ELEC ENG 3CL4: Introduction to Control Systems
Lab 1: Introduction

Tim Davidson
davidson@mcmaster.ca

Objective

To establish safety protocols, including those that would be applied if the lab were being done on the hardware, and to introduce the laboratory simulation set up.

Venue

The labs in this course will take place on Microsoft Teams, in a team named "ELEC ENG 3CL4 LXX ..." where XX is the number of your lab group. In each of these teams you will find channels named "TA 1" and "TA 2" where you can communicate with the TAs. You will also find channels named "Group YY" in which you can communicate with your lab partner. At the beginning of the session, partners should be arranged, and agreements should be reached as to which group will operate in which channel. Due to one of the limitations of Teams, it is not possible to display the "Group YY" groups to everyone. You can find the channels by clicking "Manage Team" and then searching for the channels. If you wish to communicate with your lab partner over a different platform, you are welcome to do that. However, interaction with the TAs will be through their channels on the Team.

Assessment

This laboratory is conducted in groups of no more than two students. Students are required to attend their assigned lab section. The assessment of this lab will occur during the lab and after submission of the lab report. You will earn a maximum of 100 marks from Lab 1 activities. Lab 1 will contribute to a maximum of 5% of your total lab grade for this course.

- As in all scenarios in which you find yourself in unfamiliar circumstances, or dealing with unfamiliar equipment, the first step is to go through an informed assessment of the risks, keeping in mind that you have the legal right to refuse unsafe work. As a result, the first component of the lab is to complete the Department's laboratory safety quiz (see Section 2). If you do not successfully complete this quiz, you will not be able to proceed with this laboratory, nor the remaining laboratories.

- The second component of the lab is to demonstrate to the TAs that you have understood the specific safety information for the regular ELEC ENG 3CL4 labs, and those

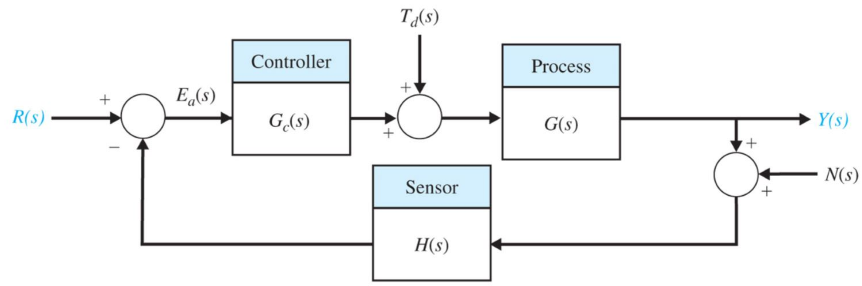

Figure 1: Feedback system with $y(t) = \theta(t)$. In our models, the sensor will be assumed to have a constant gain over the frequencies of interest. (Figure 4.3 of Dorf and Bishop, *Modern Control Systems*, 11th edition, Prentice Hall, 2008.)

associated with this instance of the labs (see Section 3).

- The third component of the lab is to demonstrate to the TAs that you have familiarized yourself with the simulation environment, and the equipment that we are simulating, by demonstrating components (chosen by the TA) of the familiarization exercises, and answering the TAs questions (see Section 4);
- The final component is a laboratory report discussed in Section 5.

1 Description of Laboratory Equipment

In the laboratories in this course, we will deal with a simulation of a closed-loop angular positioning system based around a DC motor. Such systems are often used to position heavy or difficult to move objects using a 'command tool' that is easy to move, in which case they are often called servomechanisms. One example of a servomechanism is that involved in moving the control surfaces of an aircraft using a lever in the cockpit. The goal of this lab is to introduce the (simulated) positioning system that will be used in all the laboratories in the course, and to establish the appropriate safety protocols.

A block diagram of the system that we will develop in the laboratories is illustrated in Figure 1. The "process" that we wish to control is the motor and its associated electronics. We will use the function $\theta(t)$ to denote the output of the motor, rather than the function $y(t)$ in Figure 1, because the output of the system is the angular position of the motor shaft measured by an incremental optical encoder. In this case, the sensor can be assumed to have an ideal all-pass transfer function $H(s) = 1$. The broad goal of the series of laboratories in this course will be to design controllers that ensure that the feedback system as a whole performs in desirable ways.

If we were able to do the labs for this course in person this year, you would see a system that looks similar to the one in Figure 2. On the right hand side you can see a DC motor and on top of that motor is a black flange. Although it is a bit hard to see in the figure, that black flange has angles marked on it, from 0° to 359°. The goal of the experiments will be to

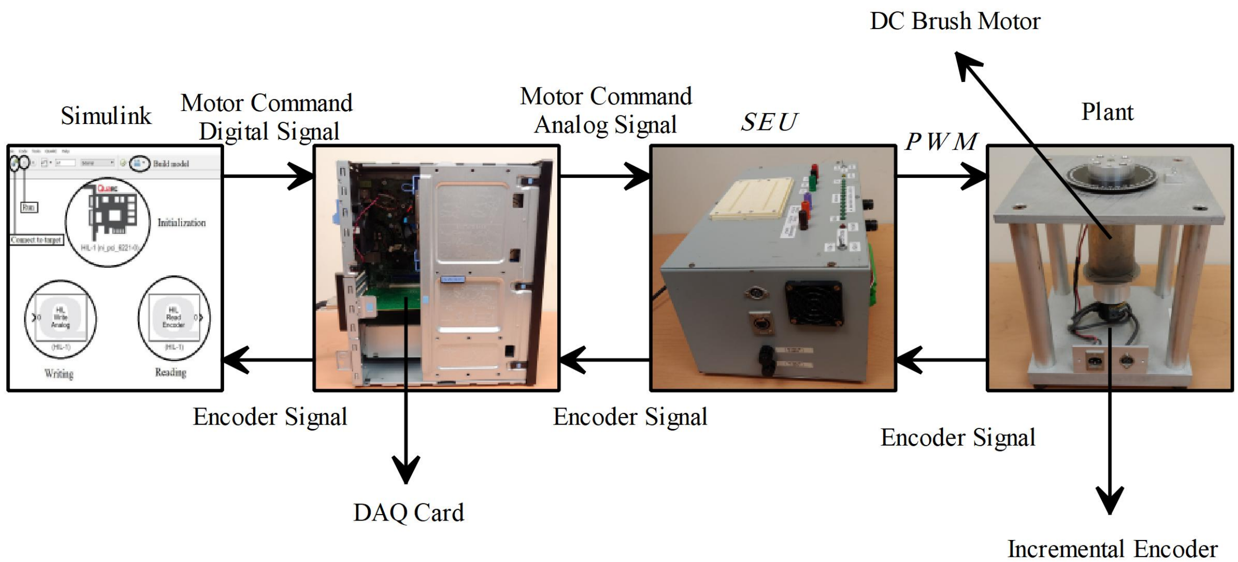

Figure 2: A view of the DC motor feedback control system used in the in-person labs.

move the flange from one angular position to another. This is a simplified version of the way that motors would be used to adjust the angles between the limbs of a robot. The picture marked "SEU" contains the electronics that drive the motor. These two systems combined constitute the "process" in Figure 1. While this process can be controlled directly in the analogue domain, by designing op-amp circuits with resistors and capacitors, if we were doing the labs in person, we would sample the encoder signal, and process it digitally, generate a digital motor command signal, and then convert that back into the analog domain. This would be done using a data acquisition (DAQ) card, which consists of analog-to-digital and digital-to-analog converters, and a real-time interface to the software. These are seen in the two boxes on the left of Figure 2. In the simulated experiments that we will perform this year, we will model the whole system in Matlab Simulink, and we will be able to visualize the behaviour of the DC motor using Quanser Interactive Labs.

As mentioned above, the "process" that we wish to control is the motor and its associated electronics. The input to the process is a control voltage, denoted $x(t)$, and the output of the process is the angular position of the shaft, denoted by $\theta(t)$, and measured by an optical encoder. A detailed description of the operation of the motor that would be used in the in-person labs is provided in the addendum to these instructions; see Avenue-to-Learn. For the purposes of our ELEC ENG 3CL4 labs, we will assume a linear model for the process. Using information about the structure of the motor and rotational Newtonian mechanics, the linear model for the operation of the motor can be described by the following differential equation:

$$J\frac{d^2\theta(t)}{dt^2} + b\frac{d\theta(t)}{dt} = K_m x(t), \tag{1}$$

where J is the rotational inertia of the motor, b is the coefficient of viscous friction in the motor structure, and K_m is the (internal) gain. Taking Laplace transforms of both sides of

(1), for a system initially at rest, we obtain the transfer function of the process:

$$s^2 J\Theta(s) + sb\Theta(s) = K_m X(s) \tag{2}$$

$$\implies G(s) = \frac{\Theta(s)}{X(s)} = \frac{A}{s(s\tau_m + 1)}, \tag{3}$$

where $A = K_m/b$ and $\tau_m = J/b$. In typical industrial applications, the gain A and the time constant τ_m are unlikely to be known in advance. Therefore, in the second laboratory we will develop experimental techniques by which A and τ_m can be estimated. In the laboratories that follow that one, we will design controllers for the servomechanism that provide desirable performance characteristics.

2 Departmental Safety Information

For the safety of everyone in the laboratory, it is critical that everyone is fluent in the Departmental laboratory safety procedures and in the specific procedures for the ELEC ENG 3CL4 laboratories. To that end, the first activity for the laboratory is to:

- Read and understand the Departmental laboratory safety information, which is available at the following url: `http://www.ece.mcmaster.ca/ug_cours/Lab_Safety.html`. Please ask the instructor or a TA if you have any questions or concerns.
- Complete the Departmental laboratory safety quiz, which is available on Avenue-to-Learn under Assessments/Quizzes.

You will not be allowed to proceed with this or any other laboratory if you have not completed the safety quiz.

3 Specialized Safety Information

Following the completion of the Departmental laboratory safety quiz, the TAs will lead the class through a safety information session that addresses the specific equipment that would be used in the laboratories, and the safety issues that arise from doing the simulations. Each pair of students will be required to demonstrate that they understand the information that has been provided before they will be allowed to proceed with the laboratory.

4 Familiarize Yourself with the Equipment

4.1 Equipment List

As mentioned above, a schematic of the DC motor feedback control system that is used in the in-person labs is given in Figure 2. The system is comprised of the following components:

i) **Simulink-based Controller**: As stated earlier, the cascade combination of the motor and its electronic drive system can be modelled in the continuous-time domain and represented by a transfer function in the Laplace domain. However, all controllers in these labs will be implemented in Matlab/Simulink environment on a personal computer equipped with a data acquisition (DAQ) card. The implementation of the control system using a digital computer, in principle, requires methods of analysis and design from the theory of discrete-time control systems. This is due to the fact that the control signal applied to the system is held constant between consecutive *sampling times* using a zero-order-hold operator (a digital-to-analog convertor). The control computations are also based on a feedback of the system output at the discrete sampling times. While there is a well-developed theory for analysis and design of such mixed discrete/continuous-time systems, in this course and its labs we will only deal with continuous-time control systems. To mitigate any potential performance and stability issues arising from the application of the continuous-time control techniques to this system, we choose a sampling time of 2 ms, which is very fast compared to the dominant mechanical dynamics of our plant and expected bandwidth of the closed-loop feedback control system. Analogously, to mitigate any potential issues arising from digital representations of the signals, we choose 16-bit analog-to-digital and digital-to-analog converters in the DAQ card. The quantization errors induced by this conversion are small compared to other sources of noise in our system. Therefore, any approximation errors due to the computer-based implementation of continuous-time controllers can be safely neglected. This will be our assumption throughout all the laboratory experiments in this course.

ii) **Data Acquisition Card**: The DAQ board links the control software to the servomechanism electronics unit (SEU). It employs a 16bit digital-to-analog (DA) convertor to convert the digital control signal to an analog voltage command for the SEU. It also reads the digital incremental encoder output which measures the angular position of the motor shaft.

iii) **Servomechanism Electronics Unit**: This module contains circuits that convert the analog motor control command to a PWM voltage that drives the motor. It also interfaces the incremental encoder signals to the DAQ card. Note that the "process" in Figure 1 refers to the cascade of the SEU and the motor.

iv) **DC motor**: The motor unit to be controlled.

v) **Incremental Encoder**: The encoder measures increment changes in the angular position of the DC motor. It produces 2400 ticks per revolution of the motor, yielding an angular position measurement resolution of 0.15 degrees.
Remark: Note that the flange visually indicates the angular position of the shaft. It needs to be manually positioned at its "zero" setting each time before running the code. This would ensure that the incremental encoder is properly initialized before starting the experiments.

4.2 Simulation Environment

The simulation environment we will use for these laboratories is a combination of Matlab Simulink and Quanser Interactive Labs. This combination works under the Microsoft Windows operating system, and the instructions below are for a system that operates under Windows. Students who use the MacOS operating system can use dual boot options, or can choose to construct a Windows virtual machine. Instructions for doing so are provided in the Appendix.

From within a Windows environment, do the folllowing, making sure that you do it in the order that is specified:

- Download and install Matlab R2020b and Simulink, from the McMaster campus portal. You can do this using the following steps:
  - Point a browser window to `https://www.mathworks.com/academia/tah-portal/mcmaster-university-31501097.html`
  - Choose "Log in to get started", and log in with your MacID.
  - Create a Mathworks account, or log in with your existing account.
  - Complete the authentication processes as necessary, and link you Mathworks account with your MacID.
  - Now you will have access to download Matlab R2020b.
  - When downloading, make sure you download Matlab, Simulink, and the Control System Toolbox. There may be other toolboxes that are of interest, but be aware of the amount of storage that this will require, and the time that will be needed to download the material. You can always add other toolboxes later on.
  - With your Mathworks account, you will have access number of features, including Matlab online, Matlab drive, and a large range of self-paced courses.
- Download and install Quanser Interactive Labs from the Quanser portal. You can do this using the following steps:
  - If you already have a version of Quanser Interactive Labs installed, please uninstall it.
  - Visit the Quanser Academic Portal at `https://portal.quanser.com`. Follow the on-screen instructions to register and activate your account. Please use your MacID.
  - From the Quanser Academic Portal, download and install the latest version of the Quanser Interactive Labs (QLabs) for Windows application. This may be called "QLabs Virtual Experiments".
  - Open the Quanser Interactive Labs application and login using your account.
  - Before commencing the familiarization exercises, close Quanser Interactive Labs.

4.3 Familiarization Exercises (50 marks)

If you are not familiar with Matlab or Simulink, and do not find the following exercises to be sufficiently helpful, you may wish to consider working your way through the following "on ramps" from the Mathworks:

- Matlab on-ramp: `https://www.mathworks.com/learn/tutorials/matlab-onramp.html`
- Simulink on-ramp: `https://www.mathworks.com/learn/tutorials/simulink-onramp.html`

To familiarize yourself with the simulation environment we will perform a few simple design examples. To prepare, please do the following, and make sure that you do it in this order.

(i) Close any open instances of Matlab and Quanser Interactive Labs.

(ii) Launch Matlab R2020b, and wait for Matlab to load completely. To test that, type "2+2" and the Matlab prompt and wait for the answer to be displayed.

(iii) Launch Quanser Interactive Labs. (Make sure that this is the new version that you downloaded according to the instructions above.)

- Select "Qube 2 – DC Motor"
- Select "Servo Workspace"
- Observe that you have a 3-D view of a box that contains a motor, and a red disk that sits on top of the motor shaft, and will be turned by the (simulated) motor.
- Click on the second icon from the left in the top right hand corner of the window to get a "top view" of disk. Click back on the left-most icon to return to the original view.
- *Troubleshooting:* If you see a yellow triangle with an exclamation mark as the left most icon, you may need to quit both Quanser Interactive Labs and Matlab, and start them again. Be sure to start Matlab first, and wait for it to load completely, before launching Quanser Interactive Labs.

(iv) Download the file `qube_servo2_quick_start_EE3CL4.slx` from the Lab 1 section in the Avenue-to-Learn shell for the course. This is a simulink file that will enable us to interact with the visualization of the motor, and to design controllers to modify its behaviour.

(v) Open that file in Matlab, and rearrange the windows on your screen so that you can see the Simulink block diagram, the windows labelled "Motor Voltage" and "Servo Angle," and the Quanser Interactive Labs window.

(vi) Observe the parts of the Simulink block diagram

- In the top left corner we have a square-wave generator. If you click on the box, you will see that the square wave is at 0.4Hz, and has a particular amplitude. A square wave of this amplitude corresponds to a command that the motor alternately position the disk so that it is 45° to one side or the other of the neutral position. After an amplitude scaling and "Degrees-to-Radians (D2R)" conversion, this becomes the position command signal $r(t)$ in Figure 1.
- In the bottom right corner we have the measured angular position of the disk.
- In the middle at the top, this signal is subtracted from $r(t)$ to form $e_a(t)$, and that signal is fed into a gain unit called the "Proportional Gain" because it is proportional to the position error.
- In the centre of the diagram, the measured angular position is multiplied by -1 and then filtered by the transfer function $\frac{150s}{s+150} = \frac{s}{1+s/150}$ which corresponds to low-pass filtering and differentiation. Essentially that gives us a (smoothed) measurement of the (negative of the) angular velocity of the disk. That signal is fed into a gain unit called the "Derivative Gain" because it is proportional to the angular speed of the disk. (Since the position command for the disk is a square wave, this is essentially the derivative of the position error.)
- The proportional and derivative components are then added to form a "Motor Voltage" signal that will be used to drive the (simulation of the) motor.

(vii) Press run to begin the simulation.

- Observe in the Quanser Interactive Labs window that the (simulated) motor turns the disk to and from points that are around 45°on either side of the neutral position. Confirm this by using the "top view" in the Quanser Interactive Labs window.
- Observe the Servo Angle position and the Motor Voltage in the "oscilloscope-style" windows. Note that the Servo Angle does not quite make the desired position. Go back to the Quanser window and confirm this. (This is due to non-linear effects in the motor, notably stalling, when it is moving slowly.)

(viii) Click on the arrow below "Pause" in the Simulink window and select "Simulation Pacing". Set "Simulation time per wall clock second" to 0.5 to slow the simulation down a bit so that we can see what is going on a little more clearly.

- Observe the overshoot each time the disk turns, both in the Servo Angle and Quanser windows. Also observe the fact that we don't quite get to 45°.

(ix) Click on the "Proportional Gain" box, and increase the proportional gain to 2.

- Observe that there is now greater overshoot, but that the disk settles down at an angle much closer to the target of 45°. This is because the motor is being driven harder and does not stall to the same extent. You can see this in the Servo Angle display. You can also see that the Motor Voltage is larger than in the previous case (and hence the motor is being driven harder).

(x) Increase the proportional gain to 4.

- Observe the much larger overshoot, and the fact that the motor takes longer to settle. You can see this in both the Quanser window and the Servo Angle display. Observe that the Motor Voltage is also much larger.

(xi) Click on the "Derivative Gain" box to see if we can smooth out the motion with some derivative gain, like we did in the first tutorial, while maintaining the high speed of the response. Set the derivative gain to 0.1.

- Observe the far more desirable response, both in the Quanser window and the Servo Angle display. Also, observe that there is still a rather large peak in the motor voltage, but it settles very quickly.

(xii) Increase the derivative gain to 0.15.

- Although the spike in the Motor Voltage has become a little larger, the performance of this particular controller in terms of the Servo Angle is really quite good.

These familiarization exercises will be assessed via a demonstration of successful completion of the exercises to the TAs, and through a question and answer session with a TA.

5 Laboratory Report (50 marks)

Each group must submit an electronic report through the course webpage due by 11:59pm one week from the day of the lab. For example if your lab is on Monday, your report would be due by next Monday at 11:59pm. The report should be formatted in single-column, single-spaced, using Times New Roman 12 or equivalent font. The group members should clearly state their individual contributions to the report in a statement in the beginning of the report. The laboratory report must include the following items:

- A full description of simulation environment for the lab.
- Plots of data obtained during the lab along a brief description of the data.
- A discussion of your observations in the familiarization exercises.

Appendix: Options for users of MacOS

Unfortunately, the Quanser Interactive Labs tool is only available under the Windows operating system. Quanser does not have any plans to port that tool to MacOS. Fortunately, for those who have Macs with an Intel processor, there are several options that are available to you, notably constructing a dual boot system, running Windows within MacOS using Parallels, and running a Windows virtual machine within MacOS using VMwareFusion. Since both Windows and VMwareFusion are available free of charge on the McMaster Hub `https://mcmaster.onthehub.com/WebStore/Welcome.aspx`, the dual boot and VMwareFusion options are quite attractive. That said, Parallels is often said to be the easiest option.

- A video tutorial on the boot camp option is available here: `https://www.youtube.com/watch?v=Hmm9Q-T0oTo`. A more involved option that uses an external storage device is available here: `https://www.youtube.com/watch?v=3IhW722IMwE&feature=youtu.be`
- A video tutorial on the VMwareFusion option is available here: `https://www.youtube.com/watch?v=mtxtqw95484`

Please note that you will need to download Matlab (and Quanser Interactive Labs) from within your Windows environment.