University of Toronto

Department of Electrical and Computer Engineering

ECE557F – Systems Control

Experiment 3

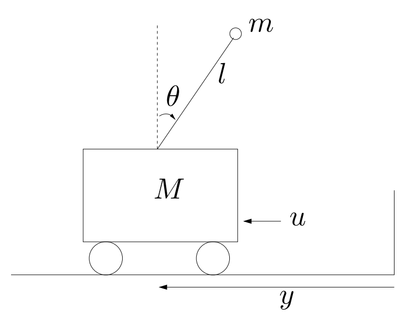
CONTROL OF AN INVERTED PENDULUM ON A CART

1 Purpose

The purpose of this experiment is to familiarize you with modelling nonlinear systems using SIMULINK, performing linearization, designing and evaluating different control laws based on pole placement and optimal control, designing observers, and finally verifying your design.

2 Introduction

An inverted pendulum on a cart is often used as an illustrative model for control system design. The following figure provides an illustration of the physical system.



The differential equations governing the dynamics of the inverted pendulum can be shown to be given by

$$\begin{aligned}\ddot{y} &= \frac{u - ml\dot{\theta}^2 \sin\theta + mg\cos\theta\sin\theta}{M + m\sin^2\theta} \\ \ddot{\theta} &= \frac{(M+m)g\sin\theta + u\cos\theta - ml\dot{\theta}^2\sin\theta\cos\theta}{l(M + m\sin^2\theta)}.\end{aligned}$$

It is well-known from one's common experience with trying to balance a broom that the inverted pendulum on a cart is a difficult system to control. The design specifications are:

(a) The inverted pendulum is balanced at its vertical position;

(b) The cart is asymptotically moved back to the origin and remains there.

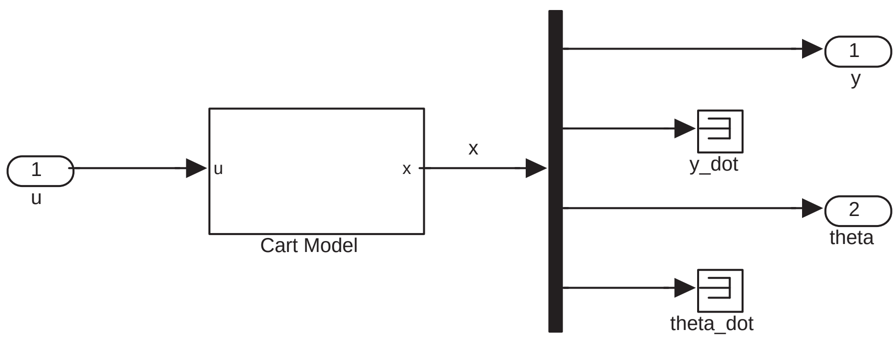
(c) Good transient response is obtained. This is deliberately vague, and better transient response is often obtained by tuning control parameters.

3 Preparation

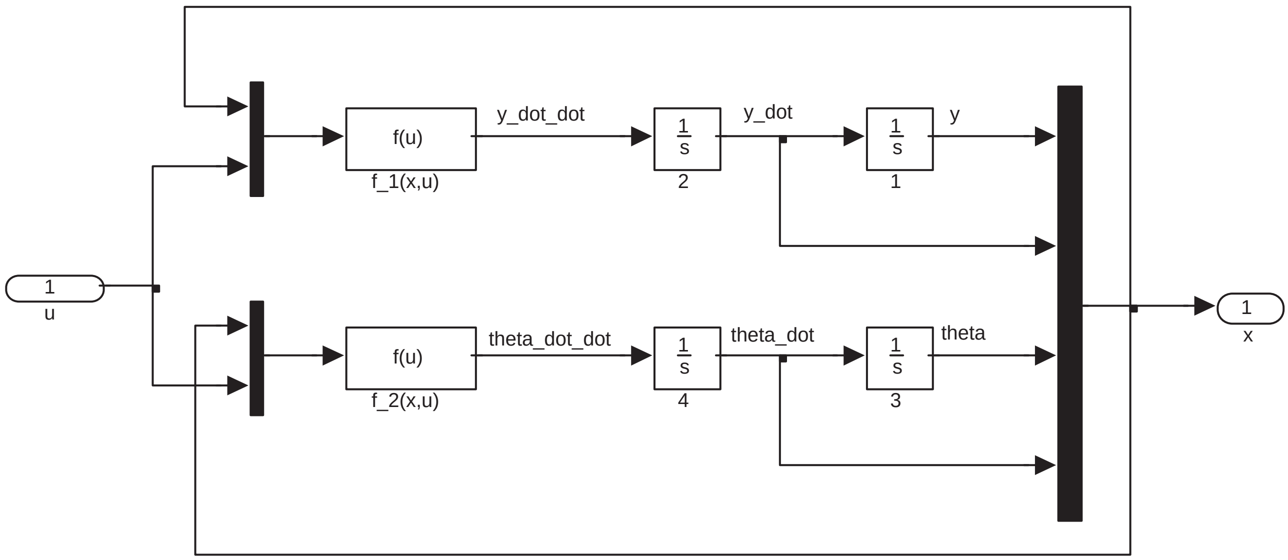
3.1 Building the SIMULINK Model

We shall first build a SIMULINK model of the nonlinear system, and then linearize it to get a linear approximation.

1. Take the state vector $x = [y\ \dot{y}\ \theta\ \dot{\theta}]$. Write down the nonlinear differential equation for x. You should have 2 nonlinear functions on the right hand side of the differential equation.
2. Linearize the equations of motion at the equilibrium point $x = 0$, $u = 0$ analytically (by hand) in terms of the parameters M, m, l, and g.
3. Set up a SIMULINK model in the form described by the following figure:



where the input u to the nonlinear state equation is a force input to the cart, and the outputs y and θ are the position of the cart and the angle of the pendulum with respect to the vertical, respectively. ion. The block "Cart Model" corresponds to the nonlinear state equation and is a subsystem of the SIMULINK model. The cart model block can be built in the form described by the following figure:



The Fcn blocks are used to specify the nonlinear functions in the system. Based on these suggestions, set up the complete SIMULINK model corresponding to the inverted pendulum on a cart system. Save your model in SIMULINK as, say, `invertpen.slx`.

4. For one of the real inverted pendulums in the lab, we have the following physical parameters:

   - $M = 1.0731$ Kg
   - $m = 0.2300$ Kg
   - $l = 0.3302$ m
   - $g = 9.8\ m/s^2$

   Before you carry out linearization, it is useful to check what SIMULINK interprets as the state vector. This can be done using the command

   $$[\texttt{sizes}, \texttt{x0}, \texttt{states}] = \texttt{invertpen}([], [], [], \texttt{0})$$

   Note that SIMULINK's state vector may be different from the state vector at the outport. Use the SIMULINK command `linmod` to linearize your system at the equilibrium point $x = [0\ 0\ 0\ 0]^T$, $u = 0$. This yields the system matrices A, B, C, with $D = 0$.

5. Plug in the real parameter values of the previous step in the linear model you derived in Step 2. Compare your model to the model obtained using SIMULINK. You should get the same result, up to order of the state variables.

6. Consider the second nonlinear differential equation for $\ddot{\theta}$. Take the output as θ and the input as u. Linearize this new model analytically at the equilibrium point $\theta = \dot{\theta} = 0$ and $u = 0$ in terms of the parameters of the system. Next, derive the open-loop transfer function and determine its poles and zeros. Using these results, explain why the inverted pendulum on a cart is difficult to control, especially if you use classical design methods such as root locus.

3.2 Control Design

We now use the linearized model to design the controller. We shall use both pole placement and optimal control as our design methodology. All steps should be done using MATLAB or SIMULINK.

1. Assume that the entire state of the cart-pendulum system is observed. Design a state feedback law $u = -Kx$ so that the closed loop poles are located at $\{-1, -2, -3 -4\}$. Call this controller SF-controller I.

   WARNING: The Matlab command `place(A,B,poles)` places the poles of the matrix $A - BK$, **not** $A + BK$ as we did in class. So while in class we use the state feedback $u = Kx$, here you'll use $u = -Kx$. Note that Matlab's sign convention in `place` will affect your observer design as well. To make sure you don't have a sign problem in your observer gain, check the poles of the observer error matrix and verify that they have negative real part.

2. Repeat the design but with the closed loop poles at $\{-0.8, -1.2 \pm i, -2\}$. Call this controller SF-controller II.

3. Now assume that only the cart position y and the pendulum angle θ are measured. We use a full-order observer to estimate the state. Determine the full-order observer gain L so that observer poles are located at $\{-8, -9 \pm i, -10\}$.

4. Next you will design an optimal controller and explore the effect of the cost function parameters Q and R on the closed-loop response. See the Appendix for background on optimal control design. Let

$$Q = \begin{bmatrix} q_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & q_2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}.$$

   The parameters q_1 and q_2 influence the speed of response of the cart position and the pendulum position, respectively. Using the MATLAB command `lqr`, design nine different optimal controllers for the following parameter values:

   - $q_2 = 5$; $R = 0.5$; $q_1 = 0.005, 0.05, 0.1$.
   - $q_1 = 0.05$; $R = 0.5$; $q_2 = 0.001, 0.1, 100$.
   - $q_1 = 0.05$; $q_2 = 5$; $R = 0.1, 1, 10$.

   Warning: the values of Q and R given here correspond to our convention for the order of the state vector. You may have to rearrange these according to the state vector you got with MATLAB.

Write a MATLAB script which carries out all of the design steps above automatically. The script should include the definition of the parameters of the SIMULINK model at the top. Bring to the lab your hand calculations, your MATLAB script, and the files for the nonlinear SIMULINK model to show to the TA. You will also need these files to complete the experiment.

4 Experiment

4.1 Linear Verification of Control Design

In this section you will first verify the response of the closed-loop system using the linear model. All steps should be done using MATLAB or SIMULINK.

1. Suppose the initial cart position $y(0) = 0.5$, the initial pendulum angle $\theta(0) = 0.175$ (corresponding to approximately $10°$), and the initial velocities $\dot{y}(0) = 0.1$, $\dot{\theta}(0) = 0$. Call this set of initital conditions IC. Simulate the response of the system under SF-controller I using MATLAB, and plot the response of the various state components. Repeat the simulation for SF-controller II.

2. Combine SF-controller I and the observer to give the output feedback control law $u = -K\hat{x}$. Call this OF-controller I. Simulate the response of the pendulum under the same initial conditions IC as before. You may set the initial conditions of the observer to zero. Plot your results on 4 plots, each containing one state component and its observer estimate.

3. Repeat for OF-controller II, which is the combination of SF-controller II and the observer.

4. Simulate the response of the system starting from initial conditions IC for each of the nine optimal controllers and observe the effect of varying each of the three parameters. Produce four plots for each of the state components, with each plot containing the response for the following three optimal controllers:

   - $q_1 = 0.1$, $q_2 = 5$, $R = 0.5$.
   - $q_1 = 0.05$, $q_2 = 100$, $R = 0.5$.
   - $q_1 = 0.05$, $q_2 = 5$, $R = 10$.

5. Repeat the previous step, but now combine the optimal controller with the full-state observer.

4.2 Nonlinear Verification of Control Design

Having carried out a design using a linear approximation of the nonlinear system, you must verify that the design is satisfactory for the nonlinear system.

1. Add SF-controller II to your SIMULINK diagram. Simulate the response of true nonlinear continuous time system under initial conditions IC using SIMULINK and view the state response on the SIMULINK scope. Is the response satisfactory? Is it different from the linear approximation?

2. Add also the observer to your SIMULINK diagram and implement OF-controller II. Repeat the simulation. Again record the system state response.

3. Now increase the initial pendulum angle to $30°$, still using OF-controller II. Are the design specs still achieved? Is is different from the linear approximation? If the design specs are still achieved, can you increase the initial angle even further?

4. Add the optimal controller for the cost function parameters $q_1 = 0.05$, $q_2 = 100$, $R = 0.5$ to your SIMULINK diagram. Simulate the response of the true nonlinear system with initial conditions IC using SIMULINK and view the state response on the SIMULINK scope. Is the response satisfactory? Is it different from the linear approximation?

5 Report

For the report, follow the instructions in the "report template" which will be announced during the lab.

A Optimal Control Design

The purpose of this appendix is to give you an introduction to optimal control design. The steps of the procedure will be derived and further explained in lecture.

You have learned that if a linear system is controllable, then the poles of the closed-loop system can be arbitrarily assigned. But we have not discussed *where* to place the poles for good performance of the closed-loop system. In a third year control course, you would have learned about techniques for placing the poles of a second-order system for good transient response. If the system is of higher order, then this method will not work. Optimal control gives a way to assign the poles based on two performance criteria: speed of response v.s. control effort. These performance criteria are specified using a cost function.

We are given the LTI system

$$\dot{x} = Ax + Bu\,,$$

and we are given a cost function

$$J = \int_0^\infty \left(x^T Q x + u^T R u\right) dt\,,$$

where $R \in \mathbb{R}^{m\times m}$ is a symmetric positive definite matrix and $Q \in \mathbb{R}^{n\times n}$ is a symmetric, positive semidefinite matrix. For the purposes of this lab, these are diagonal matrices with positive constants on the diagonal. The matrix Q penalizes the state x. By adjusting the size of the diagonal elements of Q we can penalize the different components of x. The matrix R penalizes the control u in the same way. Thus, a Q matrix with large terms on the diagonal means a high penality on x, which means the poles will be placed so that the state goes to zero quickly. Conversely, an R matrix with large terms on the diagonal means a high penality on u, which means the poles will be placed so that the input remains small. Notice you may not chose $R = 0$ otherwise the control could be infinite. Finally, the absolute values of the elements of R and Q are less important that the relative values between them.

The procedure to solve for the optimal control u which minimizes the cost function J while stabilizing the system begins with solving the *algebraic Riccati equation*

$$A^T P + PA - PBR^{-1}B^T P + Q = 0\,,$$

for the unknown symmetric, positive semi-definite matrix $P \in \mathbb{R}^{n \times n}$. Once this equation is solved for P the optimal control is the feedback

$$u = -R^{-1} B^T P x \,.$$

All of this can be done with the single MATLAB command `lqr`, which computes the state feedback gain $K = R^{-1} B^T P$.