

Inverted Pendulum

Objectives

The objective of this lab is to experiment with the stabilization of an unstable system. The inverted pendulum problem is taken as an example and the animation program gives a feel for the challenges of manual control. A stabilizing linear controller is designed using root-locus techniques, and the controller is refined to enable stabilization of the inverted pendulum at any position on the track.

Introduction

The inverted pendulum system is a favorite experiment in control system labs. The highly unstable nature of the plant enables an impressive demonstration of the capabilities of feedback systems. The inverted pendulum is also considered a simplified representation of rockets flying into space.

Fig. 1 shows a diagram of the experiment. A cart rolls along a track, with its position x being controlled by a motor. A beam is attached to the cart so that it rotates freely at the point of contact with the cart. The angle of the beam with the vertical is denoted θ , and an objective is to keep the angle close to zero. Since the pendulum may be stabilized at any position on the track, a second objective is to specify the track position. However, recovery from non-zero beam angles may require significant movements of the cart along the track, and stabilization may be impossible if an insufficient range of motion remains.

A model of the inverted pendulum may be derived using standard techniques. A careful derivation (H. Khalil, *Nonlinear Systems*, 2nd edition, Prentice Hall, Upper Saddle River, NJ, 1996) shows that

$$(I + mL^2)\frac{d^2\theta}{dt^2} + mL \cos(\theta)\frac{d^2x}{dt^2} = mgL \sin(\theta) \quad (1)$$

where m is the mass of the beam, $2L$ is the length of the beam, $I = mL^2/3$ is the moment of inertia of the beam around its center of gravity, and g is the acceleration of gravity. For small angles θ , it follows that

$$\frac{4}{3}L\frac{d^2\theta}{dt^2} - g\theta = -\frac{d^2x}{dt^2} \quad (2)$$

and, the transfer function of the system is

$$\frac{\Theta(s)}{X(s)} = \frac{-s^2}{(4/3)Ls^2 - g} \quad (3)$$

The system has poles at $s = \pm\sqrt{3g/4L}$. The positive root is unstable, and instability worsens when the beam is short (as one might expect). Another tricky problem is that the

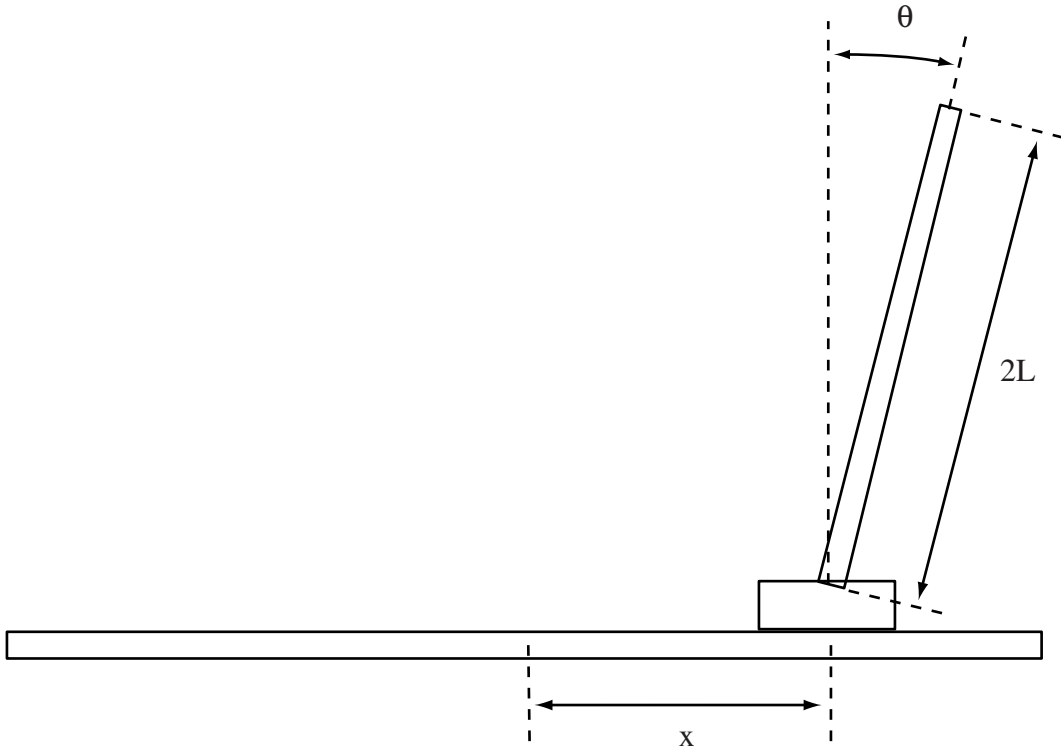


Figure 1: Inverted Pendulum System

system has two zeros at $s = 0$. This is due to the fact that an *acceleration* of the cart is required to impact the beam angle. On the other hand, this property makes stabilization possible for arbitrary cart positions.

More complex models assume that the control variable is the force applied to the cart, rather than its position. This assumption is more realistic in many cases, but makes manual control very difficult. Instead, the simulation program of this lab represents the delay in the motion of the cart by the response of a first-order system

$$X(s) = \frac{f}{s + f} X_{com}(s) \quad (4)$$

where x_{com} is the commanded cart position, x is the cart position, and $f > 0$. The overall transfer function of the system is then

$$\frac{\Theta(s)}{X_{com}(s)} = \frac{-fs^2}{((4/3)Ls^2 - g)(s + f)} \quad (5)$$

or

$$\frac{\Theta(s)}{X_{com}(s)} = \frac{bs^2}{(s + a)(s - a)(s + f)} \quad (6)$$

where $a = \sqrt{3g/4L}$, $b = -3f/4L$.

Manual control

The simulation program resides in the file *invpend.m*. The program assumes that $L = 1$ m, and that x is limited to $\pm 2L$. The beam angle is prevented from exceeding $\pm 30^\circ$. The value of f is 5, so that the cart responds relatively fast to commands. Even with rapid position control of the cart, however, the system is difficult to control with a joystick. The program has a “*cheat*” variable that allows you to make the problem easier. The variable reduces the magnitude of the gravity constant, and the choice *cheat* = 6 is preset in the program (it ’s as if you controlled the pendulum on the moon!).

Have fun trying to keep the beam balanced. If you feel comfortable with it, try to move the cart from the left line to the right line, while keeping the beam balanced. Describe in words the challenges of the control problem and your strategies for control. How close to 1 can you let the *cheat* variable be? For how long? You may be asked to demonstrate!

Automatic control

Stabilizing controller

First, you will design a controller to bring the beam angle to zero at the zero track location. Try to find a stabilizing controller using root-locus techniques, and noting that the plant transfer function is

$$P(s) = \frac{\Theta(s)}{X_{com}(s)} = \frac{bs^2}{(s+a)(s-a)(s+f)} \quad (7)$$

Explain why it is necessary to use an *unstable* controller, in order to have a chance at stabilizing the system.

Then, consider the second-order controller (with zero reference input)

$$C(s) = \frac{X_{com}(s)}{-\Theta(s)} = k \frac{(s+a)(s+f)}{(s-a)(s+c)} \quad (8)$$

Note that, with the cancellations, the closed-loop system has three poles. The controller has two parameters k and c to be adjusted. Show that it is possible to choose the parameters so that all three closed-loop poles are placed at $s = -a/2$. Compute the parameters required for the system and draw the root-locus for varying k . Verify that the closed-loop system is stable by use of the Nyquist criterion (function *nyquist* in Matlab). Plot the Bode plots and compute the gain and phase margins of the system (function *margin* in Matlab). Indicate how much the gain of the system can be increased or decreased while preserving stability.

The controller must be implemented in discrete-time, assuming a sampling rate of 20Hz. Check the help files for the Matlab functions *tf*, *c2d*, and *ssdata* to get information on how to do it (you may also look in the simulation code how the transfer function (3) was implemented).

Use the option 'zoh' for the discretization. Implement the controller transfer function in a file *invpendc.m*, with its initialization in a file *invpendcinit.m*. Test the response of the system by setting the initial value of the variable *theta* at 5 degrees (the initial value can be set at the beginning of the simulation program). Plot the responses of the beam angle and of the cart position resulting from this initial error. Check that the beam angle returns to zero within the expected time period and with the expected dynamics, given the pole locations.

Tracking controller

Modify the controller so that an arbitrary set-point of the cart position can be imposed. Replace $X_{com}(s) = -C(s)\Theta(s)$ by

$$X_{com}(s) = X_{ref}(s) - C(s)\Theta(s) \quad (9)$$

Note that this is an unusual configuration for a feedback system, because the reference command is applied to the plant input rather than the controller input. Interestingly, the plant output *cannot* be set to an arbitrary position in this problem, but the plant input can. Generate steps of position reference alternating between $1m$ and $-1m$ every 20 seconds, and plot the responses of x , x_{com} , and θ for 60 seconds. Let the initial beam angle be zero.

You should observe an overshoot in the response of the cart. Compute the transfer function from $X_{ref}(s)$ to $X(s)$ and observe that there is a zero at a frequency lower than the closed-loop poles. This problem may be addressed by using a pre-filter $C_F(s)$ so that

$$X_{com}(s) = C_F(s)X_{ref}(s) - C(s)\Theta(s) \quad (10)$$

where $C_F(s)$ is a first-order filter with unity DC gain and a pole located at the same value as the low-frequency zero. Plot the responses of x , x_{com} , and θ for 60 seconds, with steps of position reference alternating between $1m$ and $-1m$ every 20 seconds. Explain the fact that the variable x_{com} initially moves in the direction *opposite* to the step applied. Can this effect be eliminated by prefiltering of the reference command (as was done for the overshoot)? Test the controller for the *cheat* variable equal to 6, and again for the variable equal to 1. Show your code and demonstrate the real-time operation to the TA.

Report at a glance

Be sure to include:

- Description of challenges and strategies for manual control.
- Design and evaluation of the stabilizing controller with the variable *cheat=6*. Include controller parameters, root-locus, Nyquist criterion, gain and phase margins, and responses to initial beam angle.

- Design and evaluation of the controller for set-point tracking with the variable $cheat=6$. Plots of responses of x , x_{com} , and θ to step changes in set-point of the cart position.
- Design and evaluation of the controller for set-point tracking with the variable $cheat=1$.
- Observations and comments.
- Listing of *invpendc.m* and *invpendcinit.m*.