Multidisciplinary Lab-Based Controls Curriculum

Gregory L. Plett, David K. Schmidt
University of Colorado at Colorado Springs

Abstract

This paper describes a multidisciplinary lab-based controls curriculum under development. One of the main focuses of the lab is that it be a multidisciplinary facility. It is shared by Electrical and Computer Engineering (ECE) and Mechanical and Aerospace Engineering (MAE) students. This arrangement allows more efficient use of space and equipment, better use of funds, and elimination of overlap among individual departmental labs. Interdisciplinary instruction also adds to the richness of both the ECE and MAE curricula.

Another main focus of the lab is that it include visually stimulating physical devices to control. A very comprehensive undergraduate controls lab has been developed around controlling *Educational Control Products* Magnetic Levitation systems. Using a single general-purpose device for all laboratory experiments rather than a plurality of devices (which each have a special purpose) results in economies of space, money, and student time (as only one device needs to be thoroughly understood; hence, more time may be devoted to studying how control-systems theory applies to it).

The laboratory we have built comprises four work centers. Each work center has a Magnetic Levitation system to control. These devices may be configured to study control of linear or nonlinear, stable or unstable, SISO, collocated SIMO, noncollocated SIMO and full MIMO control. Control is accomplished using a *Comdyna* GP-6 analog computer or a digital computer running the Real Time Linux operating system, via *MathWorks*' Matlab/ Simulink/ the Real Time Workshop (RTW) and *Quality Real-Time Systems'* Real Time Linux Target (RTLT).

I. Background and Goals

The control-systems laboratory at the University of Colorado at Colorado Springs (UCCS) needed attention. Occupying a small dark room, the lab comprised a few *Comdyna* GP-6 analog computers,[1] some decaying test-and-measurement equipment and one operational X-Y recorder. Then, the ceiling started to leak.

It was a simple matter to get the leak fixed, paint the room and improve the lighting; however, major deficiencies remained. The control-systems lab had not a single device to control! All lab experiments were accomplished via simulation, either on an analog computer, or on one of the lab's digital computers using Matlab and Simulink by *MathWorks*.[2]

Simulation using either method has its limitations. The need to control real hardware, and not just simulations, is known to all who design and build real control systems. How this applies to control-systems education is emphasised in a paper by Bernstein.[3] Modeling and simulation rarely capture the complete picture—physical system identification is required; control experiments often focus attention on performance and implementation issues that are overlooked and difficult to capture in simulation; experiments can reveal whether or not assumptions made when making a control design are realistic; and experiments provide a way to identify control methods that seem to work under real-world conditions as well as those that clearly don't. This final point leads to real learning.

Bernstein's paper discusses a number of devices built to demonstrate different control concepts. A problem we had with duplicating his approach was that we could neither afford the time to build these experiments ourselves, nor did we have the budget to outfit many workstations. We also felt that it would be more beneficial to the students if we required that they learn to control many aspects of a single device. Then, the dynamics of only a single system need be thoroughly understood; hence, more time may be devoted to studying how control-systems theory applies to it.

We came across another article promoting the control-systems laboratory at the University of Illinois at Urbana-Champaign.[4] An appealing quality of this facility is that it is shared among several departments. The control-systems laboratory at UCCS had previously been owned and operated by the ECE department, but a new MAE program in the college also needed similar facilities.

We concluded that a revived control-systems laboratory was essential, and we formulated two goals:

1. Hands-on: The new lab should promote control-systems education with experimentation, requiring identification and control of physical device(s). The laboratory course should be designed to complement and synchronize with the lecture course in order to best reinforce concepts learned in class with hands-on experience.
2. Economy: As much as possible, space, money and student time should be economized. A multidisciplinary facility, shared between ECE and MAE classes would allow efficient use of space and equipment, better use of available funds, and elimination of overlap among individual departmental labs. Focusing experiments on a single device rather than a plurality of devices would result in economies of space, money and student time.

Grant DUE–981009 from the National Science Foundation Directorate of Undergraduate Education has allowed us to accomplish these goals. A description follows.

II. Choice of Lab Devices

We decided to base our new lab primarily around the Magnetic Levitation (MagLev) Unit and Control-Moment Gyroscope (Gyro) Unit by *Educational Control Products* (ECP).[5] These two devices are shown in Figure 1. Together, they exhibit many important properties of dynamic systems from the point of view of control theory. A matrix of important attributes in dynamic devices, as well as the coverage by specific devices is listed in Table I.

Figure 1. The two lab devices. The MagLev device is to the left; the Gyro device is to the right.

TABLE I

Attractive attributes of the selected dynamical devices.

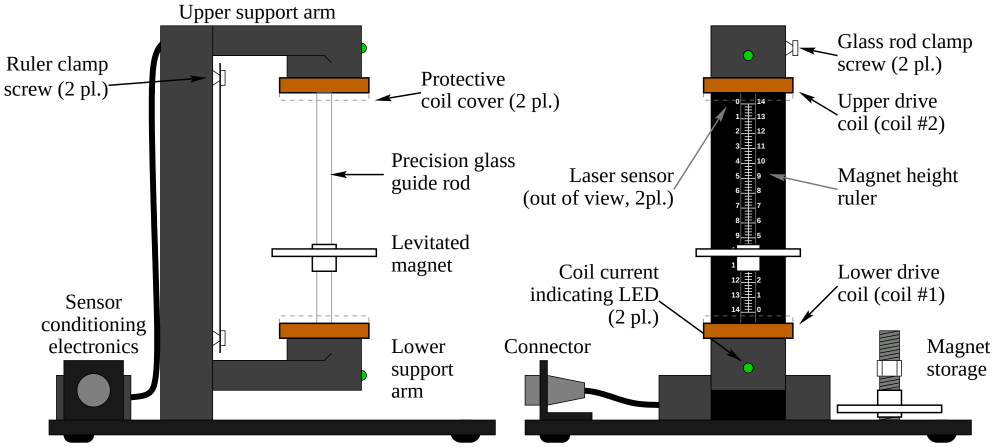
Desirable dynamic attribute	MagLev	Gyro
1. Linear single-variable, stable	Y	Y
2. Linear single-variable, unstable	Y	Y
3. Nonlinear single-variable, stable	Y	Y
4. Nonlinear single-variable, unstable	Y	Y
5. Linear multi-variable, little I/O interaction	Y	N
6. Nonlinear multi-variable, large I/O interaction	N	Y
7. Dynamically rich system	N	Y
8. Electromechanical system	Y	Y

The MagLev (described in more detail in Section III) may be used to exercise many skills. It can be configured as open-loop stable or unstable, so may be used to teach practical concepts of stability and stabilization. It may be configured as a single-variable system (controlling the position of a single disk) or as a multi-variable system (controlling two disks). Additionally, the plant is nonlinear, so techniques for small-signal and feedback linearization must be employed. In small operating ranges it is approximately linear, so standard linear control techniques work. Not to be underestimated, this device provides dramatic and interesting demonstrations. The actuators and sensors are clean, high-quality devices, and the entire system is ruggedly constructed. This device is especially well-suited to demonstrate analysis and design techniques taught in classical analog and digital control courses, and to teach introductory modern analog and digital control.

The Gyro may also be used to exercise many advanced skills. It is not currently used in the undergraduate laboratory, so we do not discuss it in further detail here. It will be used in more advanced controls courses to demonstrate multivariable control, specifically for a dynamically rich system with large input-output interaction.

III. Description of the MagLev Device

Two views of the magnetic levitation system are depicted in Figure 2. Upper and lower electromagnetic drive coils produce a magnetic field in response to a dc current. One or two magnets travel along a glass guide rod. By energizing the lower coil, a single magnet is levitated by a repulsive magnetic force. As current in the coil increases, the field strength increases and the levitated magnet height is increased. For the upper coil, the levitating force is attractive.



Side View

Front View

Figure 2. Two views of the MagLev device.

The magnets are of an ultra-high field strength rare earth (NeBFe) type. A dry-lubricated guide bushing at the center of the disk slides up and down the rod. A white reflective surface covers most of the disk. Two laser-based sensors make use of the reflective properties of the disk surface to measure the magnet positions. The laser beams are spread by an optical element into a fan shape and are projected onto the diffuse white surfaces of the magnets. Photodetectors view the beams and generate voltages proportional to the amount incident beam power. The lower sensor is typically used to measure a given magnet's position in proximity to the lower coil, and the upper one for proximity to the upper coil (both $\approx 8\,cm$ range). Sensor-conditioning circuitry makes the design immune to stray light noise, such as turning room lights on and off, and rejects most induced electronic disturbances. Thus a relatively low noise signal is output from the amplifier box.

For many control scenarios, a general-purpose PC is used as the controller. An interface card in the PC contains D2A and A2D circuits connected to a "breakout box" which the student can access. A power amplifier/sensor conditioner box drives the MagLev device. The student may connect the breakout-box signals directly to the amplifier box using "banana cables." A pictorial description of the system setup is shown in Figure 4. The software running on the PC is discussed in Section V.

IV. Hardware Interface

In order to interface the MagLev and Gyro units to the host computer, a data acquisition card for the PC is needed. The MagLev requires two standard D2A and four standard A2D channels, and the Gyro requires four optical encoder inputs in addition to two standard D2A outputs. We also wanted to find a board which would work with both Real-Time Windows Target (RTWT) by *MathWorks* and Real-Time Linux Target (RTLT)[6] by *QRTS*[7] as we were initially unsure which operating-system we would choose to use on the host computer. Boards with both optical encoder inputs as well as A2D and D2A channels are rare. Finding a compatible set of hardware and software drivers was also a challenge.

One solution we considered was to use a general-purpose DSP board to perform data acquisition. The board would be outfitted with sufficient analog I/O, and could be programmed to decode encoder inputs using digital I/O. An advantage of the DSP approach is that high control-loop bandwidths are often possible due to the efficiency of the DSP. A disadvantage is expense.

The other solution we considered was to use a generic I/O card with encoder inputs. This is a less-expensive approach, but places the control-loop processing burden on the main system CPU and so tends to decrease the control bandwidth which can be achieved.

Our search lead us to four I/O boards which could meet our purposes from a hardware point of view: The interface board *ECP* sells with their control devices, the *Humusoft*[8] MF604, the *Quanser*[9] MultiQ and the *ServoToGo*[10] Model 2. At the time we made our decision, the *ECP* board was only supported by *ECP* proprietary software (it is now supported under RTWT and RTLT via software from *QRTS*), the *Humusoft* board was supported under RTWT only (and did not have enough I/O to support some lab devices not described here), and both the *Quanser* and *ServoToGo* boards were supported under RTLT only. Helping our decision, we received indication from *QRTS* that they had plans to support the *ServoToGo* board under RTWT (it now is). We chose to use the *ServoToGo* board because it allowed us flexibility with regard to operating system, and was the most cost-effective solution.

Among other features, the *ServoToGo* board supplies eight A2D channels, eight D2A channels and eight encoder input channels. This is more than sufficient for the needs of our laboratory.

V. Software Interface

In order for students to access the I/O board and control the MagLev, a software interface to the board is required. Rather than requiring that the students write C-language code and interrupt-service routines (much less, debug same) we chose to use a Matlab/ Simulink/ RTW/ RTLT interface.

Matlab is a software environment that provides great computational power and professional graphical output. The Control-Systems Toolbox, in particular, greatly aids control-system analysis and design. Simulink is a block-diagram graphical-user-interface based simulation package which works in within the Matlab environment and allows linear and nonlinear, continuous-time and discrete-time simulation. An example Simulink diagram is displayed in Figure 3.

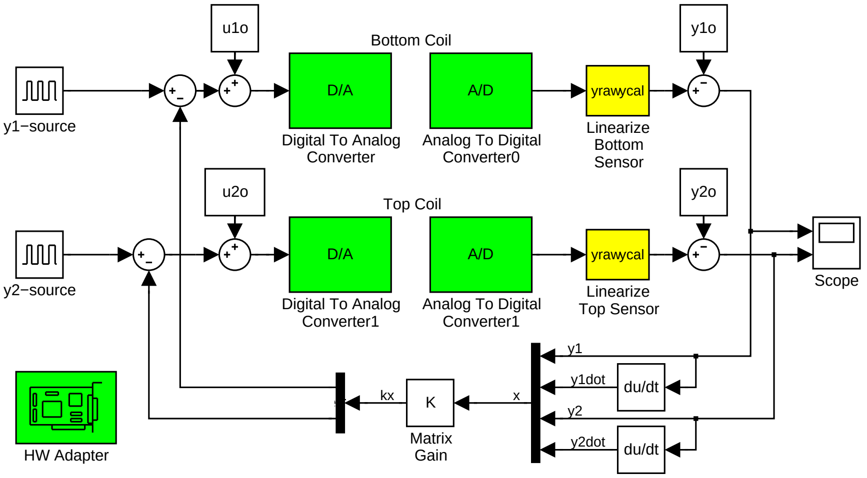


Figure 3. Example Simulink diagram to perform full MIMO control of MagLev.

Simulink does not directly access the *ServoToGo* interface board. Rather, the *Real-Time Workshop* (RTW)—accessible from Simulink through a pull-down menu item—writes generic C code to implement the Simulink block diagram. Then, either the *Real-Time Windows Target* (RTWT) or *Real-Time Linux Target* (RTLT) is used to generate host-machine specific code and execute it. All of this happens with the click of a mouse; the student writes no code at all!

Using the host CPU to execute the control algorithm—as done with RTWT and RTLT—results in economies since a separate DSP or CPU is not required. However, other performance issues arise. Specifically, traditional computer operating systems such as MS-DOS, Windows 95/98/NT/2000, Unix and Linux are not designed for real-time operation. Although RTWT is available, there is debate regarding whether or not Windows NT is yet a reliable platform for real-time systems.[11,12] There is no guaranteed maximum latency between an interrupt and its service, which makes real-time applications unpredictable.

Other alternatives include QNX or VxWorks, commercial real-time operating systems. These have been used for educational purposes before; an example is the QMotor program developed for QNX.[13] We decided not to use these systems due to the costs involved.

The alternative we chose is Real-Time Linux, or RTLinux. Linux[14] is a free Unix-like operating system for i386 (and family), Alpha and Sparc processors. By itself, Linux is not suitable for real-time systems, but a free patch called RTLinux adds functionality to Linux to allow real-time code to execute.[15,16] One paper has already described an RTLinux approach to control education, using Matlab and Simulink, but not using RTW and RTLT.[17] A disadvantage of this implementation is that we would need to write code to interface with the I/O boards; The Matlab/ Simulink/ RTW/ RTLT system does not require that we write any code at all.

One disadvantage of using RTLinux is that basic Unix system-administration skills are required by the lab administrator in order to set up and maintain the lab computers. This was not a problem for us as we have some experience in this area. Another potential disadvantage that concerned us is that our students tend to be more familiar with Windows than Linux. We wanted the lab to provide control-systems education and not operating-system education! So, we were careful to set up the lab computers to minimize the specific knowledge of Linux required (almost none), and students do not seem to have been deterred.

Figure 4 shows a pictorial representation of the entire control system, including software and hardware components. The student works within a Matlab environment where he or she enters a block diagram into Simulink. When the diagram is complete, the student selects “RTW Build” from the Tools menu, and RTW generates C code and builds it with help from RTLT. The student can then start the code running using “Start” from the simulation menu. The code runs as a kernel module, accessing the MagLev through the *ServoToGo* interface card. Signals from the card are routed through a cable to a breakout box. The student wires the breakout box to the appropriate power amplifier using “banana cables.” This setup has proven to be very usable.

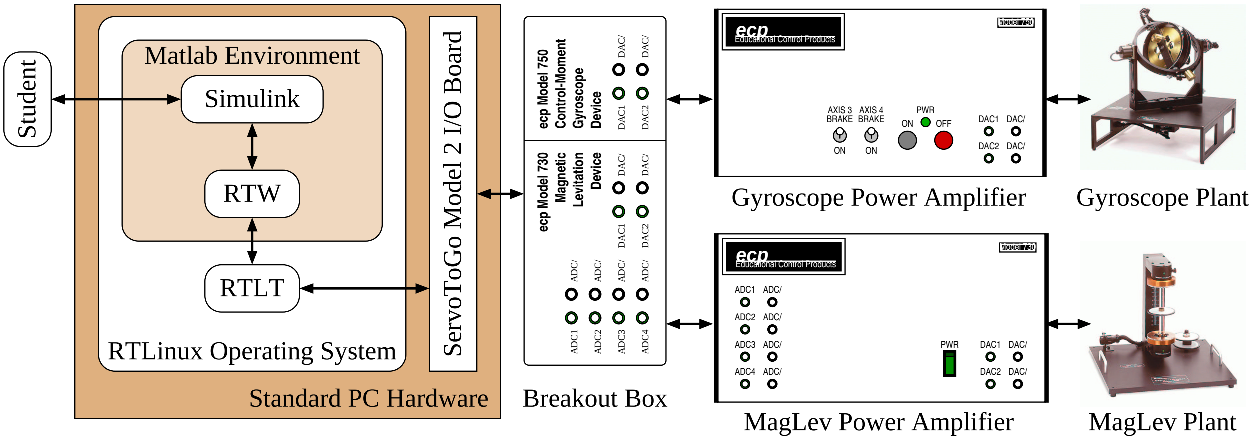


Figure 4. Pictorial representation of system setup including hardware and software integration.

The final lab setup comprises the following hardware for each workstation: One *ECP* MagLev “plant-only” unit, one Pentium-III class computer with a *ServoToGo* Model 2 I/O board, one *Comdyna* GP-6 analog computer, one *Protek* 3003B dc power supply, one *Protek* B-803 sweep function generator and one each *Agilent* HP-3468B multimeter HP-54602B digital oscilloscope (with HP-54657A unit for phase measurement). The test-and-measurement equipment are used for system identification and control-system debugging.

The following software is used on each lab computer: Red-Hat Linux 6.1 (free download available from `http://www.redhat.com`), RT-Linux 2.0 (free download available from `http://www.rtlinux.org`); the following *MathWorks* products: Matlab 5.3, Release 11, for Linux, Simulink 3.1, Release 11, for Linux, Real-Time Workshop 3.0, Release 11, for Linux; and the *QRTS* product RTLT Version 1.1. The C development packages should be selected when installing Red Hat Linux Version 6.1.

VI. Course Organization

The undergraduate senior elective *Feedback Control Systems* is a standard lecture-based course covering control systems from a frequency-domain point-of-view. A textbook by Franklin and colleagues is used.[18] The *Control Systems Laboratory* course meets in the laboratory described in this paper, and has the lecture course as a co-requisite. The two courses are designed to coordinate with each other as much as possible so that the experimentation compliments and illuminates the theory. A Gantt chart showing the relative phasing of the two different courses is shown in Figure 5.

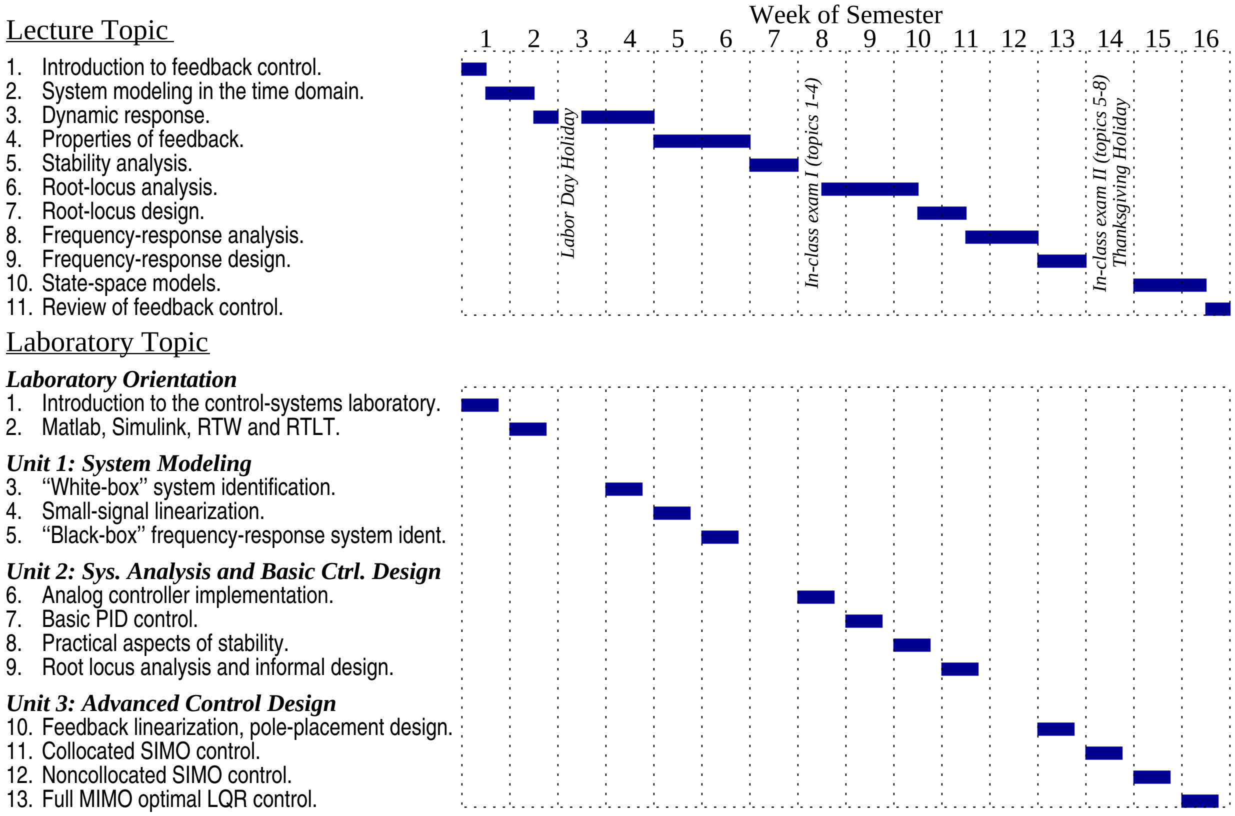


Figure 5. Gantt chart illustrating synchronization between lecture topics and lab topics. Lectures are on Monday/Wednesday, and labs are on Tuesday.

The laboratory course is organized into two introductory labs and three main units of labs. The introductory labs introduce the students to the laboratory facility, instruct them on how to log on to and use the lab computers, and prescribe proper care of the equipment. The students are given a simple assignment in Matlab/ Simulink/ RTW/ RTLT. In their junior year the are exposed to Matlab in their curriculum, but Simulink may be unfamiliar to them, and they will not have seen RTW or RTLT before.

The first unit following the introductory labs has the students make a mathematical model of the MagLev device. (The lecture course has already covered, by the appropriate time, system modeling in the time domain and dynamic response from the frequency-domain perspective.) In the first lab, they measure data to characterize the static nonlinearities of the actuator and sensor, and fit curves to data in order to generalize the results. In the second lab, the students perform small-signal linearization to make linear models of the MagLev in various operating configurations. In the third lab, they use frequency-response methods to verify and tune their time-domain model.

The second unit has the student explore various aspects relating to feedback. (The lecture course has already covered, by the appropriate time, basic properties of feedback, stability using the Routh test, and root-locus plotting.) In the first lab they implement a given controller on the GP-6 analog computer. This lab is designed to demonstrate that an expensive digital controller is not required—op amps suffice to implement a linear control system. In the second lab they use a Ziegler-Nichols[18] method to design and implement PID control. The students measure transient and steady-state response to step inputs, investigate disturbance rejection and robustness (they implement their controller on several MagLev units and compare response). In the third lab they explore stability by controlling a magnet in the upper (unstable) position. In the fourth lab, they use root-locus concepts in an informal way to design a lead controller for the magnet in upper and lower positions.

The third unit covers advanced control-design topics. (The lecture course has already covered, by the appropriate time, root-locus design, frequency-response analysis and design and state-space models.) In the first lab, the students use feedback linearization and root-locus pole placement to control a single magnet in the upper and lower positions. In the second and following labs, the MagLev is configured with both disks on the glass rod (which adds an additional resonant mode). In the second lab, collocated control is investigated as the students control the lower disk position using the lower actuator and lower sensor as primary feedback. They use root-locus pole placement design. In the third lab, the students perform noncollocated control using frequency-response based design of lead/ lag and notch filters. They control the upper disk position using the lower actuator as input and the upper sensor as primary feedback. The approach is to close a high-bandwidth inner loop around the lower disk position, and then an outer feedback loop around the upper disk position. In the fourth and final lab, the students compare dual SISO design (using their controllers from the first lab of the unit) with LQR MIMO design. This lab provides a climax to the course in which students design their first multivariable controller.

A lab reader has been prepared for this class, and is available on the Internet.[19] If you would like to use lab(s) from this reader, please contact the first author at: `glp@eas.uccs.edu`.

VII. Initial Evaluation and Student Feedback

Although the lab has been operational for only one semester, we have begun evaluating its effectiveness using quantitative and qualitative means. We wanted to test two hypotheses:

Hypothesis: A lab experience provides hands-on learning to improve basic understanding. Therefore, students taking the lecture course only (the lab is not required) will do more poorly in the lecture course than those students taking the lab as well.

Results: We have data for three semesters for students taking the lecture course only versus students taking both the lecture and lab. The lab syllabus for the first two semesters was based on simulation only; the lab for the final semester is the one based on experiment and presented in this paper.

Over the three semesters, grades in the lecture course for students taking both the lab and the lecture averaged to 78%; grades in the lecture course for students not taking the lab averaged to 69%. There is a 9% gap between these two groups of students. There are enough students in the sample to make this result statistically meaningful.

Hypothesis: An experimental lab provides better learning than a lab based on simulation.

Results: For the first two semesters, the gap between lecture-only students and students who took both the lecture and lab was a 5% difference in course grade in the lecture course (both times). For the semester where students had an experimental lab experience, the gap was 16%. This result is interesting but probably not statistically meaningful since only two students elected *not* to take the lab course this last semester.

Overall, the quantitative results are in favor of a lab experience whether or not it is experimental. The results seem to be in greater favor of an experimental approach versus a simulation approach, but the sample size of statistics is too small to tell for sure as of yet.

Qualitative evaluation has been done by recording some responses from student lab writeups. As you will be able to tell, they are quite candid:

> *[On analog computer lab]* “This lab was boring because we did not get to play with the MagLev.”
> *[On analog computer lab]* “This was a painful experience, I hate the analog computer with a passion, and it hates me.”[1]
> *[On system identification]* “These methods are pretty cool in that I never realized that they took into account friction, which makes them more useful.”
> *[On linearization, comparing linearized model with real system]* “Just for fun, we inputted some other waves: square wave, triangle wave, ramp function and so on. The coolest one was when we put in a random signal, and raised the value for u_{mg} [the dc-offset]. The magnet would jump around like a crazy grasshopper, and the model output matched the actual output almost exactly.”
> *[On frequency-response/ Nyquist]* “This lab clears up a lot of concepts learned in class. Like the Nyquist plot, I didn’t realize that the Nyquist plot was just a polar plot until this lab (although I am sure that Dr. Plett probably mentioned it a million times).”
> *[On the course in general]* “To be quite honest, I’m learning more from this lab than previous labs taken and labs that I’m taking right now. I am also impressed how much it actually follows ECE4510 Feedback Control Systems and uses the things that we learned in class.”

Some of these comments indicate that real learning has occurred. One shows that the students were so involved in the lab that they tried some experiments which were not required. The final comment validates that the lecture and lab courses are well coordinated.

---

[1] The analog computer lab which the students did not like has been modified in the lab reader now available. It now includes experimentation with the MagLev, and now has a clearer presentation on how to use the analog computer to perform feedback control.

VIII. Conclusion and Future Plans

A control-systems laboratory has been developed around the *ECP* MagLev and Gyro plants, using Matlab/ Simulink/ RTW and RTLT software, on the RTLinux operating system. As of the writing of this paper, the lab has been in use for one semester. A lab manual has been developed for the undergraduate *Control Systems Laboratory* which uses the MagLev device. The lab schedule coordinates with the *Feedback Control Systems* senior-elective class so that experiments complement and illuminate the theory. The MagLev has also been used for the final project in the graduate *Multivariable Control Systems I* class. Initial evaluation indicates that the laboratory experience has significantly aided learning of control-systems concepts.

Future plans include the development of a *Digital Control Systems Laboratory* course, using the Gyroscope unit in an Aerospace *Digital Flight Control* course and in advanced graduate courses. We also hope to integrate the MAE and ECE undergraduate control-systems courses for a full multidisciplinary experience. We believe that the intermingled perspectives of two disciplines will lead to better-rounded learning.

Acknowledgement

This laboratory was made possible and this work was supported in part by the National Science Foundation under grant DUE–981009. Also special thanks to Ali Pak from *ECP*, and both Nick Costescu and Darren Dawson from *QRTS* for helping to design an affordable system; for email and telephone support from both. Thanks to both *ECP* and *QRTS* for allowing use of text and diagrams from their manuals in the *Control Systems Laboratory* lab reader.

Bibliography

1. Comdyna, Inc., 305 Devonshire Road, Barrington, IL 60010, USA. Tel/Fax: (847) 381–7560, `http://www.comdyna.com`.
2. The MathWorks, Inc., 24 Prime Park Way, Natick, MA 01760 USA. Tel: (508) 647–7000, Fax: (508) 647–7001, `http://www.mathworks.com`.
3. D. S. Bernstein. Control experiments and what I learned from them: A personal journey. *IEEE Control Systems Magazine*, 18(2):81–88, April 1998.
4. L. Schmitt. U of I's control systems lab one of premier labs in country. *University of Illinois at Urbana-Champaign "Ingenuity"*, 4(1):4, March 1999.
5. Educational Control Products, 1 Buckskin Court, Bell Canyon, CA 91307, USA. Tel: (800) 486–0840, Fax: (818) 703–6040, `http://www.ecpsystems.com`.
6. Z. Yao, N. P. Costescu, S. P. Nagarkatti, and D. M. Dawson. Real-Time Linux Target: A Matlab-based graphical control environment. In *Proceedings of the IEEE International Conference of Control Applications*, (Anchorage, AK: September 2000).
7. Quality Real-Time Systems, 6312 Seven Corners Center #134, Falls Church, VA 22044 USA. Tel: (703) 533–9142, Fax: (703) 533–9143, `http://www.qrts.com`.
8. HUMUSOFT s.r.o., Novákových 6, 180 00 Praha 8, Czech Republic. Tel: ++420-2-66315767, Fax: ++420-2-6844174, `http://www.humusoft.com`.
9. Quanser Consulting, 102 George Street, Hamilton, ON L8P 1E2, Canada. Tel: (905) 527–5208, Fax: (905) 570–1906, `http://www.quanser.com`.
10. Servo To Go, Inc, 1716 221st Place Northeast, Redmond, WA 98053, USA. Tel: (425) 868–4636, Fax: (425) 868–5323, `http://www.servotogo.com`.
11. M. Timmerman and J.-C. Monfret. Windows NT as a real-time OS? *Real-Time Magazine*, 2Q:6–13, 1997.

12. W. E. Dixon, D. M. Dawson, B. T. Costic, and M. S. de Queiroz. Towards the standardization of a MATLAB-based control systems laboratory experience for undergraduate students. In *Proceedings of the American Control Conference*, (Arlington, VA: June 2001). (Submitted. Available at `http://64.0.106.39/downloads/publications.shtml`).
13. N. Costescu, D. Dawson, and M. Loffler. QMotor 2.0—A real-time PC based control environment. *IEEE Control Systems Magazine*, 19(3):68–76, June 1999.
14. The official Linux Web site. `http://www.linux.org`.
15. The official RTLinux Web site. `http://www.rtlinux.org`.
16. V. Yodaiken and M. Barabanov. Real-Time Linux. *Linux Journal*, February 1997.
17. Y.-C. Chen and Jason M. Naughton. An undergraduate laboratory platform for control system design, simulation, and implementation. *IEEE Control Systems Magazine*, 20(3):12–20, June 2000.
18. G. F. Franklin, J. D. Powell, and A. Emami-Naeini. *Feedback Control of Dynamic Systems*. Addison-Wesley, Reading, MA, third edition, 1994.
19. G. L. Plett. *ECE4530: Control-Systems Laboratory* lab reader, August 2001. Available at `http://mocha-java.uccs.edu/CSL/`.

GREGORY L. PLETT

Gregory Plett was born in Ottawa Canada in 1968. He received the B.Eng. degree (with high distinction) in Computer Systems Engineering from Carleton University in 1990. He received the M.S. and Ph.D. degrees in Electrical Engineering from Stanford University in 1992 and 1998. Since 1998, he has been an Assistant Professor in the Department of Electrical and Computer Engineering at the University of Colorado at Colorado Springs. He can be reached by email at `glp@eas.uccs.edu`.

DAVID K. SCHMIDT

David Schmidt was born in Lafayette Indiana in 1943. He received his B.S. (with honors) and Ph.D. in Aerospace Engineering from Purdue University, and the M.S. in Aerospace Engineering from the University of So. California. He has served on the technical staffs of McDonnell Douglas Missiles & Space Corp, and the Stanford Research Institute. He has also served on the engineering faculties of Purdue University, Arizona State University, the University of Maryland at College Park, and now is Professor of Engineering at the University of Colorado at Colorado Springs. He can be reached at `dschmidt@eas.uccs.edu`.