MATLAB & Freshman Engineering

Patrick E. Devens
Virginia Polytechnic Institute & State University

Abstract

Virginia Polytechnic Institute and State University's (VPI&SU) College of Engineering has made the switch. MATLAB[1] is now the problem-solving software used by the freshman engineering program. The reasons for switching to MATLAB, how MATLAB was integrated into the existing introductory course, and some lessons learned after implementing MATLAB are described below. The purpose is to show both positive and negative aspects of switching to MATLAB in conjunction with recommended actions to effectively incorporate MATLAB into an existing course.

Background

The switch to MATLAB was a result of three major issues. First, the various departments within the College of Engineering desired different software expertise. Although the freshman program was utilizing both TK Solver[2] and FORTRAN during a freshman's first semester, some departments preferred that C++ and/or MS Excel[3] be taught. Second, the Department of Mathematics' software requirements were different from the College of Engineering. The Department of Mathematics required either MATLAB or Mathematica[4], as determined by individual instructors. Third, engineering students were being swamped with requirements to learn software. Since the College of Engineering has no computer knowledge prerequisite for entry, most freshman engineering students were required to learn and use computer communications, word processing, spreadsheets, engineering and mathematics problem-solving software, and a higher level programming language during their first semester. During the second semester, students were additionally required to learn a computer-aided-design program. As a result, students were overwhelmed with an abundance of program syntax and formats. It is understandable that many students had problems during their freshman year. The question was how to best meet the expectations of the College of Engineering and Department of Mathematics while still providing each engineering student a challenging, yet manageable, freshman year.

College of Engineering actions resolved the first major issue. As part of an ongoing review of departmental requirements, the freshman and sophomore curricula were altered. A freshman engineering student is no longer required to take a higher-level programming language as part of his/her first semester introductory engineering course. The programming requirement has been shifted to beyond the student's first semester. The engineering student now takes a separate course in either FORTRAN or C++ based upon her/his probable engineering major. As a result, a student is generally required to learn two software packages during the second semester. The first is the department required higher-level programming language. The second is the introduction to engineering design software package, Mechanical Desktop[5]. The benefit is the removal of the higher-level programming burden from the freshman engineering student's first semester transition period into college.

The second major issue was resolved by College of Engineering action. It involved engineering department compromises and software coordination with the Department of Mathematics. The question was, "What problem-solving software would meet most engineering **and** mathematics expectations?" The main software proposals were TK Solver, MATLAB, Mathematica, and MS Excel. After much discussion and deliberation, the engineering departments favored TK Solver for the following reasons:

- backsolving capability
- mathematical statements versus matrix format
  (no matrix knowledge required)
- minimal memorization of syntax
- some engineering departments were already using the software
- windows format driven
- limited tracing capability of numerical "units"

The Department of Mathematics, on the other hand, favored MATLAB for the following reasons:

- more & easier graphics capability
- easier to solve simultaneous equations
- basic programming was less complicated / more forgiving
- Department of Mathematics would accept as their standard software
- some engineering departments were already using the software
- less demanding up-front learning curve
  (matrix familiarity required)
- complemented the freshman linear algebra course

Since the third major issue was to streamline the freshman engineering software requirements and better meet overall engineering department desires, the final decision was to use MATLAB. Both introductory engineering and mathematics courses will be expected to use MATLAB beginning in August, 1999.

Using MATLAB in all first year courses for basic problem solving is beneficial for several reasons. It standardizes the freshman engineering student's experience which automatically fosters better communication / learning between students. It more efficiently utilizes valuable teaching time previously expended by the freshman engineering program and Department of Mathematics teaching different software programs. Thus, more time is available to cover course objectives other than learning how to use software. It also allows each student to become more fluent with one piece of software rather than having to divide time learning two or more problem-solving packages within the same period. Most importantly, MATLAB provides the freshman-engineering program and Department of Mathematics with a software package that meets both of their needs. As a result, the three major issues were accommodated by the decision to use MATLAB during an engineering student's freshman year. The next task involved implementing the decision.

Integration

Before addressing the integration of MATLAB into the existing first semester engineering course, it is imperative that one understands the course's objectives. These objectives require an understanding of the engineering problem-solving method and an ability to apply the method in conjunction with technical skills to solve problems. The quantitative objectives are stipulated in the course syllabus. The unstated and non-quantifiable / intangible objective is to nurture engineering students through a transition from varied backgrounds to a demanding college of engineering environment.

As such, the introductory course encompasses numerous "behind the scene" objectives. The course requires reading and studying administrative material required to efficiently function within the university environment. Each class also includes assigned readings and problems on both previously covered and new homework materials. Class presentations are focused on solving problems rather than simply covering material from pre-class reading assignments. Students are held responsible for reading and understanding simple concepts in preparation for class. Only more difficult concepts are reviewed during class periods. Daily quizzes are given to encourage pre-class preparation.

The purposes for this course structure are many. First, the students have an incentive to prepare for class. Second, the students are encouraged to develop their ability to read and understand written technical material. Third, the classroom learning environment is enhanced because students are better prepared. Students are able to take a more active role during classes. The result is more in-depth discussion rather than the instructor lecturing on self explanatory text topics while students take notes. This approach allows students to develop study skills beneficial to their continued success in college while learning the technical material and not wasting student/instructor class time.

The integration of MATLAB into the existing first semester engineering introductory course was beneficial and non-eventful. This is primarily due to MATLAB's less demanding up-front learning curve, the use of an easy to follow MATLAB text[6], the requirement to learn only MATLAB course supporting capabilities, and the previously mentioned course philosophy. The course's prior software package, TK Solver, required six (6) class periods. MATLAB required only 4½ periods to cover the same material in more detail. This resulted in a class timesavings of 1½ periods. These periods were allocated to other course material while more computer solutions, using MATLAB, were incorporated into the course's assigned homework problems. The necessary MATLAB classes are intertwined with the course topics that MATLAB's capabilities support. The primary uses for MATLAB in the course are for solving problems using simultaneous equations, graphing, and computer programming.

At the beginning of the course, one class period is used to introduce MATLAB to the students. This ensures students become familiar with material they will later see during assigned homework. The first class also provides an opportunity to display MATLAB's basic operations. The period is a "show & tell" for the students and is intended to alleviate initial student anxiety with MATLAB. The next step is to assign MATLAB homework from the MATLAB text while keeping in mind the underlying course objectives. The assignments are limited to teaching

simple MATLAB course supporting capabilities. These capabilities are presented by easy to follow text assignments having example problems and associated problems. During this period, handouts are also made available to assist each student toward a better understanding of the software. No class time is required or spent on the MATLAB material during this time frame. The basics are learned by students during these assigned self-study periods. Normal spot quizzes and MATLAB homework problems are used to evaluate and guide student learning.

After covering technical data display and problem solving using hand-drawn graphs, another full MATLAB period is inserted into the course to supplement the student's graphing knowledge. Here, the student is introduced to MATLAB's capability to plot both 2D and 3D graphs along with determining empirical equations for given test data. Emphasis is placed on using MATLAB's graphing capability to rework previously hand-solved problems. A student handout is again made available to supplement the course text and MATLAB learning experience. The handout consolidates MATLAB's syntax and format to allow plotting and empirical equation derivation for data based problems. This basic knowledge of MATLAB's capabilities fosters a less intimidating environment and a relatively smooth transition into programming with MATLAB.

In the programming arena, the course dedicates 2½ periods to introducing basic computer programming concepts. Selection and repetition constructs are introduced as well as the creation of functions. MATLAB's simplistic structure and forgiving nature is a major benefit to first-time programmers. A concern is that MATLAB may be too forgiving and may develop bad programming habits. This concern has yet to be validated by our higher-level programming instructors. MATLAB does, however, allow students to write and run simple problem-solving programs without a demanding learning curve. The result is a student who, without feeling overwhelmed, is capable of using MATLAB's basic operations and computer programming techniques to solve engineering problems. MATLAB also serves as a springboard for the students when they begin their follow-on high-level programming course(s).

Lessons Learned

There were numerous lessons learned from incorporating MATLAB into the introductory freshman engineering. The four primary lessons learned are provided below.

Course instructor opinions on MATLAB vary greatly. Some instructors categorize MATLAB as a mathematical software package not conducive to an engineering course. Other instructors maintain MATLAB is the ultimate software for providing a wide range of capabilities. The experience at VPI&SU has shown that the software package selection is secondary to meeting the course's main objectives. MATLAB, TK Solver, or MS Excel all provide the ability to solve engineering problems. Each has its positive and negative characteristics. It is important to realize that a software package is only a tool students use to solve problems. The ability to operate a specific software package is not the goal. As such, MATLAB served the course well, just as others might have.

A major lesson learned is the necessity to provide instructors an opportunity to learn about a software package before its inclusion in a course. Most of the course instructors had never used

MATLAB. The instructors were provided documentation, handouts, and every course problem solution. The instructors did not receive formal training on or extra time to learn MATLAB. Many instructors did not pick up their MATLAB course package until just before the course began. The lack of training and preparedness along with the increased workload imposed upon instructors caused by changing from TK Solver to MATLAB fostered an initial resentment toward MATLAB. Only after the course was complete and instructors were able to reflect upon their experiences did opinions on MATLAB change. If a software program is to be added or changed within a course, instructors must be given the opportunity to learn the software before teaching it.

The same lesson was reflected in instructor MATLAB text[6] reviews. Most course instructors were staying one step ahead of their students on text assignments. As a result, instructors were easily blindsided by student questions. This manifested an initial feeling of text inadequacy. Instructors stated; "The book was not clear enough", "The book could have done a better job", and "We need a better book next time". These statements illustrate instructor turmoil as they struggled to teach a software package without being properly prepared. After the course was completed and the instructors had time to reflect upon the text and their course experiences, the text received a more favorable review.

The dedicated MATLAB lessons also received favorable instructor reviews. One introduction, one graphing, and two programming (selection and repetition) lessons in MATLAB are a minimal course requirement. Additional lessons may be added, but they are not necessary. It is imperative, however, that the course fully integrates material covered in MATLAB lessons. This approach will increase the student's MATLAB learning experience without requiring more dedicated MATLAB time. MATLAB capabilities should not be displayed, discussed, and then stashed away. Daily homework problems should be given which require a student to use MATLAB. In this way, student problem-solving skills are constantly improved and expanded upon while using MATLAB but the technical use of a software program like MATLAB does not become a primary course objective. The course should fully incorporate MATLAB, as a problem-solving tool, in every class. At least one MATLAB homework problem is recommended for each lesson.

The above lessons can be easily accommodated. The long-term problem is the student in conjunction with the non-quantifiable course objectives. Students do not appreciate being required to read and implement a self-paced MATLAB study program. Although the material is simplistic and lends itself to individual study, many students would rather receive additional periods of in-class MATLAB presentations. The students do not believe the study of MATLAB in parallel with other course material is appropriate. In fact several instructors felt uncomfortable holding students accountable for material not specifically covered during classroom sessions. This fosters a constant debate among course instructors concerning teaching style and student responsibility. It is the author's opinion that not all assigned readings need to be covered during class periods. Freshman engineering students can and should be held accountable for reading and understanding well-presented text material. If this approach is not assumed, the integration of any software package will require additional class time, subsequently affecting other course objectives.

CONCLUSION

MATLAB is a satisfactory software program for any introductory freshman engineering course. If coordinated with mathematics requirements, the use of MATLAB can eliminate the requirement that freshman engineering students learn more than one problem-solving software package during their first semester. The software can be fully integrated into an existing course with little effort and provides a viable method of solving engineering problems with a computer. The study of MATLAB can be used as a prelude to future high-level programming languages and also can be used to develop better reading and studying skills in freshman engineers. The net result is a positive, efficient, and effective learning environment.

Bibliography

1. "MATLAB", MATrix LABoratory, is a trademark of The Mathworks, Inc..
2. "TK Solver", Tool Kit Solver, is a trademark of Universal Technical Systems, Inc..
3. "MS Excel", Microsoft Excel, is a trademark of Microsoft Corporation.
4. "Mathematica" is a trademark of Wolfram Research, Inc..
5. "Mechanical Desktop" is a trademark of Autodesk, Inc..
6. Joe King, Engineer's ToolKit, *MATLAB 5.0 for Engineers*, Addison-Wesley Select Edition (1998)

PATRICK DEVENS

Pat Devens is a professor at VPI & SU and teaches computer-aided-design, programming, and engineering fundamentals. He received his B.S. at the United States Military Academy and M.S. in Civil Engineering at VPI & SU. He has written many publications and developed & directed several engineering programs. His project accomplishments include a $23 million renovation and a $30 million new facility. He has managed annual facility operation/maintenance budgets exceeding $2.5 million and provided engineering support throughout the world.