than many students were not doing well as measured by their grade in the class. This is a puzzling result. Hornaes and Royrvik (2000) found that "students seemed to be more optimistic about their grades than results from the exams gave them reason to be." Their results may be due to a bias in who took the survey. They did find, however, that there was a correlation between students who were strong in math and those who believed that computational software was beneficial. In fact they found a small but significant result that the majority of students felt that computational software was useful.

To be able to take the first serious engineering technology course requires calculus-based physics, which has a prerequisite of Calculus I. Presumably students who start out with remedial math eventually realize that engineering technology is not a good choice of a college major for them because when I informally surveyed upperclassmen, only one had started out with remedial math; the rest started with college algebra or calculus. The one student who started out relearning arithmetic and basic math had been out of school for several years. Freshmen appear to self-select out of ET based upon the number who take ET 100 and those enrolled in the upper level ET courses.

In the survey at the beginning of the semester and at the end of the semester, I asked the students if they would be able to solve questions #1 and #2 shown above. I've compared their answers with the percentage of students who actually did the problems correctly on the final. Table 3-1 shows that the students tend to overestimate their abilities even at the end of the semester, despite having encountered difficulties with some of the simple Matlab calculations during the semester. This is in keeping with the findings of Hornaes and Royrvik (2000).

Table 3-1: Student Confidence in Ability to Solve Matlab Problems

Problem	% Confident at Beginning *n=30*	% Confident at End *n=22*	% Correct on Final Exam *n=29*
$e^{(-4.1)^3}+3.57\log(36)+\sqrt[4]{142}$	83%	92%	45%
$E = 10^{4.4}10^{1.5M}$	76%	100%	17%

In the middle of the semester, I gave two quizzes where I broke up the two problems into several smaller problems. For the first quiz, students used their calculators and for the second quiz, they were required to use Matlab. At this point in the semester approximately half of the class was having difficulties with the math in general, not just Matlab (see Table 3-2).

At the time of the quiz, the commands for raising numbers to a power and how to find a logarithm to base 10 were relatively fresh in their minds. So, I expected that a larger percent of students would be able to solve the quiz problems. About half of the students were able to do pieces of the problem, but as Table 3-1 shows, many students had considerable difficulty putting all the pieces together in solving the entire problem even by the end of the semester.

I noticed in correcting the final exams that students tended to use Matlab as a fancy calculator instead of breaking up problems into pieces, saving the results, and then calculating the final solution. While using Matlab as if it were a calculator may be possible for smaller pieces of problems, it would ultimately prove an unsatisfactory method for solving longer, more complex problems. Those few students who used Matlab to break up the problem into pieces (which would be very hard to do on a calculator) were most successful; on many occasions I pointed out that they should break up the problem but very few followed my advice. Since the survey was anonymous, I was not able to correlate how students responded to the survey with how well they performed. But in speaking with the students informally, students would admit to me that they did not feel comfortable with Matlab; these were the students who did not do well.

Table 3-2: Student Ability to Solve Problems with Calculator and Matlab

	% Correct on Quizzes	
Problem	**Calculator** *n=26*	**Matlab** *n=27*
3.47 log(14)	70%	59%
$\sqrt[4]{287}$	56%	52%
$10^{4.4}10^{1.5M}$ when M = 3	40%	48%

Since Matlab is new to campus, no students in the class had any training in using the program before the training provided in class. Interestingly, students who had completed a course in a computer programming language (not a course in Matlab) picked up not only the Matlab commands more quickly but also the strategy of solving problems in pieces rather than writing out one long calculation. They tended to be far more accurate than the students who had not had a computer science course. This does not necessarily mean that a course in a computer language is necessary to learn Matlab. However, it may mean that students who chose to take computer science already have higher problem solving skills than their peers.

During the semester I discovered that I spent even more time than usual going over basic mathematics. Since I was squeezing Matlab into an already full curriculum I was not able to cover more complex story problems. I spent less time on how to convert units and it showed – students were not as comfortable or as successful at solving problems that required changing units on either their homework or the midterm. Also, I normally spend more time on how to illustrate results using graphs created with Excel spreadsheets. I find that requiring students to graph results makes them think about how to approach situations which involve calculations and how to illustrate results to draw a conclusion.

These particular students appeared to learn only the basic concepts, not the higher level learning that I had hoped would take place. I wanted them to be able to apply even the most basic math to real world problems. But this requires higher level thinking compared to simply solving a math problem. As I mentioned earlier, one of my goals was to turn these inexperienced students into expert learners of engineering technology. I have found that graphing helps many students learn to think more deeply about problems; unfortunately, there wasn't enough time in this class to fine tune the graphing interpretation skills. This is something to consider in future iterations of the course.

Clearly more research is needed to make a conclusion on the use of Matlab as a tool to increase self confidence and math ability in freshmen ET students. Also, a more thorough examination of the *motivation* of students who pick ET majors is needed in order to start to understand why some students stick with ET and others do not. I have found that ET 100 is turning out to be the "weeder" class. Students who do not do well in this class tend to drop their ET major. Often technical programs have a second admission into the major or a difficult sophomore or junior class to weed out the students who are not suited for the major; I would prefer that freshmen discover for themselves what interests them and what motivates them.

Students in technical majors need to learn to go from solving the mathematics of a problem to transferring their knowledge so that they can analyze new situations and *apply* math. Despite the inconclusive results with Matlab, I feel that as educators, we need to teach our students the tools required to make them "expert" learners in their discipline. This is a skill that students will need for future college work as well as life in general. I still think that Matlab is one approach to achieving this goal; my experiences make me wonder if refinement of other teaching strategies are needed as well.

References

Al-Holou, Nizar, Nihat M. Bilgutay, Carlos Corleto, John T. Demel, Richard Felder, Karen Frair, Jeffrey E. Froyd, Marc Hoit, Jim Morgan and David Wells. 1999. "First-Year Integrated Curricula: Design Alternatives and Examples." *Journal of Engineering Education* 88: 435-448.

Gallaher, Janna and Frances Pearson. 2000. "Women's Perceptions of the Climate in Engineering Technology Programs." *Journal of Engineering Education* 89: 309-314

Gibbons, Michael. 2006. "Engineering by the Numbers." www.asee.org/publications/profiles/index.cfm

Hodge, B. K. and W. Glenn Steele. 2002. "A Survey of Computational Paradigms in Undergraduate Mechanical Engineering Education." *Journal of Engineering Education* 91: 415-417.

Hornaes, Hans P. and Ola Royrvik. 2000. "Aptitude, Gender, and Computer Algebra Systems." *Journal of Engineering Education* 89: 323-329.

Knight, Daniel W., Lawrence E. Carlson and Jacquelyn F. Sullivan. 2003. "Staying in Engineering: Impact of a Hands-On, Team-Based, First-Year Projects Course on Student Retention." Proceedings of the 2003 American Society for Engineering Education Annual Conference & Exposition, Session 3553.

Palm, William J. 2005. *Introduction to Matlab 7 for Engineers*. New York: McGraw Hill.

Pond, Robert J. 2005. *Introduction to Engineering Technology*, 6th Edition. Upper Saddle River, NJ: Prentice Hall.

Rotter, Julian. B. 1966. "Generalized Expectancies for Internal Versus External Control of Reinforcement." *Psychological Monographs* 80 (Whole No. 609).

Wineburg, Sam. 2001. *Historical Thinking and Other Unnatural Acts: Charting the Future of Teaching the Past*. Philadelphia: Temple University Press.

Wolcott, Barbara. 2007. "Filling the Void." *Mechanical Engineering* 129:24-25.

Yokomoto, Charles F., Maher E. Rizkalla, Carol L. O'Loughlin, Mohamed A. El-Sharkawy and Nancy P. Lamm. 1999. "Developing a Motivational Freshman Course in Using the Principle of Attached Learning." *Journal of Engineering Education* 88: 99-1.06.