

# SFWR ENG 4A03 Lab 2: PID Control of a Motor

Due: Noon Friday October 22, 2004

Before the lab, one of you checkout your lab group's working directory from the subversion repository:

```
svn checkout https://websvn.mcmaster.ca/trunk/se4a03/groups/groupX
```

where X is your lab group number. Create a revision controlled subdirectory for lab2 with the following commands:

```
cd groupX
svn mkdir lab2
```

Don't forget to commit it for your other groups members! All of the files you are asked to create below should be placed under revision control in this directory. The answers to your prelab and postlab questions should be put into a PDF file called `lab2-report.pdf` that you put in your group's `lab2` subdirectory. You can use LaTeX, OpenOffice, or some other system, as long as it generates PDF.

For an example latex source for this lab see:

```
https://websvn.mcmaster.ca/trunk/se4a03/labs/lab2/
```

Once you have completed the lab, you must submit it by "tagging" it for marking before the lab due date with the command:

```
svn copy https://websvn.mcmaster.ca/trunk/se4a03/groups/groupX/lab2 \
https://websvn.mcmaster.ca/tags/se4a03/groups/groupX/lab2 \
-m "Tagging lab2 for marking."
```

(where X is again replaced by your group number). Alternatively you can use the subversion Windows Explorer interface TortoiseSVN or some other subversion client. This is the submission that will be marked by the TAs!

## Pre-Lab Exercises

1. Given that the model for the (open loop) transfer function for the angular position  $\theta(t)$  of a motor is:

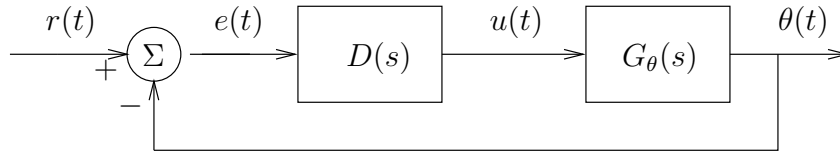
$$G_{\theta}(s) = \frac{\Theta(s)}{U(s)} = \frac{A}{s(\tau s + 1)}$$

where  $A$  and  $\tau$  are positive, real valued constants.

- a) What is the transfer function for the angular velocity  $\dot{\theta}(t)$  denoted  $G_{\dot{\theta}}(s) = \frac{\dot{\Theta}(s)}{U(s)}$ ?
- b) What is the steady state value of  $\theta$  in open loop response to a step input:

$$u(t) = \begin{cases} U_o, & t \geq 0 \\ 0, & t < 0 \end{cases}$$

2. Consider the motor given by  $G_{\theta}(s)$  in Question 1 in the feedback control configuration:



where  $D(s) = K$  (i.e. proportional control), with  $K > 0$ .

- a) Sketch the root locus.
- b) What is the closed loop transfer function:

$$G_{\theta_{cl}}(s) = \frac{\theta(s)}{R(s)}$$

- c) Does the closed loop system track steps? i.e. what is the steady state error of the system:

$$e_{\infty} = \lim_{t \rightarrow \infty} e(t)$$

- d) Suppose that the closed loop system is critically damped (i.e. characteristic equation has repeated roots) for  $D(s) = K_*$ . In this case, what is the value of  $\tau$  in terms of  $K_*$  and  $A$ ?

3. Repeat Question 2 (a) and (b) using Matlab to do the root locus for  $A = 2$  and  $\tau = 20$  with

- i) Integral control  $D(s) = \frac{K}{s}$
- ii) Proportional Integral Control  $D(s) = \frac{K(s+z)}{s}$
- iii) PID Control  $D(s) = \frac{K(s+z_1)(s+z_2)}{s}$

For each of the cases indicate whether it is possible to stabilize the system. For (ii) and (iii) show how your choice of zeros can affect closed loop stability.

4. The Root Mean Square (RMS) value of a signal  $f(t)$  that is periodic with period  $T$  is given by the equation

$$\sqrt{\frac{1}{T} \int_0^T (f(t))^2 dt}$$

RMS values are often easier to measure and more accurate than trying to determine peak amplitudes.

- a) Show that the RMS value of  $u(t) = A \sin \omega t$  is  $\frac{A}{\sqrt{2}}$ .
- b) Suppose  $u(t) = A \sin \omega t$  is the input to a transfer function  $G(s) = \frac{Y(s)}{U(s)}$ . Show that in the steady state the RMS value of  $y(t)$  is  $|G(j\omega)| \frac{A}{\sqrt{2}}$ . Thus:

$$\frac{\text{RMS value of } y(t)}{\text{RMS value of } u(t)} = |G(j\omega)|$$

- c) Another way of estimating  $\tau$  is to experimentally determine the bandwidth of the closed loop system  $G_{\theta_{cl}}(s)$  corresponding to the -3dB frequency  $\omega_{bw}$ . Then we have:

$$|G_{\theta_{cl}}(j\omega_{bw})| = \frac{1}{\sqrt{2}}$$

In this case what is the value of  $\tau$  in terms of  $K$ ,  $A$  and  $\omega_{bw}$  when the compensator in the closed loop system is  $D(s) = K$ ?

## Lab Exercise

To do the following exercises, login as user `rtlab` under Windows.

1. Run the `MAX` tool (click on the National Instrument Measurement & Automation icon on the desktop) and create:
  - A virtual input channel called `position` with units degrees with a range of  $[-180, +180]$  degrees, sensor type `voltage` with range  $[-5, +5]$  Volts. Be sure to specify that it is `range mapped` and it is a single-ended referenced signal.
  - A virtual output channel called `control` with units degrees with a range of  $[-360, +360]$  degrees, with output voltage range  $[-10, +10]$  Volts. Be sure to specify that it is `range mapped`.

### Part I: The importance of $K_i$

2. Connect the potentiometer sensor from the motor to Analog Input Channel 0. Make sure that the `BNC/temp` switch is set to `BNC` and the `FS/GS` switch is set to `GS`.
3. Connect Analog Output Channel 0 to the input of the operational amplifier.
4. Test the connections from within `MAX` by right clicking on the `control` output and selecting `Test`. Change the output value to `60`. The motor should start to spin. Click on the `position` tab and you should see a graph of the position sensor changing linearly from  $-180$  to  $+180$ . Click on the `control` sensor again and set the output to `0` to stop the motor before exiting `MAX`.
5. In Labview, load the PID demo and save it as `mypid.vi`. Edit this file to make the controller suitable for use with the motor. Note that the lower and upper limits on the output should be change to  $-360$  and  $+360$  respectively. Save the `mypid.vi` file and ask the TA to inspect it before running it. **Once the TA has approved your design, have the TA help you connect your design to the motor.**
  - i) Set proportional gain  $K_p$  to a value of `1.2` and set the integral  $K_i$  and derivative  $K_d$  gains to `0`. Test the different setpoints  $-120, -90, -45, 45, +90, +120$  and record any steady state offset in the output from the setpoint.
  - ii) Now set  $K_i$  to be greater than zero, between say `1` and `5`. Check to see what the steady state errors at the previous set points are now. Record the steady state error values.

### Part II: Estimating the DC gain $A$

6. Connect the motor's position encoder to the digital inputs for the counter (the TA will do this for you).
7. The encoder has 1024 pulses/revolution. Load the `BNC-2120` encoder example from the previous lab. Save the file as `MotorEncoder.vi` and edit the example so that it read the correct angle for the motor's position.
8. Create a VI called `velocity.vi` to do open loop control of the motor (i.e., have some sort of way of setting the analog output channel `control` to a desired value. Your VI should read the encoder for the position of the motor and use this to estimate the value of  $\dot{\theta} \approx \frac{\Delta\theta}{\Delta t}$ . The VI should plot the

estimate of the derivative and have a numeric display of the sliding average of the derivative over a user settable number of samples.

**HINT:** You might want to see the internals of the PID block for their method of computing the derivative.

9. **Once the TA has approved your design, have the TA help you connect your design to the motor.** Test the different setpoints  $-120, -90, -45, 0, 45, +90, +120$  and for each setpoint record the steady state value of your estimate of  $\theta$ .

### Part III: Estimating the time constant $\tau$

10. Open your `mypid.vi` file and save it as a file called `pidbode.vi`. Edit this file to replace the setpoint control with a basic function generator. Add in controls to allow you to change the amplitude and frequency of a reference signal into the motor control loop. Add plots of the RMS value of the reference signal and the motor position as well as a numeric display of these values.
11. **Once the TA has approved your design, have the TA help you connect your design to the motor.** Using this set up, make  $K_i = K_d = 0$  and then tune the value of  $K_p$  until the system is critically damped (i.e. just before overshoot occurs). Record the value. Use a square wave reference signal of a reasonably low frequency to help you tune the system. Note that this is the value of  $K_*$  in Question 2 for your motor.
12. Leave  $K_i = K_d = 0$  and  $K_p = K_*$ . Change the input to your system to a low frequency sinusoid of amplitude 90. Increase the frequency of the sinusoid until the RMS value of the output is  $\frac{1}{\sqrt{2}}$  of the RMS value of the input. Record this frequency as  $\omega_{bw}$  of Question 4.

## Post-Lab and Other Exercises

### 1. Facilitating Good Design

In step 1 of the Lab Exercise you used the Labview MAX tool to convert the position potentiometer voltage reading into an angle within Labview and the controller output from an angle into a voltage to move the motor to the correct angle. Later in the lab the angle potentiometer was replaced by the encoder. Think of the resulting impact on your Labview controllers. Explain, referring to the 4 variable model decomposition described in class, how Labview's MAX tool helps non-software engineers create well designed "code".

### 2. System Identification

Analyze your data to obtain estimates of  $A$  and  $\tau$ . You should have two estimates of  $\tau$  from Part III of the lab. How do the two estimates compare? What do you conclude about your results? For the value of  $\tau$  obtained from  $\omega_{bw}$  in Part III and the value of  $A$  obtained from Part II of the lab, what should be the value of  $K_*$ ? What is your best estimate of the open loop transfer function  $G_\theta(s)$ ?

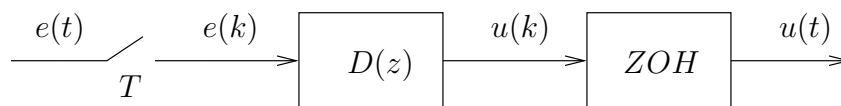
### 3. Validation

Use Matlab to simulate the closed loop step response of the transfer function you obtained in Question 2 when  $D(s) = 1.2$ . Explain any discrepancies from the data you obtained in Part I of the lab and the simulation results and mathematical results of Pre-Lab Question 2(c).

#### 4. Practical Considerations

In the lab we used two different sensors to estimate the motor's position. In Part I we used a potentiometer with readings between  $[-5,+5]$  V being read by a 16-bit A/D on a Data Acquisition (DAQ) card. In Part II we used an encoder with 1024 pulses/revolution and A and B channels going into a 24-bit up/down counter also on the DAQ card. Both of these sensors have their readings scaled in software to give  $[-180,+180]$  degrees for one full revolution of the motor.

- a) What is the resolution in both percentage and degrees of
    - i) The encoder (assuming no other sources of error other than the finite number of bits)?
    - ii) The potentiometer read by the 16-bit A/D (assuming the potentiometer provides a perfect reading and there are no other sources of error other than the finite number of bits)?
    - iii) The Datasheet for the NI PCI-6035E data acquisition board used in the lab lists the absolute accuracy in a  $[-5,+5]$  volt range as 1.790 millivolts (mV). How does this compare to your previous answer? Is this not unexpected?
    - iv) In reality the potentiometer provides a voltage that has an accuracy of  $\pm 0.5\%$  of what a perfect linear sensor would produce. What size error does this translate into in degrees?
    - v) Which sensor's estimate of  $\theta$  is more accurate when the potentiometer's tolerance is taken into account?
  - b) The analog potentiometer has a discontinuity from  $+5$  to  $-5$  or vice versa as the system moves beyond 180 degrees of rotation. While it is possible to take this discontinuity into account in software, what might still be a problem?
  - c) Assuming that the motor starts with the 24-bit counter initialized to zero, how many revolutions would it take for the encoder to roll over the counter possibly resulting in a large discontinuity in the estimated position (if the software does not take this into account)? What would be the approximate size of the discontinuity?
  - d) What are the advantages and disadvantages of each type of sensor?
5. Discrete systems with poles on the unit circle are not BIBO stable. As an example to illustrate this fact, consider the following system:



Assume the sampling period  $T = \frac{1}{2}$  second, input  $e(t) = \cos(2\pi t)$  and

$$D(z) = \frac{U(z)}{E(z)} = \frac{z}{z+1}$$

We will show that  $u(k)$  (and hence  $u(t)$ ) grows without bound in this case.

- a) Sketch  $e(k)$  for  $k = 0$  to 4.
- b) Compute  $E(z) = \mathcal{Z}\{e(k)\}$ .
- c) Compute  $u(k)$ . Is it clear that  $|u(k)|$  is not bounded?
- d) Sketch  $u(t)$ , the output of the ZOH, for  $0 \leq t < 2$ .
- e) What other bounded inputs of the form  $e(t) = \cos(\omega t)$  will cause an unbounded output from  $D(z)$ ?