Lab 3: Model-based Position Control of a Cart

I. Objective

The goal of this lab is to help understand the methodology to design a controller using the given plant dynamics. Specifically, we would do position control of a cart by developing various controllers and then comparing their performances.

II. Equipment

- Cart system (no attachments) and power supply.

III. Theory

1. Plant Dynamics

Since the plant here, a motor with a cart, is the same as Lab 2, we just provide a summary of its dynamics below. Please refer to Lab 2 for the derivation.

$$\left(m_c r^2 R_m + R_m K_g{}^2 J_m\right)\ddot{x} + \left(K_t K_m K_g^2\right)\dot{x} = \left(r K_t K_g\right)V \tag{1}$$

In the equation above:

V is the input voltage (volts)
m_c is the mass of the car (kilograms)
r is the radius of the motor gear (meters)
R_m is resistance of the motor windings (ohms)
K_t is the motor torque constant (N*m/A)
K_m is the back EMF constant (V*s/rad)
K_g is the gearbox gear ratio (no units)
J_m is the moment of inertia of the motor (kg*m^2)

The motor-cart system model is a second-order model, whose dynamics we now examine:

2. Second order dynamics

A second order linear system is described by the general differential equation of the form:

$$\ddot{y} + 2\xi\omega_n\dot{y} + \omega_n{}^2 y = bu(t) \tag{2}$$

The above equation when expressed in the Laplace domain becomes:

$$\frac{Y(s)}{U(s)} = \frac{b}{s^2 + 2\xi\omega_n s + \omega_n^2} \tag{3}$$

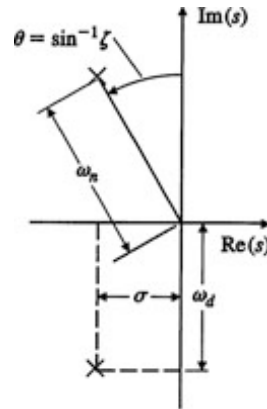For $\xi \leq 1$, the above system has complex poles, which are depicted on a graph in the figure below:

Figure 1: Location of a pole in terms of ω_n and ξ.

The parameter ω_n is called the natural frequency and is a measure of the speed of the response of the second order system while ξ is a measure of the damping in the system.

For complex poles in the left half plane, we make the following important observations from the figure above:

- The length of the complex vector from the origin to the pole is ω_n
- The sine of the angle of the vector with the positive imaginary axis equals ξ i.e. $\sin\theta = \xi$

A typical step response for a second order system is as shown in figure below:
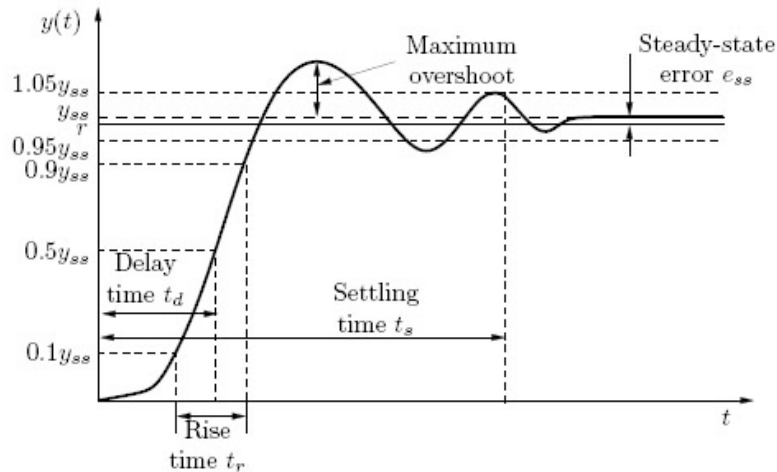

Figure 2: Typical step response of a control system.

Two important performance metrics for second order systems are their <u>rise time</u> and <u>maximum overshoot</u>. In terms of parameters ω_n & ξ they are given as:

$$t_r \approx \frac{1.8}{\omega_n} \tag{5}$$

$$M_p = e^{-\pi\xi/\sqrt{1-\xi^2}} \tag{6}$$

NOTE:

In order to get faster response (small t_r) and smaller overshoot (small M_p) we would like the closed loop poles to have large radial distance (leading to a large ω_n) and large angle θ (leading to a higher ξ).

III. Pre-lab

1. Position Controller Design

We set the following performance objectives to be achieved by the feedback system for cart position control (step response) for step amplitude of 0.1m:

1. **Rise time $t_r \leq 0.17\ s$**
2. **Maximum overshoot $M_p \leq 4\%$**

The objective is to design a feedback system that will help us achieve these desired performance specifications. The general guidelines to proceed with the design are outlined below:

Step 1: Plant Model

Use your state space or transfer function representation of the system from Lab 2. Determine the poles of this transfer function. Is the plant stable?

Step 2: Design parameters

Using the given desired performance objectives and equations (5) and (6), come up with desired values of ω_n and ξ (these are ranges).

Step 3: Proportional Controller

The first controller that we will try is a proportional controller K. Feel free to use your Simulink diagram from last lab, but please include it again in this lab report. Vary the value of K over the range 10-50. Put 4-6 of these plots superimposed on each other in your report.

- As K increases, what happens to the rise time and overshoot?
- With just a P controller, can the desired performance specifications be achieved?

Find the *smallest integer K* value for which at least the rise time performance specification is met. Plot the result for this K and show what specifications it meets.

The above observations can also be made from the root locus plot. Plot the root locus for the plant transfer function and answer the following:

- For complex poles of the closed loop transfer function, as K increases what happens to the radial distance and the angle θ?
- Going back to Figure 1 and equations (5) and (6), what does this mean will happen to ξ and ω_n as K increases?

Note that this is not the desired behavior we stated in the enclosed note above.

Step 4: PD Controller

In order to meet the design constraints, we will need derivative action which will help "apply the brakes earlier."

In order to reduce the overshoot we use derivative action in conjunction with proportional action. Its form is as follows:

$$k(t) = k_P\big(x_{ref} - x\big) + k_D\big(\dot{x}_{ref} - \dot{x}\big) = k_P e(t) + k_D \dot{e}(t) \qquad (7)$$

Observe that a PD controller introduces a zero in the plant transfer function at $s = -k_P/k_D$.

MATLAB does not allow you to put a pure zero. **Why is introducing a pure zero a bad idea?** Instead, we will introduce a pole/zero pair with the pole so far away that it hardly affects the rest of the system dynamics. We will arbitrarily set the pole at $s = -200$ by making the denominator of the controller $0.005s + 1$.

Now put the plant dynamics equation from Step 1 in unity feedback with this new PD controller and **obtain the transfer function from x_{ref} to x.**

For further analysis we assume that we will place the zero at $s = -15$. This location of the zero can be found through design techniques which will be covered in subsequent labs. This changes the form of the PD controller numerator to $K(s + 15)$. Plot the root locus for this transfer function of the "modified" plant which includes the controller (zoom in towards the origin). By comparing this root locus with the one plotted in Step 3 make the following observations:

- A left half plane zero tends to pull the root locus towards it.
- There exists a portion of the root locus which has the following properties:
    i.   Has complex closed loop poles and
    ii. ω_n and ξ both increase as K increases

This is what we desire as stated in the enclosed note above.

Determine a *small K* value from this <u>root locus</u> for which the performance specifications are met and then use this value to determine initial values of k_P and k_D for the controller.

Note: For this task, you will find two things handy: 1) You can specify gain values to plot on your root locus (see `help rlocus`). 2) Once you plot your root locus, the Data Cursor (Tools → Data Cursor) will display the gain (K), damping (ξ), and frequency (ω_n). You can drag this cursor across all the plotted points until you find a K value that meets the parameters you solved for in Step 2.

Now create two Simulink block diagrams to simulate the cart in feedback with a PD controller:

1) The zero is set at s = -15, so controller zero is of the form $K(s + 15)$.
2) The more general PD controller of the form $k_D s + k_P$.

First verify that your chosen value of K meets the design specifications. **Why is there a discrepancy in rise time and overshoot values between the root locus plot and simulation?** (**Hint:** see `help rlocus` and compare system architecture) Then increase K by some regular interval and plot on the same graph. Then do the same with k_D on the other diagram (doesn't have to be the same interval).

- As K and k_D increase, what happens to the rise time and overshoot? (make sure trends are evident on the plots)

IV. Lab

1. **Simulink Model for Experiment**
   Replace the plant in your simulation model with the actual physical plant model (using QuaRC components).
   Note: for this lab, please set the saturation limits for the voltage input as 8 (V) and -8 (V). Save the data (workspace variables) in the experiments for Post-Lab use.

2. **Proportional Control**

Experiment with the proportional controller on the hardware by trying out various gain values and observe the variation of rise time and overshoot. Provide the K value and plot for the following system responses: 1) that is as close to meeting both criteria at the same time as you can get it, and 2) that meets the rise time criteria but ignores the overshoot.

3. **PD Control**

Now implement the PD controller from Step 4 of the pre-lab that satisfies the design constraints in theory. It is your choice which controller to use ($K(s + 15)$ or $k_D s + k_P$). When you run it on the hardware, are the desired performance specifications met? Include a plot of your hardware response.

If the constraints are not met, then do an "intelligent" tuning of the gain values until the required specifications are met. By now you should know three different tuning methods: modifying k_P, K, and k_D as well as the general trends when you change each. Show a plot of your final run and show that it meets the design specifications. Don't forget to report your final controller values.

The lab instructions for operation of the hardware remain the same as for Lab 2.

Note: Please show the lab GSI your results for each controller you designed (in IV-2 and IV-3).

4. **Extra Credit (1 pt each)**

Using your TUNED PD controller from part 2, what is the performance like for the following input signals? MAKE SURE YOUR END TIME IS FINITE AND ≤ 10 (in case something goes wrong).

- Pulse generator (50% pulse width, amplitude ≤ 0.06m, period ≥ 3 sec)
- Sine wave (amplitude ≤ 0.08m, frequency ≤ 1 Hz)
- Saw/Triangle wave (amplitude ≤ 0.04m, frequency ≤ 0.5 Hz)
- Exponentially-DECAYING sine wave (initial amplitude ≤ 0.12m, pick a reasonable time constant and frequency so that the decay is evident, but not too sudden or gradual)

Include your modified Simulink/QuaRC models as well as plots of the input signal and the actual hardware output (superimposed) for at least 2 periods.

V. Post-Lab

1. **Proportional Control**
Plot your simulation results together with your hardware responses for the case that meet the rise time criteria. Use the K value that you found in the lab hardware experiments. What is the steady state error for each case? Can you observe any difference between the simulation and hardware response? Explain that. Can you improve the hardware steady state performance by designing another controller? How?

2. **PD Control**
Plot your hardware responses of Proportional control (the one that meets the rise time criteria) and PD control (the one that meets both criteria) together. What is overshoot percentage for each case? Why are we able to tune the PD controller to perform better than just P control?

Find the pole locations in the root locus using the K (or k_D, k_P) values you found for these two cases.

Compute the overshoot and rise time from the root locus using Equations (5) and (6). Recall the observations you made from the root locus in the prelab. Do the hardware experimental results match your observations?

VI. Revision History

Semester and Revision	Author(s)	Comments
Fall 2010 Rev. 1.4	Jansen Sheng, Wenjie Chen	Modifications made based on Fall 2009 lab and solutions.
Fall 2009 Rev. 1.3	Justin Hsia	Corrections made based on Fall 2009 student reactions – changed rise time criterion and pole/zero locations
Fall 2009 Rev. 1.2	Justin Hsia	Incorporated moment of inertia and changed rise time criterion
Winter 2008 Rev. 1.1	Justin Hsia	Formatting, made corrections based on Fall 2008 student reactions
Fall 2008 Rev. 1.0	Justin Hsia	Lab formatting, extra credit
Fall 2008 Rev. 0.0	Pranav Shah	Initial version of lab write-up