122nd ASEE Annual Conference & Exposition
June 14 - 17, 2015
Seattle, WA
Making Value for Society

Paper ID #11184

From ’system modeling’ to ’controller hardware testing’ in three hours: a robotic arm controller design lab using MATLAB Real Time Windows Target to reinforce classical control theory

Dr. Charles Birdsong, California Polytechnic State University

Charles Birdsong has expertise in automotive safety, vibrations, controls, signal processing, instrumentation, real-time control, active noise control, and dynamic system modeling. He received his B.S.M.E. at Cal Poly San Luis Obispo, and his M.S. and Ph.D. at Michigan State University, where he worked on active noise control applications for the automotive industry. He has worked in the vibration test and measurement industry helping to drive new technologies to market and working with industry to meet their emerging needs. He is currently a Professor at California Polytechnic State University at San Luis Obispo in the Department of Mechanical Engineering teaching dynamics, vibrations and controls. He is involved in several undergraduate and master’s level multidisciplinary projects and interested in engineering education research.

From 'system modeling' to 'controller hardware testing' in three hours: a robotic arm controller design lab using MATLAB Real Time Windows Target to reinforce classical control theory

Abstract

A 3 hour hands-on lab experience was designed to give students the opportunity to model, design and test a controller for a nonlinear electro-mechanical robot-arm system as a culminating project in mechanical engineering controls course. Students used linearization to model a nonlinear pendulum DC servo system using first principles. They used experimental frequency response data to derive a numerical transfer function model. Next they used analytical techniques including root locus and Simulink modeling to design a PID controller. Finally they tested their controller design using a hardware-in-the-loop system with MATLAB's real-time windows target system to assess the performance of their controller. The entire process was started and completed in one three hour lab period. The goals of the exercise were to give students the chance to complete an entire control system design cycle from modeling to hardware testing in one sitting, incorporate as many of the course concepts as possible and give the students a practical understanding of the application of the theory. Assessment was conducted using pre and post online quizzes testing conceptual understanding of the major topics such as linearization, frequency response, and the effect of proportional, integral and derivative control. The assessment indicate a significant improvement in understanding of the theory and positive attitudes regarding the experience.

Introduction

Our university's philosophy is that students learn best through a combination of lecture and lab experiences and industry feedback indicates that our undergraduates are unparalleled at hitting the ground running and working with real world problems. While many have reported on the effectiveness of including hands-on laboratory exercises to enhance learning [1], these labs are expensive and there is pressure to eliminate them in the times of increasing budget pressure. This study suggests that the cost is justified because learning outcomes are significantly improved compared to a lecture only course.

Mechanical Controls is a four unit, required senior level course that consists of three-one hour lectures and one-three hour lab per week for the ten week quarter. The course covers single-input single-output linear system modeling, time domain analysis, transfer functions, root locus, frequency response methods, PID and lead lag controllers. The lab is taken concurrently with the lecture and is designed to support the topics covered in lecture while also illustrating the realities of real world systems, modeling and controller implementation. There are currently four exercises: a two week Matlab/Simulink simulation exercise, a two week analog DC servo position control experiment, a two week digital controller tank water level regulator experiment and a two week digital controller hydraulic servo control experiment.

The digital control experiments use Matlab Real Time Windows Target (RTWT) with a National Instruments data acquisition board (DAQ) interface to implement a simple PID controller in real-time. This software interface is ideal for the learning environment and is use in many universities because the Simulink code is displayed as a block diagram that is identical to the theory that is presented in the textbook [2] [3] [4]. This makes modifying and developing new controllers simple for the student with minimal programming experience. This hardware was introduced in this lab in 2008 in an experiment using a hydraulic control system [5].

This paper describes the DC servo experiment that is performed early in the term and also the addition of a new one-week PID Design Project that is intended to include elements from many of the lecture topics and end the term with a single creative and self-contained experiment. Assessment of the new project was conducted using a pre and post quiz and surveys.

Linear DC Servo Setup

The new PID Design project is conducted in the last week of the term. However, students test and analyze the DC servo at the beginning of the quarter with only an inertia disk and gear system as the mechanical load. The experimental apparatus shown in Figure 1 is a DC servo control system based on the Motomatic Inc. hardware. The angular position is measured by an angular potentiometer sensor. The sensor signal is measured by a DAQ interface to a Matlab/Simulink RTWT PID controller. The controller generates an actuator signal via the DAQ analog output channel which is connected to the motor amplifier to close the loop.

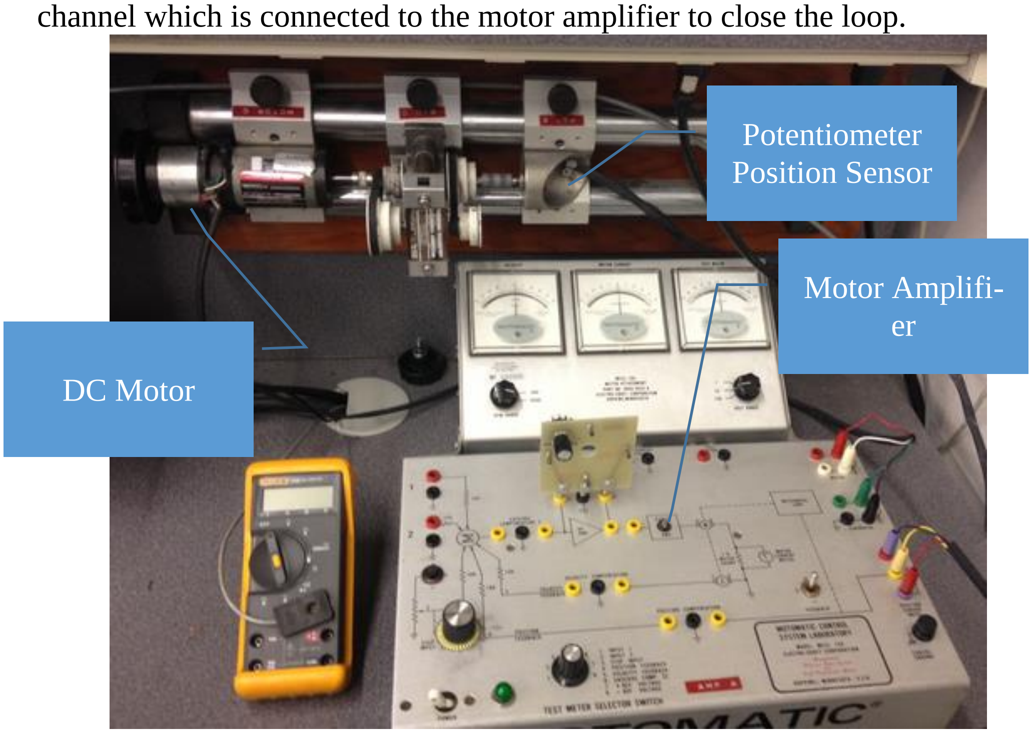


Figure 1. Photograph of DC servo control apparatus showing DC motor, potentiometer position sensor and motor amplifier.

The DAQ communicates with the MATLAB/Simulink RTWT software [6], [7]. The software is a special block-set within the Simulink software that allows DAQ inputs and outputs to be used as sources and sinks in a Simulink model. Before the model can be run in external real-time mode it must be compiled into C code. This process is initiated by clicking on a button and is then handled automatically by MATLAB. The student does not need to have any special programming skills beyond Simulink programming.

Data can be logged and exported to the MATLAB workspace for controller evaluation. The sampling rate of the system depends on the complexity of the model and the speed of the PC processor. The PC processor is used to perform the real time calculations while the RTWT software manages the Windows operating system and guarantees the control process receives the maximum processor resources. Using a Pentium 1.9 GHz PC the PID Positioner model can be run as fast as 10 kHz however the system was limited to 1 kHz which is adequate for the servo control experiment. Faster sampling rates could probably be attained for other applications with faster PC processors.

The DC servo motor model is simply a first order transfer function and produces acceptable agreement with experimental results. A significant amount of static friction in the motor and gears results in a nonlinear step response, where the shaft overshoots only once then sticks at a final position instead of oscillating and decaying gradually as a linear system. A nonlinear model can be created by adding a dead-zone function between the power amplifier and the DC servo blocks to introduce the static friction effect. Figure 2 shows a block diagram of the system in closed-loop feedback.

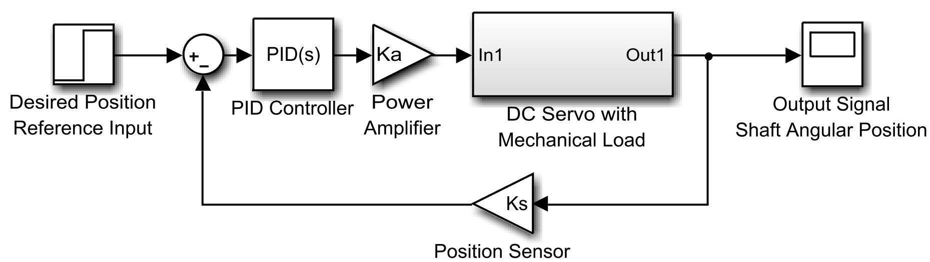


Figure 2. Block diagram of DC Servo position control system.

Figure 3 shows a typical comparison of closed-loop step response of the model and experimental results from a typical student report. The experimental response is the solid line, the non-linear model including the dead-zone block is the thin solid line and the tuned non-linear model is the dashed line. As usual, the model does not agree with the experiment perfectly because of experimental error in approximating the system parameters plus an inexact model of the nonlinearities of the system. The model and can be improved by tuning the system parameters. The tuned model is the result of modifying the system parameters in an ad-hoc fashion to match the experimental results, such as peak time, steady-state value, etc., as well as possible.

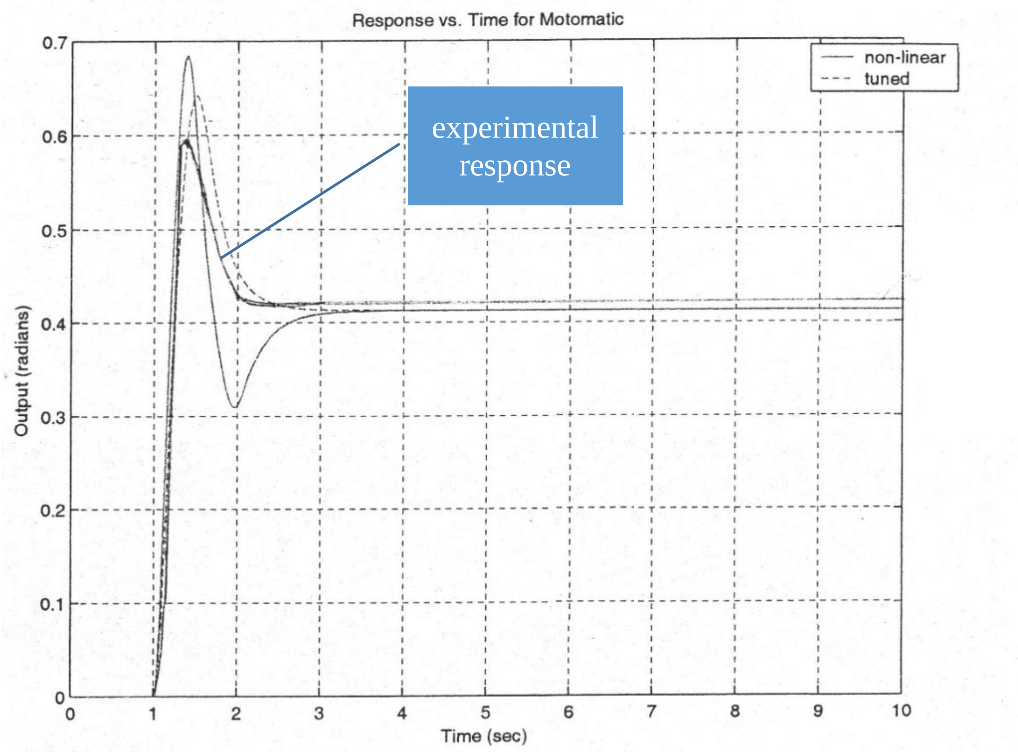


Figure 3. Comparison of closed-loop model (thin solid line), tuned model (dashed line) and experimental results (thick solid line) of closed-loop proportional controller with DC servo control and step input.

This experiment is conducted early in the quarter term at a time when the students may not have enough theory from the lecture to be able to design a PID controller based on design specifications. Although the model can be used to try PID gains in a trial and error fashion to determine acceptable controller gains. Students are asked to vary the controller gains and observe the qualitative effects of the proportional gain on response measures such as settling time, percent overshoot, damping, and steady state error [8]. The lab is very successful and gives the students an opportunity to get a hands-on understanding of the effects of PID controller gains. Because the lab is conducted before the theory for controller design is complete, it helps motivate the concept of controller design as an improvement over the ad hoc method of tuning the controller used in the lab.

New PID Controller Design Project

Near the end of the term, after 4 weeks working on two other systems, students returned to the DC Servo for the new PID Design Lab. The DC servo system was modified by the addition of a cantilever beam with a mass at the end fastened to the shaft. The arm creates a moment that depends on the angular position of the shaft. This results in a nonlinear model where the step response is different at each steady state angular position. The challenge for the students is to predict the system model based on a description of the system, and to use the model to design the controller. While the nonlinear system has many possible advanced control schemes [9] [10] [11], for this introductory course, students were limited to linearization and PID control about a steady-state operating point which is covered in the lecture. This can be done by linearizing the nonlinear equations of motion then using classical root-locus techniques.

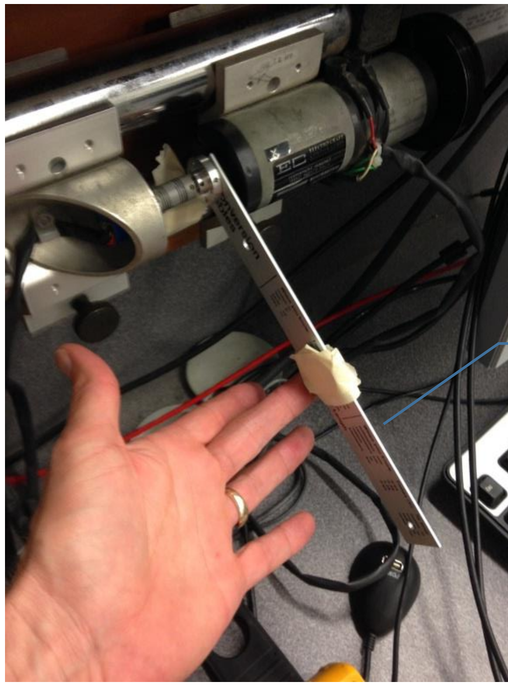


Figure 4. Photograph of DC motor with cantilever beam attached to the shaft to create a nonlinear system.

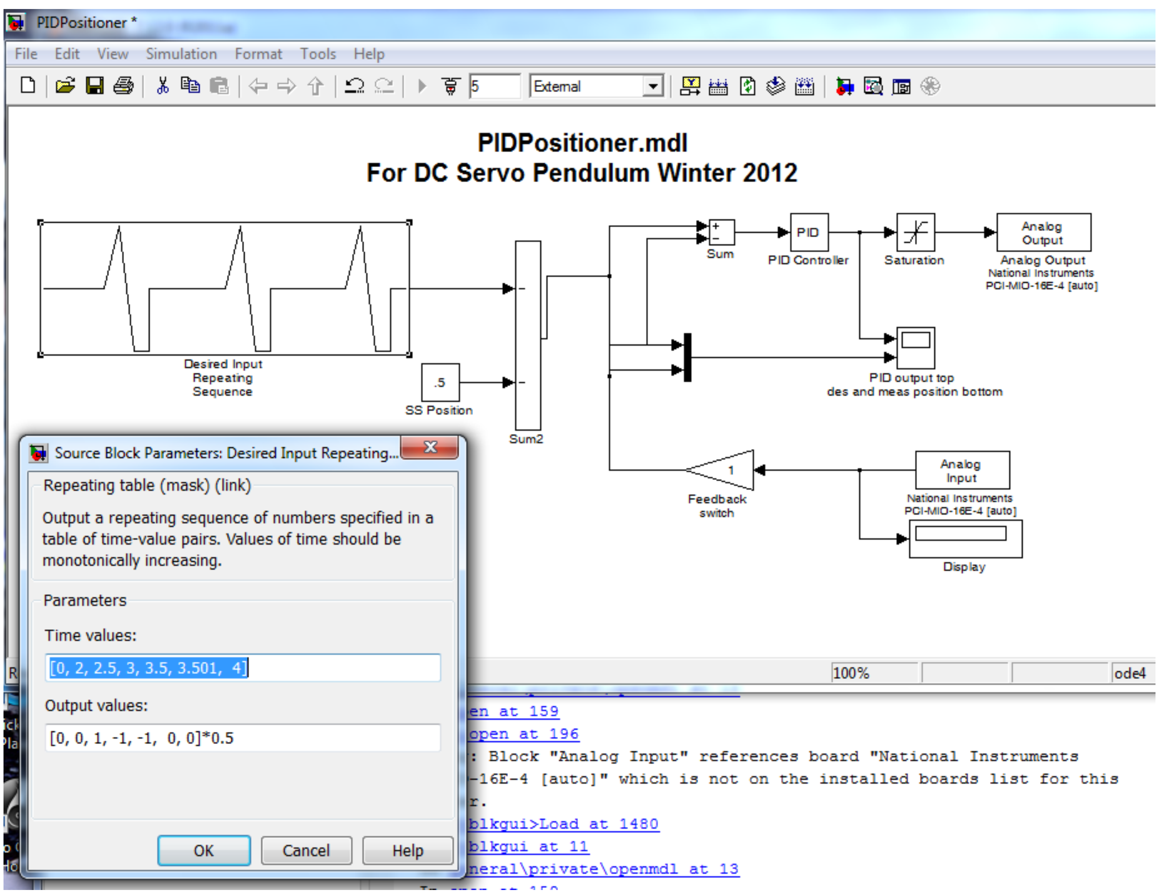


Figure 5. Closed-Loop PID Controller Interface in Matlab/Simulink RTWT

Figure 5 shows the closed-loop PID controller model that is used to implement closed-loop position control of the system. This interface has the advantage that it looks like the closed-loop feedback block diagram that the students study in the lecture. This helps the students understand and visualize the program very quickly. A special pulse reference function was used to force the students to consider several different effects. The desired input repeating sequence is a combination of ramp up, ramp down, dwell and return to start. A constant is added by the “SS Position” block to move the pendulum arm from the neutral position to a steady-state position before the repeating sequence is started. This results in a steady state angle of about 20 degrees from the bottom, though this value can be changed to any value desired.

Students predict the linearlized transfer function model based on a description of the system as a homework assignment. Then during lab, rather than perform system identification, students were shown a demonstration of an open-loop swept sine frequency response test using an LDS-Dactron Focus real-time signal analyzer and given the experimentally measured frequency response (bode) data. They were asked to use the bode data to estimate the system transfer function, and then use the transfer function to perform analytical PID controller design. To emphasize the importance of modeling and design versus trial and error controller tuning, they were then given only two chances to test their design with the real system. Students filled out a form with the controller transfer function and PID gains and submitted it to the instructor who entered the gains on the test system and ran a test.

Figure 6 shows the results of a typical PID controller performance evaluation. The top graph shows the controller output in volts. The middle plot shows the desired reference signal and the measured position under the closed-loop PID control. The ramp from 2 to 3 seconds was included to force the students to consider adding integral control to reduce the steady state error in this region. The bottom plot shows the error function which was computed from the absolute value of the difference between the reference and measured signals and summed for each data point. The sum of the error values was displayed as the “Error Function” value and used to measure the performance of the PID controller design. The Error Function value of 83 shown in the figure was among the best performing controllers in the class. The evaluation and the figure was automatically generated after the test by implementing an m-file in the Simulink callback function. Students were given the plot and could use the data to refine their design for a second test.

A significant amount of electrical noise was present from the potentiometer position sensor, which can be seen in the error signal. The overall Error Function is affected by this noise. Furthermore, the high noise level limits the use of derivative control since derivative control amplifies the high frequency noise signal and corrupts the actuator signal. A higher quality sensor with less noise would be a good improvement in the system and is under consideration for the future. On the other hand, the noise forces the students to face the common problem with noise and derivative control.

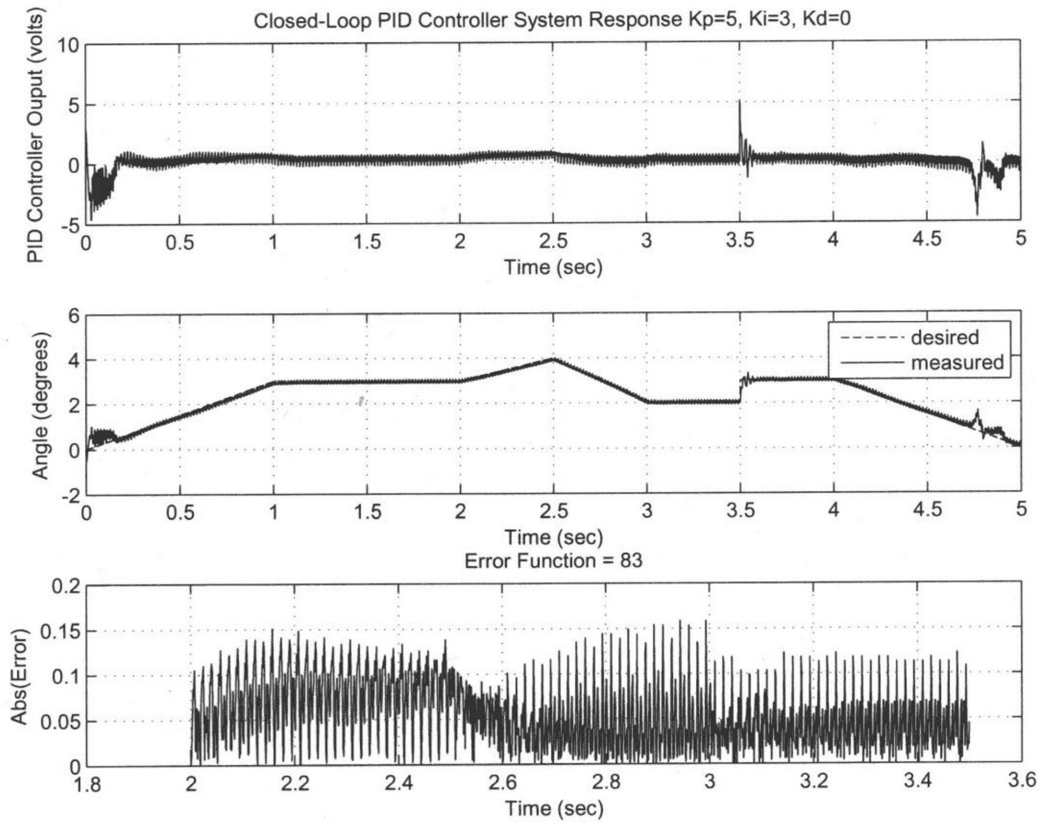


Figure 6. PID controller design evaluation results with controller output (top), measured and reference signal (middle) and error function (bottom).

The figures below show some other student results. Figure 7 shows a controller that produced an acceptable response, though the Error Function value is worse than the previous example due to more steady-state error during the ramp; a higher integral gain would improve the performance. Figure 8 shows a controller that produces poor response where the output does not track the input well; the proportional gain was too low. Figure 9 shows a controller with acceptable response until the step change at 3.5 seconds after which the nonlinearity of the system causes instability; the proportional gain was too high in this case.

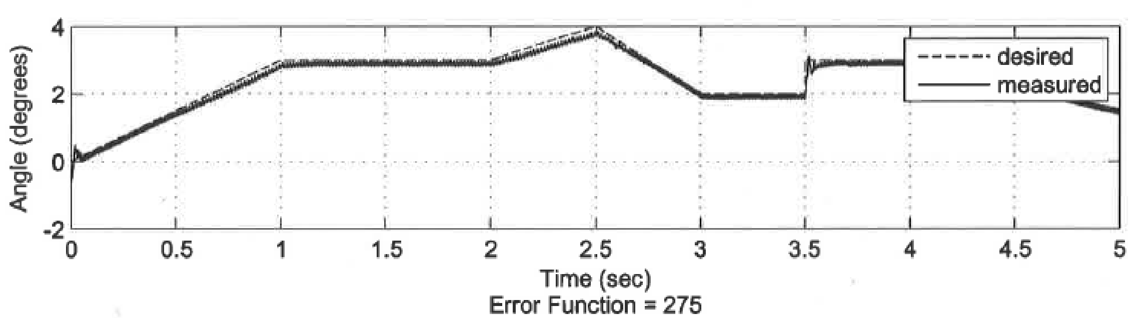


Figure 7. PID feedback control response with acceptable results.

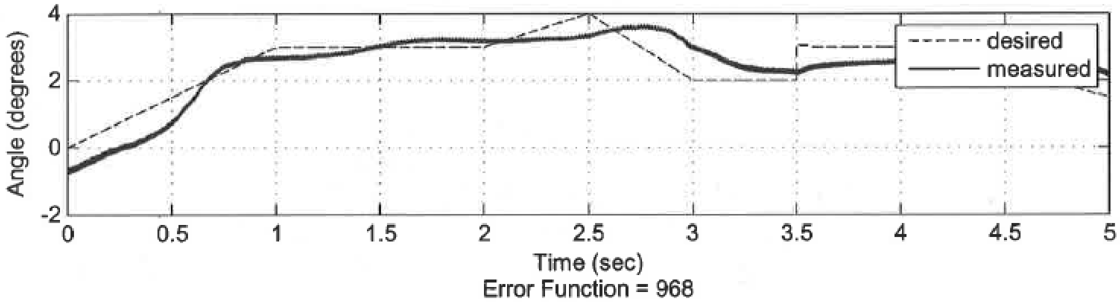


Figure 8. PID feedback control response with poor response.

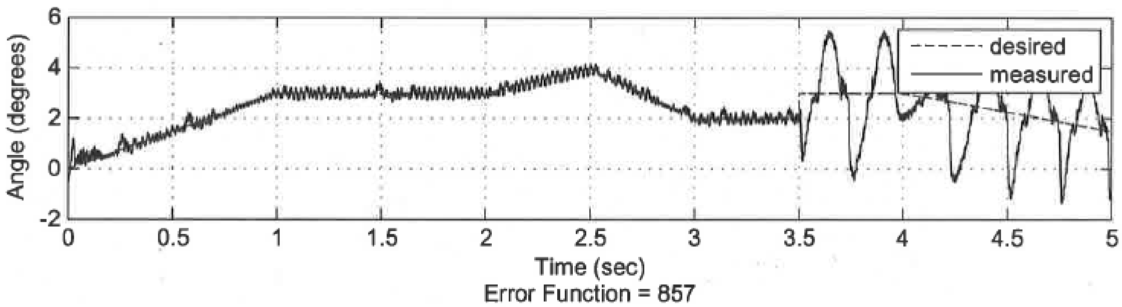


Figure 9. PID feedback control with good response until the step change; after which the nonlinearity results in instability.

Students were asked to document their design methods and turn in a report showing the design steps and any supporting calculations used in the process. Students were encouraged to model the closed-loop response with their PID control design before trying their controller design on the real system. Students were told that the report grade would be based mostly on documentation of the design procedure, not the performance of the controller, though, the top three designs would receive extra credit. This seemed to help motivate the students in a somewhat competitive environment and make the three hour project more fun.

The design project will be used in the future and has the benefit that the system can be modified to change the dynamics by modifying the mass and arm length and steady-state angle. Also different reference functions can be used to change the best controller design results. Hopefully this will eliminate the possibility that students will get results from previous years and get a short cut to a good design. The challenge is in formulating the design constraints in a way that allows classical linear control theory and PID controller design methods to result in good solutions while at the same time illustrating the real world effects such as saturation, noise and other nonlinear phenomena.

Implementation Issues

While this exercise was conducted using the DC servo hardware that is already installed in the controls lab, it should be possible for others to reproduce a similar exercise with a limited budget. This exercise can be conducted with a single experimental setup to minimize the cost of multiple stations: students use analytical methods to design a controller and then the controller is tested on one unit by the instructor. The hardware consists of a typical DC motor, a potentiometer position sensor, and a power amplifier. Data acquisition hardware that is compatible with the Matlab RTWT is available below $400. The most expensive item is the Matlab/Simulink with the Real Time Windows Target license, though many universities purchase a site license that includes all the required software. Matlab is the key to the experiment because it allows the instructor to input controller gains and run the system in few seconds so that many students can test their designs in a short amount of time. On the other hand, it would be possible to implement a real-time PID controller using a low cost microcontroller such as the Arduino [12] and achieve the same results without the Matlab license. Hopefully the reader agrees that giving the student the opportunity to complete the entire system modeling, controller design and testing experience in one three hour setting is worth investing in the hardware and software.

Assessment

The PID Design Lab exercise was assessed to measure how it improved students understanding of the main concepts and their own confidence in the concepts embedded in the experiment. It should be noted that all of the concepts that were assessed had been covered in the lecture portion of the course in the preceding weeks. Also students were told to complete the homework assignment that involved deriving equations of motion, linearizing the nonlinear terms and deriving a linearized transfer function model from a description of the system before the assessment. Therefore, the students should have had significant knowledge from the course and the homework assignment to answer the questions. A pre and post assessment was given using an online quiz and survey instrument. Eight technical questions were given in an online quiz the night before the experiment asking students to predict how the system will respond to different PID controller configurations. The next day the students conducted the experiment in the lab. They were then told to complete the post assessment online within twenty-four hours so that as little time as possible and other confounding events would not affect their responses. In the post assessment, they were asked the exact same technical questions to determine if their understanding had improved. Figure 10 illustrates the results. The results only include students that completed both pre and post assessments. The students were told they would get full credit worth one homework assignment for completing the assessments regardless of their performance.

In most cases the students performed much better after the lab experience as indicated by the significant positive change between pre and post. The text of the eight questions is presented in the appendix.

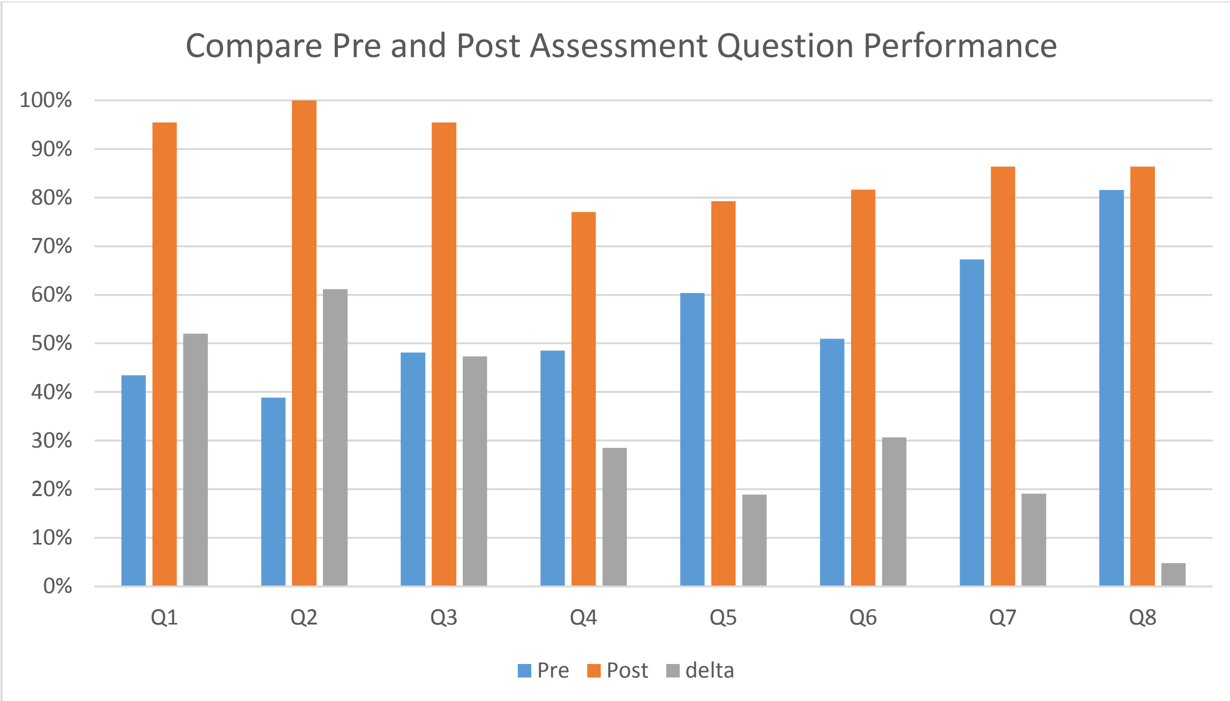
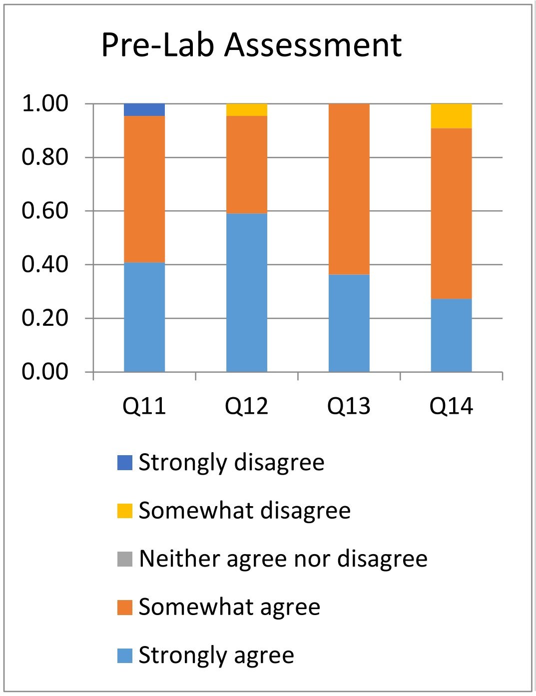


Figure 10. Pre and post assessment showing percent of students selecting the correct answer to eight technical questions, and the change (delta), n=22.

In addition, students were asked to rate their own understanding of four main concepts (Questions 11-14 are listed below): linearization, proportional, integral and derivative control gains, before and after the experiment. Students answered using a Likert scale [12] [13]. Student responses changed significantly towards indicating a better understanding after the exercise.

> Q11: I understand the difference between a linear and a non-linear system and how to model both.
>
> Q12: I understand the effects that a proportional controller has on a closed-loop feedback system
>
> Q13: I understand the effects that an integral controller has on a closed-loop feedback system
>
> Q14: I understand the effects that a derivative controller has on a closed-loop feedback system.



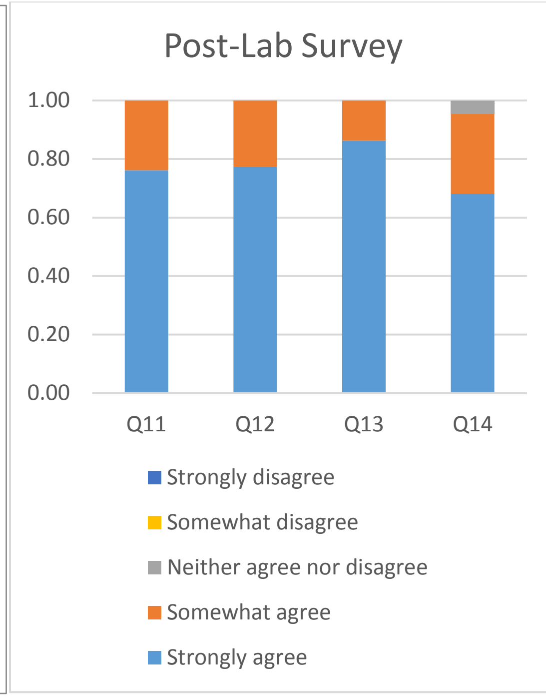


Figure 11. Comparison of pre and post survey questions 11 through 14, n=22.

Table 1. Pre-Lab Assessment Results

	Q11	Q12	Q13	Q14
Strongly agree	9	13	8	6
Somewhat agree	12	8	14	14
Neither agree nor disagree	0	0	0	0
Somewhat disagree	0	1	0	2
Strongly disagree	1	0	0	0

Table 2. Post-Lab Assessment Results

	Q11	Q12	Q13	Q14
Strongly agree	16	17	19	15
Somewhat agree	5	5	3	6
Neither agree nor disagree	0	0	0	1
Somewhat disagree	0	0	0	0
Strongly disagree	0	0	0	0

Finally, after the experiment, students were asked to compare the lab experience with solving textbook problems with their response to the following, "Q15: The PID Design Lab helped me understand the controller design process more than solving textbook homework problems." (Post-lab survey only).

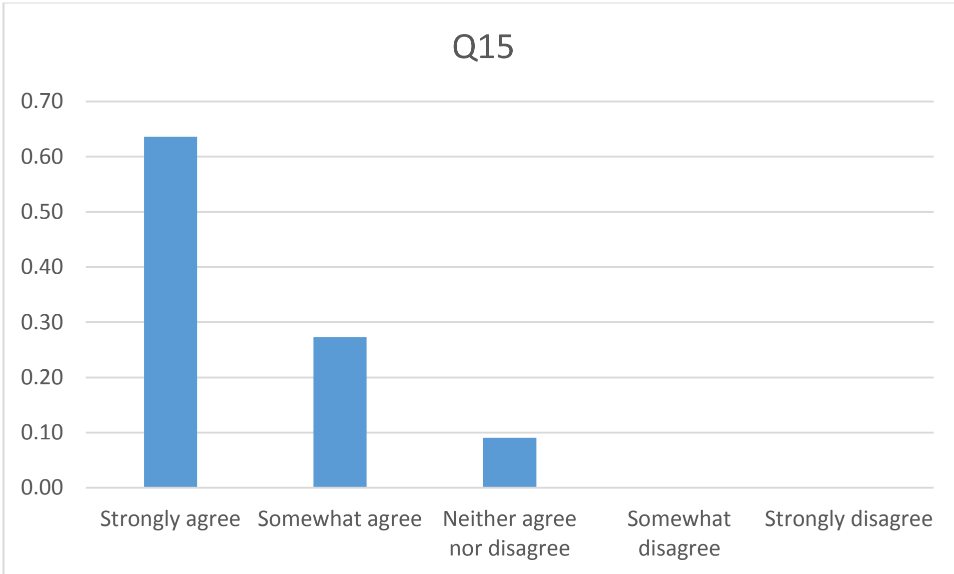


Figure 12. Response to Question 15: "The PID Design Lab helped me understand the controller design process more than solving textbook homework problems" (Post-Lab Survey Only), n=22

The assessment results show that even though the concepts had been covered in the lecture portion of the class in previous weeks, student understanding was relatively low before the lab. After the 3 hour exercise understanding was significantly improved. If the course did not have the practical lab component it seems that students would be left with the weaker understanding of the concepts illustrated in the pre-lab assessment. Students also strongly agreed that the lab helped them to understand the controller design process more than solving textbook problems.

Conclusions

The MATLAB/Simulink RTWT software was introduced into the Mechanical Engineering Controls lab at Cal Poly and applied in PID Controller Design project. The software is easy to program and easy for the students to understand in the context of linear control theory. In addition the ability to modify the controller with very little effort enabled a PID design project to be added to the curriculum. The project forces students to apply many of the theoretical topics covered in the lecture in a single three hour lab setting. It forces students to apply analytical tools to design a controller and then to immediately see the actual performance of the physical system. Assessment shows that the new project improved student understanding and self confidence in the concepts of linearization of nonlinear systems and controller design. Furthermore the results suggest that the hands-on experience had a larger influence than lecture and solving textbook problems on student understanding and self-confidence. This exercise also has the advantage that it can be repeated at other universities with a single hardware setup, thus minimizing the total cost of maintaining multiple hardware stations.

Bibliography

[1] L. Chen and C. Lin, "Combining Theory with Practice in Information Security Education," in *Proceedings of the 11th Colloquium for Information Systems Security Education*, Boston University, 2007.

[2] A. Johnson and J. Rubaai, "DSP-Based Real-Time Control Systems Design, Analysis, and Implementation for Reinforement of Controls Education," in *ASEE Conference*, 2006.

[3] A. Mohammadzadeh and S. Haidar, "Analysis and Design of Vehicle Suspension System Using MATLAB and Simulink," in *ASEE*, 2006.

[4] R. Throne, "Developing Models and an Understanding of Their Limitations," in *ASEE*, 2006.

[5] C. Birdsong, "Developing A Matlab/Simulink Rtwt Based Hydraulic Servo Control Design Experiment," in *ASEE Annual Conference & Exposition*, Honolulu, 2007.

[6] MathWorks, MATLAB, version 7, Natic, MA: Math Works, Inc., 2005.

[7] MathWorks, Real-time Windows Target User's Guide Version 2, Natick, MA: Math Works, Inc., 2004.

[8] N. Nise, Control Systems Engineering, Wiley & Sons, 2004.

[9] F. Reys and R. Kelly, "Experimental evaluation of identification schemes on a direct drive robot," *Robotica, 15 (1997)*, p. pp. 563–571, 1997.

[10] R. Fernando and R. Kelly, "Experimental evaluation of model-based controllers on a direct-drive robot arm," *Mechatronics, Volume 11, Issue 3*, pp. Pages 267-282, April 2001.

[11] A. Nikranjbar, "Adaptive PID controller design guidelines for a class of non-linear systems," *International Journal of Engineering Systems Modelling and Simulation*, pp. Volume 6, pp 12-22, 2014.

[12] P. Shetti, "Design and implementation of GSM based and PID assisted speed control of DC motor.," *International Journal of Advanced Research in Electrical Electronics and Instrumentation Engineering*, vol. 3, no. 4, 2014.

[13] T. Cross and K. Angelo, Classroom Assessment Techniques, 2nd ed, San Francisco: Jossey-Bass, 1993.

[14] T. Watson, Gudelines for Conductin Surveys, Brooks AFT, Texas: Air Force Materiel Command Human Resources Directorate, 1997.

Appendix

Below is the original text used in the pre and post assessments with the 8 technical questions followed by 5 survey questions. Q15 was only given in the post-lab assessment.

Technical Questions

Q1: If the system is first driven by a positive step input, then returned to the initial position, then an identical, but negative step input is applied to the system, when comparing the positive and negative step inputs, which of the following will be true?

The direction will be opposite, but the magnitude of the positive and negative step response will be the same.

- ○ True
- ○ False

Q2: The settling times will be the same.

- ○ True
- ○ False

Q3: The steady-state errors will be the same.

- ○ True
- ○ False

Q4: For the next questions assume that the position of the arm is measured with a sensor and a PID feedback controller is used for position control. Assume that some initial setting is selected for the proportional, integral and derivative gains and step response is measured.

Which of the following will be true if the **proportional gain** is increased but the integral and derivative gains are held constant? Select one or more:

- ☐ a. The steady-state error will be reduced
- ☐ b. The settling time will get faster
- ☐ c. The overshoot will be reduced
- ☐ d. The *reset time*, or time to get to zero steady-state error will be reduced

Q5: Which of the following will be true if the **integral gain** is increased but the proportional and derivative gains are held constant? Select one or more:

- ☐ a. The steady-state error will be reduced
- ☐ b. The overshoot will be reduced
- ☐ c. The settling time will get faster
- ☐ d. The *reset time*, or time to get to zero steady-state error will be reduced

Q6: Which of the following will be true if the **derivative gain** is increased but the proportional and integral gains are held constant? Select one or more:

- a. The steady-state error will be reduced
- b. The settling time will get faster
- c. The overshoot will be reduced
- d. The *reset time*, or time to get to zero steady-state error will be reduced

Q7: Consider the system type (defined by the number of integrators in the open-loop transfer function. What is the system type? Select one:

- a. Zero
- b. One
- c. Two

Q8: With a proportional controller and a step input, the steady-state error will be zero.

- True
- False

Survey Questions

Rate your understanding of the following concepts

Q11: I understand the difference between a linear and a non-linear system and how to model both.

- a. Strongly agree
- b. Somewhat agree
- c. Neither agree nor disagree
- d. Somewhat disagree
- e. Strongly disagree

Q12: I understand the effects that a proportional controller has on a closed-loop feedback system.

- a. Strongly agree
- b. Somewhat agree
- c. Neither agree nor disagree
- d. Somewhat disagree
- e. Strongly disagree

Q13: I understand the effects that an integral controller has on a closed-loop feedback system.

- a. Strongly agree
- b. Somewhat agree
- c. Neither agree nor disagree
- d. Somewhat disagree
- e. Strongly disagree

Q14: I understand the effects that a derivative controller has on a closed-loop feedback system.

- a. Strongly agree
- b. Somewhat agree
- c. Neither agree nor disagree
- d. Somewhat disagree
- e. Strongly disagree

Q15: The PID Design Lab helped me understand the controller design process more than solving textbook homework problems. (Post-Lab Survey Only)

- a. Strongly agree
- b. Somewhat agree
- c. Neither agree nor disagree
- d. Somewhat disagree
- e. Strongly disagree