

was used. To quickly recall this command, simply type “>> t,” then press the up arrow. I learned this trick by accident.

- » Matlab is very inefficient when the code includes loops, so avoid loops as much as possible. For example, initializing a matrix as follows:

```
>> for i = 1:1e8
    X(i) = i;
end
```

can take tens of seconds to complete, whereas the following code take less than a second:

```
>> X = 1:1e8;
```

Additionally, execute

```
>> x'*A*x;
```

rather than using nested for loops.

- » If you are curious about the time it takes to execute your Matlab code, use the “tic” and “toc” feature, for example:

```
>> tic
>> for i = 1:1e8
    x(i) = i;
end
>> toc
```

Elapsed time is 22.460566 s.

```
>> tic
>> x = 1:1e8;
>> toc
```

Elapsed time is 0.404674 s.

Additionally, the Profiler lets you run a program and then look at how long each line took to execute. These tools are very valuable for improving the efficiency of

A Call for Matlab Tips

Readers are encouraged to submit their Matlab tricks to jva@calvin.edu. The best tricks will be included in future issues of *IEEE Control Systems Magazine*.

your Matlab code. Visit the Mathworks Web site [8] and search for “Profiling for Improving Performance” for more details.

- » It is usually a good idea to allocate memory for vectors and matrices ahead of time, provided that the maximum size is known a priori, rather than adjusting the size as you go. This way, the matrix elements will be stored in contiguous locations in RAM, making for more efficient calculations.

REFERENCES

- [1] Mathworks. (2013). [Online]. Available: <http://www.mathworks.com/products/daq/>
- [2] K. Ogata. *Matlab for Control Engineers*. Upper Saddle River, NJ: Pearson Prentice Hall, 2008.
- [3] R. V. Dukkupati, *Analysis and Design of Control Systems Using Matlab*. New Delhi, India: New Age International Ltd., 2006.
- [4] A. Tewari, *Modern Control Design with Matlab and Simulink*. New York: John Wiley, 2002.
- [5] Scilab. (2013). [Online]. Available: <http://www.scilab.org/>
- [6] GNU Octave. (2013). [Online]. Available: <http://www.gnu.org/software/octave/>
- [7] Physical Oceanography. (2013). MATLAB tips & tricks. University of California, San Diego, CA. [Online]. Available: <http://www-pord.ucsd.edu/~matlab/>
- [8] Mathworks, Inc. (2013). [Online]. Available: <http://www.mathworks.com>

Controls for the Masses

MAGNUS EGERSTEDT

During the spring of 2013, I taught a massive open online course (MOOC), Control of Mobile Robots, to over 40,000 students worldwide. Online resources for higher education have received significant attention during the last year, but upper-level engineering classes have been virtually absent from all the major MOOC content providers (Udacity, Coursera, edX).

WHY A CONTROLS MOOC?

The reason I decided to pursue a controls MOOC can be found on the campus of Georgia Institute of Technology (Georgia Tech). A few years back, I included a robotics project in our introductory undergraduate control course. The

students had to design controllers that would take differential drive robots through a sequence of waypoints in the shortest amount of time by carefully tuning PID parameters, as shown in Figure 1. It was a resounding success! The students were suddenly much more excited and engaged in the class, and as a control educator, it was absolutely wonderful to hear groups of students argue about the relative merits of high I-gains. The next year I decided to increase the robotics content in the course by having the students design complete navigation systems that would make the robots negotiate cluttered environments in a safe and effective manner. But this ended up being just as much of a failure as the previous incarnation had been a success!

The students stopped being systematic about their control design choices and instead ended up hacking together complex and cumbersome solutions. This observation

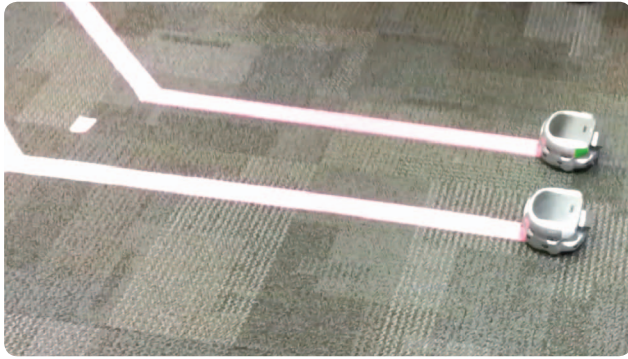


FIGURE 1 A robotics competition as part of the introductory controls course at Georgia Tech.

motivated the development of the MOOC. I really wanted some way of making the robotics projects more elaborate and exciting, yet I did not want to compromise on the technical content of the class. So I decided to outsource myself by doing a so-called flipped classroom [1], where the students obtain the technical content from some other source (the MOOC) and then apply that content to robotics projects in class.

In the interest of full disclosure, there are two other reasons why I decided to launch a MOOC. First, it is clear that online education is indeed a disruptive technology and that *it* will have significant impact on how higher education will be conducted in the future [3], [4]. It is just not clear what *it* actually means. As such, I was interested in seeing what the fuss was all about and to get first-hand experience with online courses. Second, since MOOCs are indeed gaining a lot of momentum, I wanted to be part of defining what engineering MOOCs should look like, rather than to sit back and wait for someone else to do it for me.

PRELAUNCH MECHANICS

Producing an MOOC takes a lot of time. The general wisdom is that an online course takes three times as much time to produce as a regular class. In my experience, that factor of three is an underestimate.

I started preparing for the MOOC in the summer of 2012 by first thinking through the outline of the course and deciding what content was needed to tell the story that I wanted to tell. One aspect of the MOOC that I was particularly worried about was the level of mathematics needed to do control theory well. I decided to embrace rather than shy away from the mathematics, and the outline of my seven-week course, which was provided through Coursera, was

- » Week 1: Introduction to Controls
- » Week 2: Mobile Robots
- » Week 3: Linear Systems
- » Week 4: Control Design
- » Week 5: Hybrid Systems
- » Week 6: The Navigation Problem
- » Week 7: Putting It All Together.



FIGURE 2 A sample lecture.

Each week was broken down into eight sublectures. Online education research has found that people cannot concentrate for more than 11 minutes or so, and all MOOC providers are following this model of breaking down course material into nuggets [2]. This was a rather useful experience in that it forced me to think about what the appropriate nuggets should be. As an example, Week 4 (Control Design) was broken down as

- 4.1) Stabilizing the Point Mass
- 4.2) Pole Placement
- 4.3) Controllability
- 4.4) Segway Robots
- 4.5) Observers
- 4.6) Observability
- 4.7) The Separation Principle
- 4.8) Practical Considerations.

At the end of the summer, I finished the story and started to produce lecture slides. The lectures were recorded during the fall of 2012. Each lecture consisted of my talking head superimposed over lecture slides that I could annotate in real time using a tablet, as shown in Figure 2. The lectures were supplemented by robotic experiments and demonstrations and were recorded at the Georgia Tech Global Learning Center. Without access to a studio and a recording and editing staff, the recording process would have been significantly more time consuming.

Launching an MOOC without having completed all of the recordings is a really bad idea. I have colleagues at Georgia Tech who made this mistake, and they had to play catch-up rather than being able to focus on the discussion forums and homework assignments.

It is impossible to grade 40,000 homework submissions individually. The remedy to this is automatic grading of multiple-choice questions. Not only does the instructor have to produce interesting questions with good answers, but the instructor also has to produce multiple interesting and good (albeit incorrect) answers. An example of such a multiple-choice question is shown in “An Example Question.” The production of good assignments is key to creating a meaningful learning experience, as the students

An Example Question

Assignment 3.4: Let the output of a third-order system ($\dot{x} = Ax$, $y = Cx$) be given by Figure S1.

The options in Figure S2 show the possible placements of the eigenvalues to the A matrix, with the axes being the real and imaginary parts of the eigenvalues. Which option corresponds to the system used to generate Figure S1?

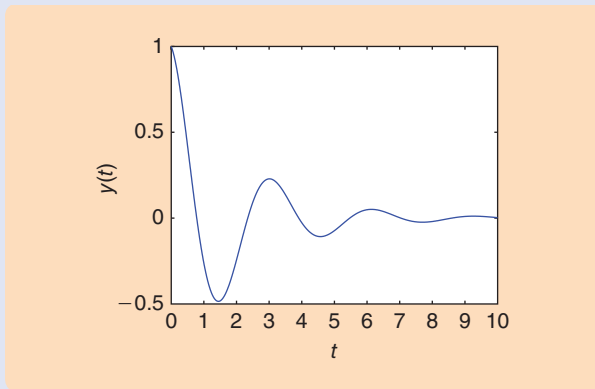


FIGURE S1.

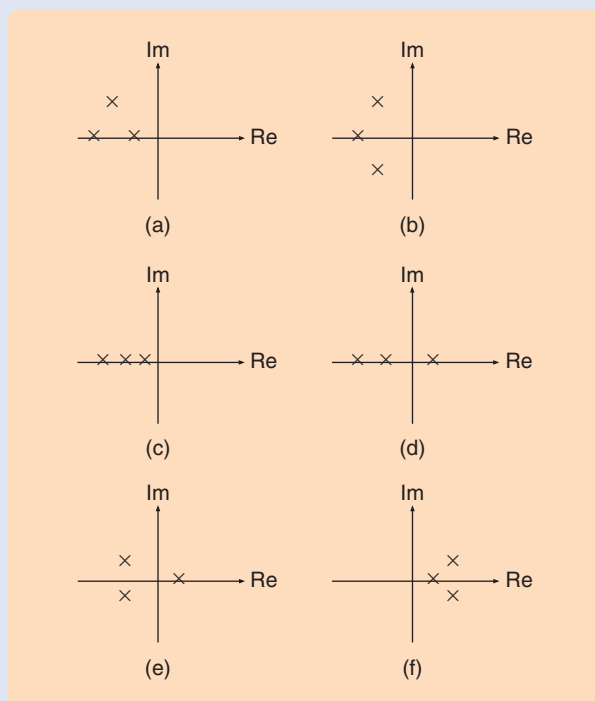


FIGURE S2.

spend a lot of time thinking about the questions and forming study groups—physical as well as virtual.

CLASS CHRONOLOGY

The class went live at the end of January 2013, with over 40,000 registered students. This number is somewhat misleading since many students sign up for a MOOC and then never participate. In fact, the number of active participants during the first week (i.e., students that had watched a video, taken a quiz, or participated in the discussion forums) was closer to 16,000.

During the first week, a small but vocal minority of the students was somewhat upset by the fact that they had not signed up for a “calculus class,” or, to quote one student on the forum, “Oh Bollocks! Matrices and integrals together!” At the end of the first week, the number of active participants was down to about 12,000. Despite this, I decided to stick to my guns and not shy away from the mathematics, which seems to have been the right decision. During the second week, I started getting e-mails thanking me for not “dumbing down the course,” and some students remarked that finally all that linear algebra that they had taken some 20 years ago was paying off.

The course quickly settled at around 9000 active participants, with about 6000 taking the weekly quizzes. A particularly pleasant surprise was that the students were helping each other on the discussion forums,

creating study groups, and searching for good online resources. The MOOC ended at the end of March, and I handed out about 4000 certificates, which is well above the typical 6% retention rate [5]. To obtain a certificate, a student needed to average 60% or higher score on the quizzes, and a 90% or higher score was needed to pass the course with “Distinction.” Almost half of the students that received certificates passed the course with distinction.

At the same time, my flipped on-campus class was progressing nicely, and the students built robotics projects of increasing complexity (see Figure 3 for a robot and one of the obstacles). To pass the course, students had to be granted a certificate from the MOOC as well as complete all of the robotics projects. The projects involved making the robots complete the following series of tasks:

- 1) Drive in a specified direction.
- 2) Drive through a sequence of goal points.
- 3) Avoid hitting obstacles.
- 4) Combine obstacle-avoidance and goal-seeking controllers without introducing chattering.
- 5) Follow obstacle boundaries.
- 6) Navigate through cluttered and nonconvex environments.

Each student had a different final project, which covered such topics as the combination of machine learning with control design, path planning with rapidly exploring

randomized trees, and implementation of simultaneous localization and mapping algorithms on the robots.

Even though the on-campus class is not yet over (at the time of writing this article), it has been quite successful, and the problem that I encountered the year before, of students hacking together solutions, has completely gone away. Instead, complex robotic behaviors are being designed in a systematic manner, using observer-based state feedback, hybrid automata, and linear-quadratic optimal control.

THE GOOD, THE BAD, THE UGLY

I learned quite a bit from the MOOC in terms of how to divide and organize course material into appropriate chunks. The experience also reinforced the importance of having a clear and compelling arc through the material. But as I have already hinted, some things went well, and others did not go quite according to plan.

The negatives of developing and teaching a MOOC were

- » **40,000 is different from 40:** The amount of e-mail traffic surrounding the MOOC is sometimes overwhelming, and the discussion forums were highly active. This meant that every morning and every night, I had to devote 30 min (or more) to managing the course.
- » **An incredible time sink:** It takes significantly more time to prepare and execute a MOOC than a standard class. I was not entirely prepared for this, and without the reduced teaching load that Georgia Tech granted me, it simply would not have been possible to pull this off.
- » **Engineering is hard:** Almost all MOOCs fall into one of three categories: humanities, computer science, or “introduction to X.” There are virtually no upper-level engineering MOOCs, for good reason—engineering requires prerequisites, such as calculus, linear algebra, and Laplace transforms. Moreover, engineering courses are made better with hands-on labs, and there is simply no way that meaningful, physical laboratories can be made a part of a free online course, given the current state of technology.
- » **Internet people can be quite mean:** When I forget a minus sign in a regular lecture, a student will raise his or her hand and point this out. No harm done. If I forget a minus sign in the MOOC, a firestorm will erupt on the discussion forums, with some comments pointing out how utterly incompetent the instructor is.

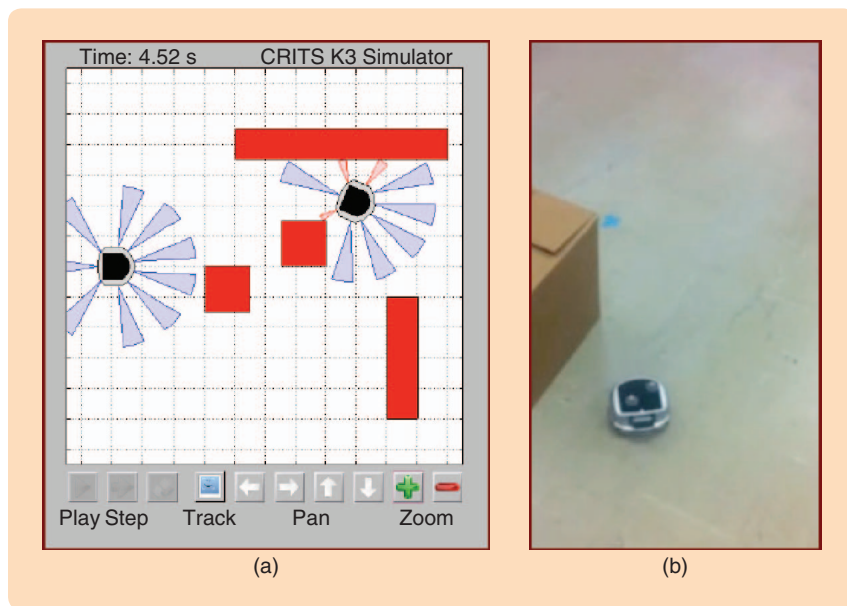


FIGURE 3 A flipped classroom robotics project. (a) Graphical user interface showing two robots and four obstacles to be avoided, with students implementing their control algorithms in Matlab. (b) A photograph of one of the Khepera III robots near one of the obstacles.

Despite these negatives, the experience was overall very positive.

The positives of developing and teaching a MOOC were

- » **Appetite for serious engineering content:** The biggest and most surprising positive aspect of the controls MOOC is that there really is strong demand for upper-level engineering content that does not skimp on the mathematics.
- » **Flipped classrooms:** My experience flipping the classroom at Georgia Tech in conjunction with the MOOC worked out very well, and I intend to follow this model in the future.
- » **Incredibly rewarding:** A nontrivial aspect of teaching a large MOOC is that you are reaching people all over the world that you would otherwise have no chance of reaching. This democratizing aspect of the MOOC experience should not be understated.
- » **Engineering leadership void:** As a final note, there really are not any (or at least many) upper-level engineering MOOCs out there, and there is a golden opportunity to fill this void. There is ample room to be part of defining what engineering MOOCs in general, and controls MOOCs in particular, should look like.

I had a great experience teaching the MOOC Control of Mobile Robots, and I believe that it is important for us, as a community, to play a major part in defining what this online disruptive learning technology will eventually end up looking like.

AUTHOR INFORMATION

Magnus Egerstedt is the Schlumberger Professor in Electrical and Computer Engineering at the Georgia Institute of Technology. His research interests include hybrid and networked control, with applications in motion planning, control, and coordination of mobile robots. He is the director of the Georgia Robotics and Intelligent Systems Laboratory (GRITS Lab), a Fellow of IEEE, and serves as deputy editor-in-chief for *IEEE Transactions on Control of Networked Systems*.

REFERENCES

- [1] B. Tucker, "The flipped classroom," *Educ. Next*, vol. 12, no. 1, pp. 82–83, Winter 2012.
- [2] S. A. Clemons, "Brain-based learning: Possible implications for online instruction," *Int. J. Instructional Technol. Distance Learn.*, vol. 2, no. 9, pp. 25–34, Sept. 2005.
- [3] *What Campus Leaders Need to Know About MOOCs*. Louisville, CO: EDUCAUSE Publications, Dec. 20, 2012. [Online]. Available: <http://net.educause.edu/ir/library/pdf/PUB4005.pdf>
- [4] J. Gains, *Will Free MOOCs Destroy Higher Education?* Cambridge, MA: Digitopoly, Mar. 26, 2013.
- [5] A. Watters, *Dropping Out of MOOCs: Is It Really Okay?* Washington, DC: Inside Higher Education, July 23, 2012.

Control Systems Take-Home Experiments

MUSA K. JOUANEH and WILLIAM J. PALM, III

Most electrical and mechanical engineering curricula include courses in controls and mechatronics, with mechanical engineering curricula also including system dynamics and vibrations courses. The laboratory component for these courses is often limited and involves using a limited number of experimental setups. At many institutions, the laboratories associated with these courses are not taken in the same semester, preventing students from practicing the concepts learned in the lecture in a timely manner. Even in laboratory courses that are offered in the same semester as the lecture courses, in many cases it is not possible to synchronize the concepts covered in the lecture with the laboratory exercises since there are usually only a few lab setups for each experiment.

While increased lab time is needed, students at many schools may work at outside jobs or live far from campus. For safety and cost reasons, most laboratories are only open during normal business hours, severely limiting the times at which students can access equipment. This type of scheduling conflict makes it harder for students to have enough time to perform an experiment in the university laboratory. On the other hand, almost all students have home computers that are suitable for take-home experiments. This accessibility makes it possible for students to perform an experiment or obtain measurements outside the lab at a time that is convenient for them, just as they would with a homework assignment.

Providing engaging laboratory experiences is one of several challenges to effective undergraduate education in science, technology, engineering, and mathematics disci-

plines [1]. Control systems concepts are often perceived by the students as a "large collection of abstract mathematics" [2]. An experiential approach, such as that offered by take-home experiments, shows the role of the course material in engineering devices and systems. Additionally, providing personalized learning is one of the 14 Grand Challenges for Engineering in the 21st century [3]. Take-home kits allow every student to perform experiments at his or her own pace, tailored to the student's individual learning needs.

This article addresses the development and implementation of take-home hardware kits and software that can be used to perform laboratory experiments and measurements to improve the understanding of system dynamics and control concepts in an undergraduate student population. In this project, we have developed five experiments (liquid thermal response, dc motor speed control, free and forced vibration response, and temperature control) that were used in various courses in the mechanical engineering curriculum at the University of Rhode Island. This article describes the two control experiments that were developed: a dc motor with tachometer and a plate with a heater. Rather than having students perform an experiment in the university laboratory, the students are given a compact low-cost kit with which they can perform the experiment at home using their own computer. The kits are designed so that the experiments can be conducted on the provided experimental setup or can be used to perform dynamic measurements on other systems that are available at home such as motor-powered devices or heating/cooling systems.

A survey of the literature showed that there is an increasing interest in performing measurements and experimentation in engineering programs outside of the traditional university laboratory. Take-home experiments