March 19th, 2017

Robotics and Control Laboratory
ECSE 493 W2017
Lab Assignment 4
The Inverted Pendulum
Due April 10th by 4pm (last day of our class)
Please upload your assignment to the class webpage

The purpose of this experiment is to control an inverted pendulum on a cart using full state feedback. Please add the necessary precautions by adding saturating elements and stopping the system when the angular deviation of the pendulum exceeds 10 degrees (0.175 radians).

Full State Feedback.

Up to know, we've specified desired damping or steady-state error and designed controllers to meet these specifications. The controllers work by moving one or more dominant poles of the system. In full state feedback, we will specify the location of all the closed loop poles of the system. One condition for full state feedback is to have access to the full state, i.e. we need access to the outputs of all the sensors measuring the state. This is usually not possible in practice, but also not necessary as what we need is enough sensors to specify a system that models the real one.

The system is described by

$$\dot{x} = Ax + Bu,$$
$$u = -Kx$$
$$X = \begin{bmatrix} x_c \\ \theta_p \\ \dot{x}_c \\ \dot{\theta}_p \end{bmatrix}$$

where K=$[K_1,K_2,K_3,K_4]$. We need to determine K.
Substituting u into the state equation yields:

$$\dot{x} = (A - BK)x$$

The characteristic equation of this system is:

$$\det[sI - (A - BK)]$$

which yields an n^{th} order polynomial in s containing the gains K. You can then proceed to specify the poles of the system, which is equivalent to picking the gains K. The gains (all the K_i) can be obtained in a variety of ways. There are special systems that make finding the gains easier. These systems have the *controllable canonical form.* Briefly, given a system described by A and B as in equation (2), we can transform the system into control canonical form, find all the K_i 's for the system in control form, and transform back to the original state to map the gains from the control form to the original state *(see the lecture notes)*. Ackermann's formula does all this for you. Enter "*doc acker*" to learn more about what this function requires. To use *acker,* you have to specify the location of the poles. This is an iterative process, proceeding by trial and error. Good design principles that will help you choose a good starting location for your poles are below:

<u>Good Design Principles for choosing poles:</u>

1) You need to consider how far open loop poles are moved by feedback (see Matlab command *pzmap*).
   a) Poles near zero require large gains to move (approach noncontrollability).
2) Avoid:
   a) Large increases in the bandwidth,
   b) Placing poles near zero.
3) Try to fix undesirable aspects of open-loop response.

One way to adhere to these principles is to choose a pair of complex poles that are dominant and choose the remaining poles to be fast and damped. Transients due to fast poles should decay long before transients due to dominant poles. (Dominant poles are really just the poles of a second order system. These are the poles that specify damping, rise time and so forth. Therefore, this suggests that we are using 2nd order systems to understand our n^{th} order system)

<u>Linear Quadratic Regulators</u>

Using full state feedback, the control problem is posed as a cost to be minimized. The cost for an LQ regulator is:

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

the goal is to choose the appropriate symmetric matrices Q and R. The integral penalizes departure from zero over time. Note that including the U term is necessary as a very large input can drive the system towards equilibrium very rapidly, only to overshoot.
Again, choosing appropriate Q and R is an iterative process. You need knowledge of the system. It is suggested that

$$Q = \begin{bmatrix} Q_{11} & 0 & 0 & 0 \\ 0 & Q_{22} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

so that Q_{33} and Q_{44} are set to 0 (these coefficients affect $\dot{x}$ and $\dot{\alpha}$). By setting these coefficients to 0, we're not penalizing the system for moving. There are difficulties when dealing with velocities. There are many rules to ensure proper design but ultimately, only practice will give you insight into these techniques. In Matlab, the command '*lqr'* will return gains for your control law.

For your Report:

-Derive the pendulum equation of motion.
-For all cases explain the design procedure you've followed and justify your choice of values.
- Explain why you chose the pole locations you did when you designed the full state feedback.
-Comment on the values of Q and R and show where your closed loop poles are located (use PZmap). What happens when you change Q and/or R?
-Once you've stabilized the inverted pendulum to the origin, experiment with the controllers by changing the individual gains and observing what happens to the system.

What happens to the controlled system if the modeled parameters change, for example the mass of the pendulum?

Please make sure that all your plots are legible. Please make the report brief and to the point.

Lab Presentation:

You are asked to informally present your design and a stable response in the lab for approximately 10 minutes to Ali and Prof Musallam. No need to have slides (although you can if you want to), but each member on the team will do one half of the presentation. **Presentations should be scheduled the week of April 3rd.** What should be in your presentation are:

a) An introduction to the control problem with a formulation of the problem
b) The methodology to design the controller
c) The implementation on the real-time system
d) A demo of a stabilization experiment where the cart controls the pendulum in the vertical position.
e) A brief conclusion