TEACHING MATLAB TO A NON-CANONICAL AUDIENCE

Robert Talbert

rtalbert@franklincollege.edu

Department of Mathematics and Computing
Franklin College
101 Branigin Boulevard
Franklin, IN USA 46131

1 Computer Tools for Problem Solving

1.1 An introductory MATLAB course for mathematics students

As part of an effort to integrate computing, especially computer programming, into its mathematics degree programs in a more meaningful way, Franklin College recently created a one-credit laboratory course called **Computer Tools for Problem Solving** (desginated **CMP 150**). The course is a prerequisite for Calculus III and is intended to be taken concurrently with Calculus II. The main intent of the course is to teach students effective problem-solving strategies and heuristics by way of computer-aided investigations and to train students on a single piece of software that can be woven into a number of subsequent mathematics courses.

Students in the mathematics department take an introductory programming course that currently uses C++ as its language. However, since this course is not a prerequisite for any mathematics course, students tend to put this course off until very late in their college careers, and programming itself makes few meaningful appearances in mathematics coursework. Additionally, C++ is not the best match for the kinds of computing tasks that students can and should be doing in their mathematics coursework and will be doing in their careers. Few of our alumni report using C++ or a similar language to craft large-scale software projects; many of them report needing to write small, targeted, functional programs, rapidly prototyped and intended for a limited user base.

MATLAB was chosen as the software platform for the course because it meets all of the above criteria and serves multiple purposes. Students can use MATLAB for calculations, visualization, data analysis, programming, or some combination of these. It is the scientific computing platform of choice among many technical professionals, and it enjoys a diverse and active community of users. There is a wide variety of "toolboxes" available for specialized use ranging from signal analysis to financial mathematics to bioinformatics. We believe that if students learn to use MATLAB over the course of their college careers, they will have a significant advantage upon entering a job market that is increasingly oriented toward those with significant computing experience.

Using this course as a launching pad, MATLAB will be integrated vertically in the mathematics curriculum, most visibly in Calculus III, Linear Algebra, Differential Equations, and a rotating topics course in Numerical Analysis. Our intent is that these courses will contain significant assignments involving MATLAB, including programming assignments in which students will write simple, functional programs to automate repetitive numerical tasks, visualize information, and publish their results. The first running of CMP 150 took place during the Spring semester of the 2009–2010 academic year.

1.2 The canonical audience and the non-canonical audience

In designing the course, some marked differences were discovered between the kind of audience assumed by most introductory MATLAB materials and the audience actually enrolled in our course:

- *Most introductory MATLAB materials assume an audience composed primarily of engineers and scientists.* However, in the first CMP 150 class, 40% of the students were either Elementary Education majors seeking a middle school mathematics endorsement or Secondary Education — Mathematics majors. Another 20% of students were Pure Mathematics majors. The rest were scattered among other disciplines or were "undeclared" in their major. There was only one Engineering major, one science student (a Biology major), and one Applied Mathematics major.

- *Most introductory MATLAB materials assume a moderate-to-high level of experience and comfort with computers and programming around the level of an introductory programming course.* By contrast, only around 20% of students in CMP 150 had any programming experience at all, in or outside of an introductory course. When asked to rate themselves on a scale of 0 to 10 in terms of comfort level using computers to do technical tasks (such as mathematical tasks, and as opposed to nontrivial but nontechnical tasks such as word processing), the average rating was 6.4, with several students rating themselves at a 5 or below.

- *Most introductory MATLAB materials do not factor in the kind of institution in which the students work or the overall goals of the institution.* That is, being designed primarily for engineers and scientists, and knowing that typically only large universities have sizable populations of students in those areas, the problems, exposition, and learning objectives of these materials reflect the philosophy of a large university. However, Franklin College is a small liberal arts college with particular over-arching liberal arts goals that we intend to apply to all classes. Such goals include effective written and oral communication, critical thinking, lifelong learning, and cross-disciplinary application of knowledge.

These differences between the audience typically intended for an introductory MATLAB course (the "canonical" audience) and the audience actually garnered by CMP 150 (the "non-canonical" audience) created significant pedagogical and course design problems.

For example, in some introductory MATLAB textbooks, the first examples or exercises students see involve relatively esoteric science and engineering concepts such as the ideal gas law or free-body diagrams. Other books assume a higher level of mathematical or computing experience than CMP 150 students typically have, introducing vector functions or programming constructs as review rather than as core concepts.

The challenge, then, was to design an introductory MATLAB course that works for an audience of second-semester freshmen, with little to no computing background and only one semester of calculus completed. To do this, some relatively unconventional approaches to design and pedagogy had to be taken.

2 Design and Pedagogy

2.1 Design principles

The CMP 150 course meets once per week for 75 minutes in a computer laboratory. This setting, in addition to the audience makeup, requires several organizing principles to be kept in mind:

- *Place a strong and early emphasis on data and visualization.* Students at all levels tend to connect well with visual concepts and real data. One of MATLAB's strengths is in working with data and visualization — indeed, this is why many professionals use it — and a recurring thread of data and visualization serves as a meaningful and satisfying touchstone for students as they learn the intricacies of the software.

- *Connect MATLAB topics to things students already know.* Given that most students entered CMP 150 with very little computing background and minimal college mathematics experience, it is crucial to leverage whatever familiarity students have with mathematical or technical topics when introducing MATLAB concepts. Giving an assignment that uses the concept of the derivative and uses only basic MATLAB functionality, for instance, is preferable to one that uses topics from chemistry or physics (which not all CMP 150 students know) and requires students to know something about data structures or input/output commands.

- *Focus on strong understanding of a small set of basic concepts rather than a weak understanding of a large set.* Many MATLAB textbooks introduce complicated functionality that, while useful, might be better introduced later, once students have spent time learning the basics of the program. For example, the `fprintf` command is important in MATLAB but also very hard to understand for those not used to interacting via software with text files and the display or with arcane formatting options. Therefore the omission of this command, while sacrificing a piece of important MATLAB knowledge, might help students learn more MATLAB and make them more capable users of the command later.

- *Emphasize individual responsibility for learning and the use of Help documentation.* Given the time parameters of the course and the fact that it is a laboratory course, lecturing is essentially out of the question aside from short question-and-answer sessions. Instead, there is a wide variety of high-quality training material freely available online for MATLAB, and these can be used to replace the traditional lecture. Students should be held accountable for some degree of self-teaching and preparation for the weekly activities. This will free up time for students to use on in-class lab assignments; it properly addresses the liberal arts criterion of developing lifelong learning habits; and most importantly, it builds students' confidence and abilities as self-directed learners and makes them better prepared to use MATLAB in future courses.

- *Offer a high level of support.* While the emphasis is on individual responsibility, there should be a rich selection available of help for students as they work. This should be a combination of face-to-face help with the professor, collaborative opportunities with other students, and electronic sources of help.

2.2   The inverted classroom model

The pedagogical setting which best fits the design principles above is known as the *inverted classroom*. In a traditional classroom setting, class time is spent in a lecture format in which the professor transmits knowledge to the students; time outside of class is focused on assimilating the knowledge that was transmitted, typically through the working of homework problems. The inverted classroom is so called because these two activities are reversed, with the "transmission" phase taking place outside of class through the use of podcasts, videos, and/or readings, and the "assimilation" phase taking place inside class through learning activities that are monitored and assessed. Remaining time outside of class is spent on completing various kinds of learning activities.

A common feature of these activities is peer instruction ([4]), in which students participate actively in each others' learning. The inverted classroom model is used in such disciplines as economics ([3]) and software engineering ([1]) and is gaining ground as self-production and internet hosting of instructional videos becomes easier and cheaper.

The decision was made to use the inverted classroom model in CMP 150 for three reasons. First, it uses the time of the class — both the relatively short class meeting and the very long time in between class meetings — with maximum efficiency. Second, it promotes the design principle of self-directed learning through the use of outside resources for concept acquisition and Help documentation for specific syntax. Third, this approach harmonizes well with the course emphasis on problem-solving, rather than passive reception of ideas via a lecture, and the liberal arts philosophy of lifelong learning.

2.3 Outside-of-class work

Each week, students were given a document[1] consisting of an overview of the week's class activity, a list of outside reading and viewing, a list of basic competencies to have mastered before coming to class, and a collection of exercises designed to assess students' acquisition of fundamental concepts. Students were responsible for completing these tasks before the class meeting, which was on a Monday, and for asking questions and getting help on things they did not understand.

Students were allowed unlimited collaboration on the outside-of-class work provided their responses to the homework questions reflected their own understanding. Weekly quizzes (see below) and the final exam ensured that students remained honest with this freedom. They were also encouraged to come to office hours or use email to direct questions to the professor. Additionally, to encourage a group setting for help, a Google Group for the class was created for questions, class announcements, sharing of M-files, and discussion threads. The Google Group was intended to mirror the real-life use of internet discussion boards by users of operating systems and software for help.

It should be noted that we used no print textbook in the course. As mentioned earlier, none of the existing commercial textbooks for MATLAB that were reviewed was a good fit for the audience of this course. Initially, the course was to use *Experiments with MATLAB* ([5]), a free electronic textbook written by Cleve Moler, the creator of MATLAB. However, in preparing the course, the book was eventually deemed too technical and over the heads of most of the students in the course; we did use individual chapters and exercises from this book as a basis for lab activities.

2.4 In-class work

Every Monday's class began with a ten-minute multiple choice/short answer quiz given on Moodle and covering the main ideas of the week's outside-of-class work (as well as an occasional review question form a previous week). Since the quiz was given using a course management system, the grading was automatic, and data from the grades could be used immediately following the quiz to motivate a brief question-and-answer session. Students could also bring in other questions about the week's work for discussion prior to the main activity.

The main activity for each week was a lab, focusing on solving problems using the MATLAB concepts of the week. The early labs focused on visualization and data analysis with an emphasis on performing tasks in a variety of settings: at the command line, in an M-file, and using some of the specialized MATLAB environments such as the Plot Tools window. Programming was not introduced until almost mid-semester, in an effort to build student

---

[1]Here is a sample: http://bit.ly/8XhNax

353

comfort level and confidence with MATLAB and with computing in general before starting something that was brand-new to nearly all the students.

Labs were done in pairs or in threes to allow for collaboration and peer instruction. At first, the entire class worked on a single problem or pair of problems. As the course progressed, it became more favorable to introduce choice into the system. Labs were made up with several problems, and students chose a subset to work that week; or, in some cases, there were one or two required problems for everyone but also a number of extra credit problems that could be solved.

The lab found at the following address serves as a good example: `http://bit.ly/c9g31I`. Prior to this lab, students had watched about 30 minutes of videos from Khan Academy ([2]) on the basics of matrices, matrix operations, and solving simple linear systems. Students had performed simple matrix operations by hand and solved a simple linear system using MATLAB on their homework, and they had answered questions about these processes on their quiz. In the lab, students then solve a system to find a cubic polynomial going through three points and then work a problem involving the long-term behavior of population of students majoring in different programs at a college. Then students could work some basic extensions of their required problems, most involving visualization tasks, for extra credit.

Note that the lab adheres to the course design principles: Topics are either discipline-neutral (such as the cubic polynomial problem) or related to something with which students are already familiar; there is an emphasis on data and visualization; and the lab focuses only on one or two big ideas and avoids minutiae. However, even with its simplicity, one can still inject some rather sophisticated mathematics "under the radar", as we have done here with the idea of interpolation (Problem 1) and Markov chains (Problem 2). This lab was cited quite commonly by students at the end of the semester as a favorite.

2.5   Final exam

While no timed tests (other than the weekly quizzes) were given, there was a comprehensive final exam. The exam consisted of a multiple choice section (20 points), an essay/reflection section (12 points), and a problem-solving section (68 points). The problem-solving section can be found here: `http://bit.ly/avPi7h`. It consists of four main areas: Visualization, Working with Data, Mathematical Analysis, and Programming. Each area contained two or three problems to solve, and students selected one problem from each area. In this way, the exam allowed for student choice but still assessed the main areas of knowledge with which a well-trained beginning MATLAB user should be familiar.

The final exam was entirely open-technology. In addition to the use of MATLAB, students were allowed to use the internet in any capacity they wanted except for communication with each other. Therefore, for example, they could go search the class Google Group for answers to a particular syntax question from earlier in the semester, an internet discussion

board for help on a topic, or their own network drives for lab or homework problems they had worked out previously. The open-technology approach invites the sort of problem solving techniques that people in the real world employ and reinforces individual initiative in finding information to help them solve problems.

3 Results and Future Plans

3.1 Outcomes of CMP 150, Spring 2010

Students were initially wary, if not rebellious, of the inverted classroom approach. This pedagogical method is unlike that used in most of the classes they have had, especially their mathematics classes. Initially, some students interpreted the approach as an unwillingness on the professor's part to help. Students also found the outside-of-class work to be very time-consuming. The first problem was solved simply by communicating the value of self-directed learning and peer instruction to the students, particularly the fact that workers in the current economy are expected to take initiative for their own learning. The second was mainly caused by a tendency to forget one week's concept before the next week's lab; for example, many students forgot the process for plotting functions which they learned in week 2 when they came in for the lab for week 3 which used plotting heavily. Stressing the connections between weekly labs, and an overall increase in comfort level with the software, generally ameliorated this problem by mid-semester.

On the final exam, students were asked: "Say a few words about how you feel you've grown as a learner and problem-solving this semester." One student responded:

> I am now comfortable attempting a problem and trying to see where I went wrong and what other methods I could try. I feel that I am much more comfortable trying to figure computers out even if I am not completely sure where to start. This course has helped show me to try the problem before I give up. It has also helped me have more confidence in using computers for a number of different things.

Another said:

> I now have more experience learning how to solve problems on my own without outside help. I can learn by using the resources provided to me rather than by being told what to do. I know that solving a problem requires multiple attempts, and now I do not become discouraged when a problem takes me some time to complete. This type of learning will help me in the real world when I have to think critically and address problems as they arrive on my own.

Still another said:

> I started off feeling that the only way I was going to be successful in this class

was to have everything lectured, or spelled out, so to speak. As the course has gone on, I become more aware of what I was truly capable of. I am now willing and able to seek out resources to find answers, and my reasoning skills have increased. The thought process needed behind solving problems in MatLab [*sic*] has forced me to learn how to think outside of the box and to be open minded and brainstorm ideas to solve problems.

Therefore, the course appears to have succeeded not only in imparting a sound technical knowledge of MATLAB but in helping students become more competent, confident, creative, and persistent problem-solvers in general.

3.2 Adjustments for CMP 150, Spring 2011

There were some key elements of MATLAB that did not receive the attention they should, especially the ability to publish MATLAB work as PDF or HTML files and the post to the web. (That particular issue was due to not having a web space for hosting the files, and the college's Information Technology Services department is working to help.) Some programming concepts were not covered in an appropriate depth. Most importantly, many of the pedagogical elements of the course were created "on the fly" as the course unfolded and student needs became clearer. In the second iteration of the course, the precise parameters of student abilities and needs will be better known, and the implementation of the inverted classroom approach will hopefully be all the more smooth.

References

[1] G. Gannod, J. Burge, and M. Helmick. Using the inverted classroom to teach software engineering. Technical Report MU-SEAS-CSA-2007-001, Miami University, Department of Computer Science and Systems Analysis, School of Engineering and Applied Science, 2007.

[2] http://www.khanacademy.org.

[3] M. Lage, G. Platt, and M. Treglia, Michael. Inverting the classroom: A gateway to creating an inclusive learning environment. *Journal of Economic Education*, **31**(1):30–43, Winter 2000.

[4] E. Mazur. Peer instruction: A user's manual. *American Journal of Physics*, **64**(4):359–360, April 1999.

[5] C. Moler. *Experiments with MATLAB*. Electronic textbook at http://www.mathworks.com/moler/exm/chapters.html.