Lab 4: Model-based Position Control of a Cart

"Remember that all models are wrong; the practical question is how wrong do they have to be to not be useful." – George Edward Pelham Box

1 Objectives

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

2 Equipment

Quanser cart system (no attachments) and power supply. See Lab 3 for details.

3 Theory

3.1 Plant Dynamics

The plant in this lab is the same as in Lab 2. Recall its dynamics

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

where the parameters are described in Table 1.

Parameter	**Unit**	**Description**
V	V	input voltage
m_c	kg	mass of the car
r	m	radius of the motor gears
R_m	Ω	resistance of the motor windings
K_t	Nm/A	torque motor constant
K_m	Vs/rad	back EMF constant
K_g	-	gearbox ratio
J_m	$\mathrm{kg\,m^2}$	moment of inertia of the motor

Table 1: Parameters of the cart system

3.2 Second Order Dynamics

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

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

where $u(t)$ is the input to the system. 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 (viscous) damping in the system.

When expressed in the Laplace domain, Equation (2) reads

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

For $\xi \leq 1$, the above system has complex poles, which are depicted in the complex plane in Figure 1. For complex poles in the left half plane, we make the following important observations from Figure 1:

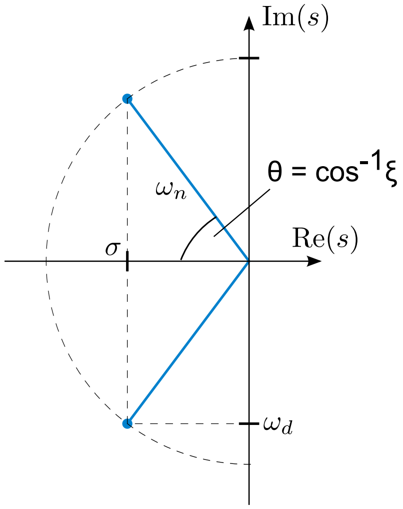

Figure 1: Location of the poles of (3) in terms of ω_n and ξ.

- The length of the complex vector from the origin to the pole is ω_n
- The cosine of the angle of the vector with the negative real axis equals ξ, i.e. $\cos\theta = \xi$

A typical step response for a second order system of the form (3) is shown in Figure 2. Two important

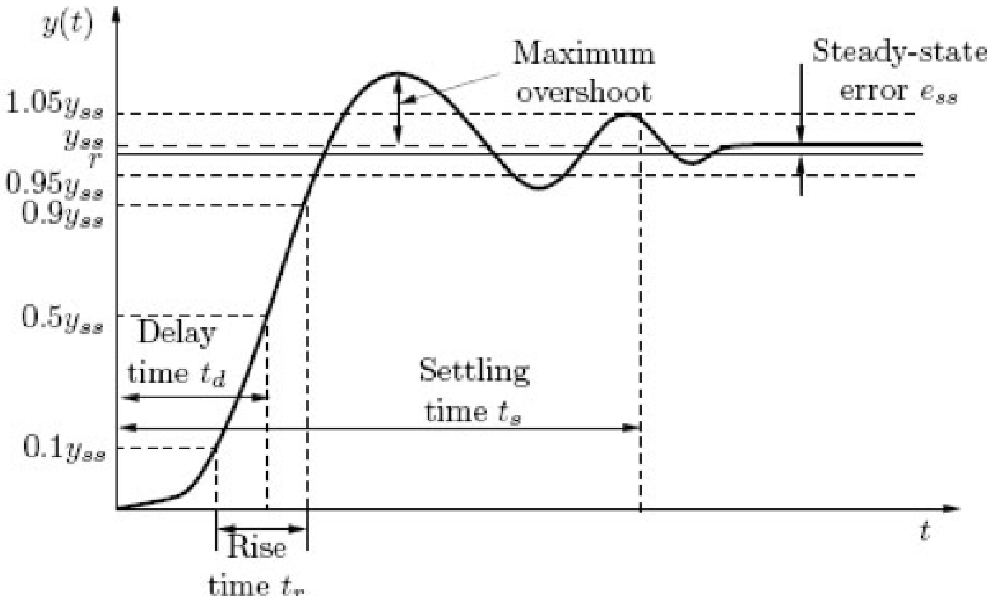

Figure 2: Typical step response of a linear second order system

performance metrics for a SISO systems is its *rise time* and *maximum overshoot*. For a second order system as in (3) the overshoot can be determined analytically and is given by

$$M_p = e^{-\pi\xi/\sqrt{1-\xi^2}} \tag{4}$$

There is no explicit formula for the rise time as a function of the parameters ω_n and ξ. However, the following third order polynomial provides an approximation with maximum error $< 0.5\%$ for values $0 < \xi < 0.9$ (see Nise, page 181):

$$\omega_n t_r \approx 1.76\,\xi^3 - 0.417\,\xi^2 + 1.039\,\xi + 1 \tag{5}$$

In order to get a fast response (small t_r) and small overshoot (small M_p) we would like the closed loop poles to have large radial distance (leading to a large natural frequency ω_n) and large angle θ (leading to a high damping ξ).

4 Pre-Lab

4.1 Position Controller Design

We set the following performance objectives to be achieved by the feedback system for cart position control (step response) for a step amplitude of 0.15 m:

1. Rise time $t_r \leq 0.2\,\mathrm{s}$ (This is several times faster than Pre-Lab 3)
2. Maximum overshoot $M_p \leq 5\%$

- Will the amplitude of the input affect rise time and maximum overshoot? Explain.

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

4.1.1 Plant Model

Use your state space or transfer function representation of the system from Lab 3.

- Determine the poles of this transfer function. Is the plant stable?
- Using the given performance objectives and equations (5) and (4), determine ranges for the desired values of ω_n and ξ.

4.1.2 Proportional Controller

The first controller that we will try is the proportional controller K from the previous lab. Feel free to use your Simulink diagram from last lab, but please include it again in this lab report. **Make sure your system output is in meters and your system input is in Volts**. Vary the value of K over the range 10-60. Plot the step functions for five of these values superimposed in a single plot.

- As K increases, how does it affect 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 the rise time performance specification is met. Plot the response for this K and mention whether it meets the overshoot specification.

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 (4), how does an increase in K affect ξ and ω_n?

4.1.3 PD Controller

In order to meet the design constraints, we will need derivative action which will help "apply the brakes earlier." We will determine the appropriate action through the following steps:

- Using the pre-determined rise time and overshoot specifications, we determine the appropriate damping, ξ, and natural frequency ω_n
- We will determine the appropriate controller values, K_d and K_p, using the root locus that correspond to the determined damping and natural frequency values.

In order to reduce the overshoot we use derivative action in conjunction with proportional action. This type of controller is called "PD Controller". Its form is as follows:

$$k(t) = k_P(x_{ref} - x) + k_D(\dot{x}_{ref} - \dot{x}) = k_P\, e(t) + k_D\, \dot{e}(t) \tag{6}$$

where $e(t) = x_{ref}(t) - x(t)$ is the error.

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 choose to set the pole at $s = -250$, which corresponds to a denominator $\frac{1}{250}s + 1$ of the controller[1].

Now put the plant dynamics equation from Step 1 in unity negative 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 $s = -12$, that is, $\frac{k_P}{k_D} = 12$. This location of the zero can be found through design techniques which will be covered in subsequent labs. The numerator of the PD controller's transfer function becomes $k_D(s + 12)$. That is, the controller has the following dynamics:

$$H(s) = k_D \frac{s + 12}{\frac{1}{250}s + 1} \tag{7}$$

Plot the root locus for this transfer function of the "modified" plant which includes the controller (make sure to zoom in towards the origin, that's where the interesting things happen). By comparing this root locus with the one plotted in Step 3 make the following observations:

[1] Note that we chose $\frac{1}{250}s + 1$ rather than $s + 250$ so that this term will have a small effect for small values of s.

- 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:
  - it has complex closed loop poles
  - ω_n and ξ both increase as k_D increases

This is a desired behavior, as we can increase the systems response speed (linked to ω_n) and at the same time decrease its overshoot (linked to the damping ξ).

Determine a *small* k_D value (less than 50) from this root locus for which the performance specifications are met. *Hint:* For this task, you will find two things handy: 1) You can specify gain values to plot on your root locus (see `doc rlocus`). 2) Once you plot your root locus, the Data Cursor (Tools → Data Cursor) will display the gain k_D, damping ξ, and frequency ω_n. You can drag this cursor across all the plotted points until you find a k_D value that meets the parameters you solved for in Step 2.

Now create a Simulink block diagram to simulate the cart in feedback with a PD controller. The zero is set at $s = -12$, i.e. the controller numerator is of the form $k_D(s + 12)$.

- Verify that your chosen value of k_D meets the design specifications. Observe that there is a discrepancy in overshoot value between the root locus plot and simulation. This is due to the fact that `rlocus` approximates the value of rise time by assuming that the system is second-order with no zeros, i.e. it computes the theoretical value of overshoot if the system had two dominant poles and no zero. Our system has a zero at $s = -12$. This zero is close to the poles for the selected gain parameter, so it will affect the step response quite a bit.
- Find the range of values of the gain $k_D \leq 50$ for which the specifications are met. Plot the system response for 4 values of k_D in this range.
- To test the effect of the zero, change the zero to $\frac{k_P}{k_D} = 15$. Can you find a value of k_D (less than 50) for which the system meets the specifications?

5 Lab

5.1 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 the system response for the following:

- minimal K value such that the rise time is less than 0.2 s
- maximal K value such that the overshoot is less than 5%
- one value between these two

You will likely observe that the system has a non-zero steady state error. Why do you think that is? We will come back to the steady-state error problem later. For now we focus on reducing the overshoot, by adding a derivative term to the controller.

Plot the response of the system for $K = 100$, and compare it to the system model from the pre-lab. You will observe in particular that the rise time is longer for the actual system. Give reasons for why this might be the case.

5.2 PD Control

Now implement the PD controller from Step 4 of the Pre-lab that satisfies the design constraints in theory. The controller is of the form $k_D \frac{s+12}{s/250+1}$. When you run it on the hardware for k_D in the range computed in the pre-lab, 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 k_D and k_P until the required specifications are met. By now you should know different tuning methods and their general effect: modifying k_D and the zero $\frac{k_P}{k_D}$. Show a plot of your final run and show that it meets the design specifications. Don't forget to report your final controller values.

5.3 Performance For Different Input Signals

Using your *tuned* PD controller from part 5.2, what is the performance like for the following input signals? Make sure your simulation time is finite and ≤ 20 (in case something goes wrong).

- Pulse generator (50% pulse width, amplitude $\leq 7.5\,\text{cm}$, period $\geq 3\,\text{sec}$)
- Sine wave (amplitude $\leq 10\,\text{cm}$, frequency $\leq 1\,\text{Hz}$)
- Saw/Triangle wave (amplitude $\leq 5\,\text{cm}$, frequency $\leq 0.5\,\text{Hz}$)
- Exponentially-decaying sine wave (initial amplitude $\leq 15\,\text{cm}$, pick a reasonable time constant and frequency so that the decay is evident, but not too sudden or too 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. Discuss the performance of the controller for the different input signals. Which signals can the controller track easily? Why is this the case?

5.4 Performance For Limited Braking Ability

Suppose you are limited in your ability to slow the cart. How does limiting the negative voltage threshold to -3V affect the step response. What properties (rise time, maximum peak, settling time, etc.) are affected? Include plots of the voltage input and the position of the cart.

How does this affect the performance when tracking the sine wave input?