EE 16B Designing Information Devices and Systems II
Fall 2015 Homework 9

This homework is due November 12, 2015, at 5PM.

1. Eigenvalues and State Space Systems

(a) Let's assume we have a system whose dynamics are described by

$$\begin{bmatrix} x_0[k+1] \\ x_1[k+1] \end{bmatrix} = A \begin{bmatrix} x_0[k] \\ x_1[k] \end{bmatrix}$$

$$A = \begin{bmatrix} 0.7 & 0.2 \\ 0.8 & 0.35 \end{bmatrix}$$

Find the eigenvalues of the transition matrix A.

(b) Use iPython to simulate the system from part (a) with an initial condition of:

$$\begin{bmatrix} x_0[0] \\ x_1[0] \end{bmatrix} = \begin{bmatrix} 1 \\ 0.2 \end{bmatrix}$$

Plot $x_0[k]$ and $x_1[k]$ vs. k for $k = 0$ to 20.

(c) Now let's assume the same type of system as part (a), but with a new transition matrix

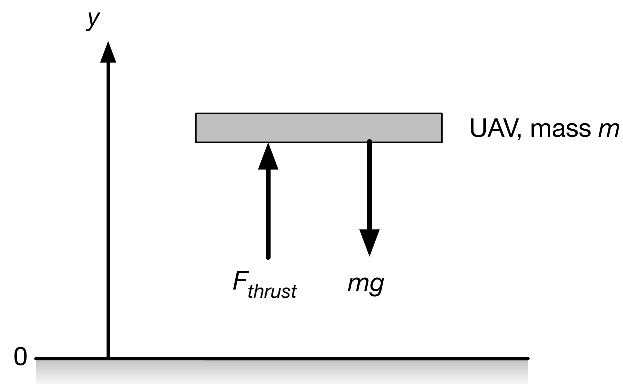
$$A' = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

Find the eigenvalues of this new transition matrix A'.

(d) Use iPython to simulate the system from part (c) with the same initial condition as part (b). Plot $x_0[k]$ and $x_1[k]$ vs. k for $k = 0$ to 20.

2. Modeling the Open-Loop Height Dynamics of a UAV

In this and the remaining problems on this homework, we will examine a simple model of a "UAV" (e.g., a quadrotor). For the sake of simplicity, we will assume that the UAV has one "thruster" that applies force directly in the positive y direction (i.e., accelerates the UAV upwards), and for later problems, that this thruster can be controlled by some input voltage. (In this way, we will be able to translate a measured velocity into a voltage and then produce another voltage that will control the thruster.) Of course, the UAV experiences gravity as well. A simple diagram capturing the situation is shown below.



After examining this situation and the sample rates of the ADC/DAC embedded into the microcontroller you will be using to control this quadrotor in later problems, you find that the discrete-time state space model for this situation can be written as:

$$\begin{bmatrix} y[k+1] \\ v[k+1] \end{bmatrix} = A \begin{bmatrix} y[k] \\ v[k] \end{bmatrix} + Bu_{eff}[k]$$

$$\begin{bmatrix} y[k+1] \\ v[k+1] \end{bmatrix} = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y[k] \\ v[k] \end{bmatrix} + \begin{bmatrix} T_s^2/2m \\ T_s/m \end{bmatrix} (F_{thrust}[k] - mg)$$

where y is the vertical position, v is the vertical velocity, $T_s = 1\text{ms}$ is the sample period, $m = 0.5\text{kg}$ is the quadrotor's mass, and $g = 9.8\text{m/s}^2$ is the acceleration due to gravity.

(a) Find the eigenvalues of the A matrix for this system.

(b) Assuming that the UAV initially starts at $y[0] = 1\text{m}$ with $v[0] = 0$ m/s, what are $v[2]$ and $y[2]$ if an F_{thrust} of 10N is applied for 2ms (so $F_{thrust}[0] = F_{thrust}[1] = 10\text{N}$, $F_{thrust}[k] = 0$ for $k \neq 0, 1$)?

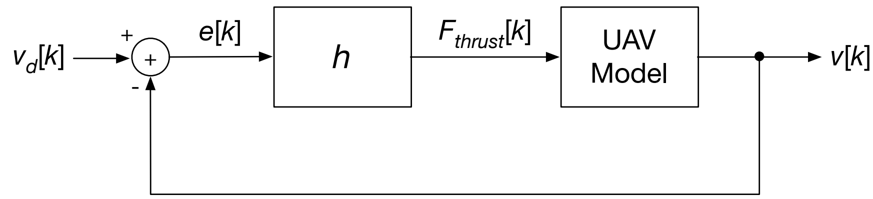
(c) For the same initial conditions as part (b), derive an expression for $v[k]$ and $y[k]$ (for any $k > 0$) if the applied F_{thrust} is a step (i.e., $F_{thrust}[k] = 10\text{N}$ for $k \geq 0$, $F_{thrust}[k] = 0$ for $k < 0$).

(d) For the same step input as part (c) and initial conditions from (b), but for a step height of $F_{thrust} = 4.9\text{N}$ (instead of 10N), derive a new expression for $v[k]$ and $y[k]$.

(e) Now simulate the open-loop feedback system - see the iPython notebook.

3. Closed-Loop Control of the UAV

Now let's look at what happens when we use a closed-loop feedback system to try and get the UAV to move at a specific desired velocity. In particular, let's assume that we have a sensor that can measure the velocity $v[k]$ (but not the height $y[k]$); the feedback loop will then be constructed by subtracting the measured $v[k]$ from a desired (input) velocity $v_d[k]$, scaling this error by a gain h (whose units are N / (m/s)) to result in an input force F_{thrust}. A block diagram representation of this feedback system is shown below.



(a) Re-derive a state-space model that you could use to solve for $y[k]$ and $v[k]$ as a function of $v_d[k]$ and the initial conditions. Your model should have the form shown below:

$$\begin{bmatrix} y[k+1] \\ v[k+1] \end{bmatrix} = A_{CL} \begin{bmatrix} y[k] \\ v[k] \end{bmatrix} + B_{CL} \begin{bmatrix} v_d[k] \\ mg \end{bmatrix}$$

In other words, you should find expressions for A_{CL} and B_{CL}.

Hint: First derive B_{CL} as if $v_d[k]$ were the only input to the closed-loop system, then modify it to account for the mg term.

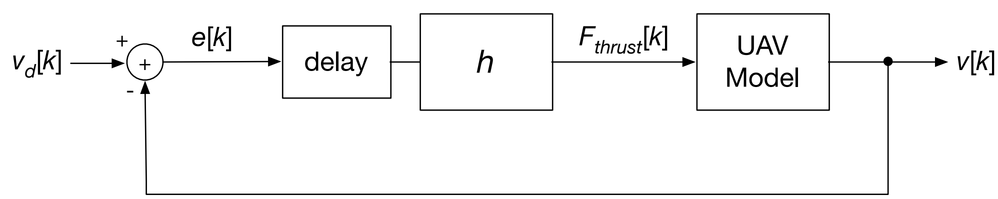
(b) Derive the eigenvalues of A_{CL} as a function of the feedback gain h.

(c) Now simulate the closed-loop feedback system - see the iPython notebook.

4. Closed-Loop Control with Delay

As we saw in last week's homework, extra delay in a feedback control loop can cause some unexpected and potentially very undesirable things to happen. In this problem we'll briefly examine how to model the addition of delay in our state-space formulation.

In particular, let's assume that in addition to simply applying a gain h, our controller introduces a delay of 1 sample (i.e., 1ms) between the input and output of the gain block. This situation is depicted below.



Using a similar form as in Problem 3(a), derive a state space model for this new closed-loop system.

Hint: You may need to define a new state variable to take into account the effect of the delay.