KUNGL TEKNISKA HÖGSKOLAN

Royal Institute of Technology
Department of Mathematics
Division of Optimization and Systems Theory
Anders Blomqvist

Computer Exercise 2
Mathematical Systems Theory, 5B1742

Fall 2002

Satisfactory completion of this computer exercise gives two (2) bonus credits for this year's finals. The examination of the computer exercises is *oral and written*. Firstly the results should be presented in a written report (use a word processor) and secondly each student should be prepared to answer question about the exercise when it is handed back. Hand in the report at the lastest during the exercise session on October 14 and pick it up after agreement with Anders Blomqvist. Do attach your commented MATLAB code to your report! Cooperation in groups of not more than two students is allowed.
The necessary MATLAB m-files can be found at `www.math.kth.se/~andersb/teaching.html`
The intention of this exercise is to exemplify how easy it is to apply the theory of the course using a computer. We will use the "Control System Toolbox" in MATLAB, which will save us lots of programming. Names written with bold font are command names in MATLAB.
Write **help control** to get a list of available functions in the "Control System Toolbox" or use the help browser.

1. Observers

In this exercise we will construct a full-order observer to the inverted pendulum system and use it to make an *output* feedback stabilization of the system.

(1) Construct an observer with stable poles to the inverted pendulum system of Lab1.
(2) Combine one of the stabilizing feedbacks (or a new better one of your choice) of Lab 1 with the observer above (as in the Figure on page 62 in the lecture notes ("Kurskompendiet")). The combination is a new linear system with twice as many states (x and $\hat{x}$), four additional outputs (the estimation errors, making the total 6) $y = (x_1, x_3, x_1 - \hat{x}_1, x_2 - \hat{x}_2, x_3 - \hat{x}_3, x_4 - \hat{x}_4)$ and one input (the original one). Construct the new (bigger) system $A_{new}, B_{new}, C_{new}, D_{new}$. Is the new system minimal?
(3) Simulate a step response with an initial estimation error (of your choice) $x(0) \neq \hat{x}(0)$ using **lsim**. Do the above simulation for two cases, one where the poles of the observer is closer to the origin than the poles of the feedback and one the other way around. Which is best and why? *Hint: One way of doing this is to choose two sets of poles with clearly different bahaviors. First give the observer one set and the closed-loop system the other set. Then swap the sets and notice the difference.*

2. L-Q Optimal Control.

In this section we use the linearized pendulum equations from Lab 1.

2.1. Infinite time horizon.
Consider the following problem

$$\min \int_0^\infty x^T Q x + u^T R u \, dt$$

when

$$\dot{x} = Ax + Bu, \quad x(0) = x_0.$$

That is driving the system towards the equilibrium over infinite time while keeping controls and errors small.

Note that *Remark 8.1.1* explains how this translates to the problem formulation of chapter 8 in the course book. Remind yourself what the compatible dimensions of the matrices are.

(1) Choose your own Q and R and calculate the optimal control (**are** or **care** might be useful).
(2) Note that the resulting control is a state feedback. Calculate the closed loop poles of the system for some different R and Q. How are the poles affected by making R big and Q small? The other way around? (Compare sufficiently large difference of R and Q to see the difference). Note that this design method is an alternative to explicit pole placement.

Note that there is also an LQ theory for the finite time horizon case, with or without fixed terminal state.

3. Kalman filter for parameter estimation

The Kalman filter is a widely used algorithm in control and signal processing. Applications are for example state estimation, sensor fusion and target tracking. Examples of more advanced target tracking are for example a camera tracking moving humans and radars tracking airplanes. The case we will study is tracking of a piecewise constant but unknown signal. Let x_k be a scalar that is a piecewise constant function of time where we do not know the values that x_k can assume and not the times when it changes. Let us say that $|x_k| \leq 1$ though (this is for all lab groups to get comparable results). The measurements that we have are noisy measurements y_k of x_k defined by

$$y_k = x_k + dw_k \tag{1}$$

where w_k is $N(0,1)$ and $d = 0.3$. The task is to design a Kalman filter for estimating x_k given y_k. The following model puts the problem in a Kalman filter framework:

$$\begin{cases} x_{k+1} &= x_k + bv_k \\ y_k &= x_k + dw_k \end{cases} \tag{2}$$

where we have modeled x_k as a random walk. Now, b is a tuning parameter that you have to choose and it is part of the lab to study how this choice affects the filter performance. Remember that (2) is a *model* and that the true x_k and thus the measurements y_k must be generated off-line where you choose the piecewise constant x_k in some way you find suitable.

3.1. Fixed b.

(1) Implement a Kalman filter that estimates x_k (Theorem 9.1.10).
(2) Feed the filter with observations as in (1), where x_k is piecewise constant (choosen by you) and w_k is random noise (use **randn**). Study the following aspects experimentally.
    (a) How does P_0 affect performance? (study the initial behavior).
    (b) How does b affect performance?
    (c) There seems to be a contradiction between between fast convergence to the true x_k and smooth estimates. Describe this and explain why.

3.2. Time-varying b. Consider now the situation where we know *when* x_k changes value. Construct a time-varying b_k that resolves the conflict between convergence speed and smoothness. Use the insight obtained when tuning b_k (i.e. when do we want smoothness and when do we need higher convergence speed?).

Good Luck! Don't hesitate to ask (by email or phone) if anything is unclear.