AE353 Homework #6:
Optimal Observer Design

with an AWESOME extra credit problem

(due at the beginning of class on Friday, March 20)

1. You have seen that the motion of a robot arm with one revolute joint can be described in state-space form as

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & -1/5 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1/5 \end{bmatrix} u$$
$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} x$$

where the state elements are the angle (x_1) and angular velocity (x_2) of the joint, the input u is a torque applied to the arm at the joint, and the output y is a measurement of the angle.

(a) Design an optimal controller of the form

$$u = -K\hat{x} + k_{\text{reference}} r$$

for weights $Q_c = I$ and $R_c = \rho$, for *your* choice of ρ.

(b) Design an optimal observer of the form

$$\dot{\widehat{x}} = A\widehat{x} + Bu - L(C\widehat{x} - y)$$

for weights $Q_o = \gamma$ and $R_o = I$, for *your* choice of γ.

(c) Express the closed-loop system in the form

$$\frac{d}{dt}\begin{bmatrix} x \\ \widehat{x} \end{bmatrix} = A_{\text{cl}} \begin{bmatrix} x \\ \widehat{x} \end{bmatrix} + B_{\text{cl}} r$$

for some matrices A_{cl} and B_{cl}. Compute the eigenvalues of this system.

(d) Simulate the closed-loop system in response to a reference signal $r(t) = \pi/2$ for all $t \geq 0$ for the initial condition and initial state estimate

$$x(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad \widehat{x}(0) = \begin{bmatrix} -1 \\ 2 \end{bmatrix}$$

over a suitable time horizon.

(e) Repeat (a)-(d) as necessary, choosing ρ and γ so that the closed-loop system exhibits a time-to-peak of less than 0.5 seconds and a peak overshoot of less than 10%. Briefly explain *your* design process. Submit whatever plots *you* feel are necessary to justify this process and to show that *your* final design satisfies the performance specification.

(f) Implement your controller and observer by modifying the function `ControlLoop` in the MATLAB script `hw6prob01.m`. Submit only your `ControlLoop` function (this should be the only part of the code that you change) and a snapshot of the figure after the simulation has ended.

NOTE: In this problem, if you like and if it helps your design process, you are also welcome to play around with the diagonal entries of Q_c and R_o.

2. You have seen that the rotational motion of an axisymmetric spacecraft about its yaw and roll axes can be described in state-space form as

$$\dot{x} = \begin{bmatrix} 0 & \lambda \\ -\lambda & 0 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u$$
$$y = \begin{bmatrix} 0 & 1 \end{bmatrix} x$$

where the state elements x_1 and x_2 are the angular velocities about yaw and roll axes, the input u is an applied torque, and the parameter $\lambda = 9$ is the relative spin rate.

(a) Design an optimal controller of the form

$$u = -K\hat{x}$$

for weights $Q_c = I$ and $R_c = \rho$, for *your* choice of ρ.

(b) Design an optimal observer of the form

$$\dot{\widehat{x}} = A\widehat{x} + Bu - L(C\widehat{x} - y)$$

for weights $Q_o = \gamma$ and $R_o = I$, for *your* choice of γ.

(c) Express the closed-loop system in the form

$$\frac{d}{dt}\begin{bmatrix} x \\ \widehat{x} \end{bmatrix} = A_{\text{cl}} \begin{bmatrix} x \\ \widehat{x} \end{bmatrix}$$

for some matrix A_{cl}. Compute the eigenvalues of this system.

(d) Simulate the closed-loop system for the initial condition and initial state estimate

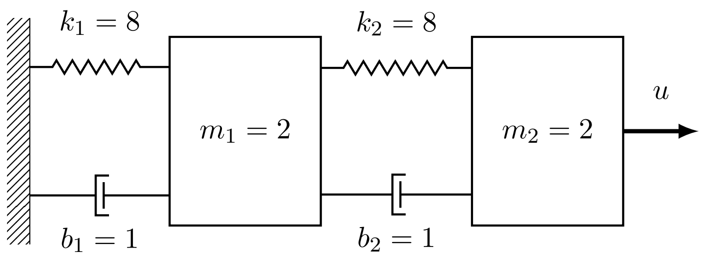
$$x(0) = \begin{bmatrix} 5 \\ -5 \end{bmatrix} \qquad \widehat{x}(0) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

over a suitable time horizon.

(e) Repeat (a)-(d) as necessary, choosing ρ and γ so that the closed-loop system exhibits a 2%-settling time of less than 4 seconds and so that $|u(t)|$ never exceeds 20. (In the context of a unit step response, "2%-settling time" would mean the minimum time t_s for which $|1 - y(t)| < 0.02$ for all $t \geq t_s$. In the context of a response to non-zero initial conditions, we mean the minimum time t_s for which $|y(t)| < 0.02$ for all $t \geq t_s$.) Briefly explain *your* design process. Submit whatever plots *you* feel are necessary to justify this process and to show that *your* final design satisfies the performance specification.

(f) Implement your controller and observer by modifying the function `ControlLoop` in the MATLAB script `hw6prob02.m`. You may find that performance is not what you expect, and that some design iteration is required. Here is the reason why. So far, you have assumed that the spacecraft is axisymmetric. A real spacecraft is never quite axisymmetric. So, the actual ordinary differential equations describing the motion of the spacecraft are nonlinear (see HW1, Problem 3). The script `hw6prob02.m` integrates these nonlinear ODEs—in other words, it simulates the "real" closed-loop system given the controller and observer you designed for the axisymmetric model of this system. Again, repeat (a)-(d) as necessary, choosing ρ and γ so that the "real" closed-loop system has a 2%-settling time of less than 4 seconds and so that $|u(t)|$ never exceeds 20. Briefly explain *your* design process, submitting whatever plots *you* feel are necessary to justify it. Also submit your `ControlLoop` function and a snapshot of the figure for your final design.

NOTE: In this problem, if you like and if it helps your design process, you are also welcome to play around with the diagonal entries of Q_c and R_o.



3. You have seen that the spring-mass-damper system shown above can be described in state-space form as

$$\dot{x} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -8 & 4 & -1 & 0.5 \\ 4 & -4 & 0.5 & -0.5 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0.5 \end{bmatrix} u$$
$$y = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} x$$

where x_1 and x_2 are the absolute displacements of each mass from their equilibrium positions, x_3 and x_4 are the corresponding velocities of each mass, u is the applied force, and y is a measurement of x_2.

(a) Design an optimal controller of the form

$$u = -K\hat{x}$$

for weights $Q_c = I$ and $R_c = \rho$, for *your* choice of ρ.

(b) Design an optimal observer of the form

$$\dot{\widehat{x}} = A\widehat{x} + Bu - L(C\widehat{x} - y)$$

for weights $Q_o = \gamma$ and $R_o = I$, for *your* choice of γ.

(c) Express the closed-loop system in the form

$$\frac{d}{dt}\begin{bmatrix} x \\ \widehat{x} \end{bmatrix} = A_{\text{cl}} \begin{bmatrix} x \\ \widehat{x} \end{bmatrix}$$

for some matrix A_{cl}. Compute the eigenvalues of this system.

(d) Simulate the closed-loop system for the initial condition and initial state estimate

$$x(0) = \begin{bmatrix} 1 \\ 3 \\ 0 \\ 0 \end{bmatrix} \qquad \widehat{x}(0) = \begin{bmatrix} 1.1 \\ 2.9 \\ -0.1 \\ 0.1 \end{bmatrix}$$

over a suitable time horizon.

(e) Repeat (a)-(d) as necessary, choosing ρ and γ so that the closed-loop system exhibits a 5%-settling time of less than 5 seconds. Briefly explain *your* design process. Submit whatever plots *you* feel are necessary to justify this process and to show that *your* final design satisfies the performance specification. (You need not submit many plots.)

(f) Implement your controller and observer by modifying the function `ControlLoop` in the MATLAB script `hw6prob03.m`. You should find that performance is not what you expect, and that some design iteration is required. Here is the reason why. The script `hw6prob03.m` considers the case in which there is a third mass, connected to the second mass with a third spring and damper. This mass causes disturbance input, since it exerts forces on the second mass that were not modeled. Furthermore, the code `hw6prob04.m` considers the case in which you measure the absolute displacement of this third mass, not of the second mass—the effect is to cause measurement noise. So, think of `hw6prob03.m` as simulating the "real" closed-loop system given the controller and observer you designed for a rough model of this system. Again, repeat (a)-(d) as necessary, choosing ρ and γ so that the "real" closed-loop system has a 5%-settling time of less than 5 seconds (doing so may be a challenge!). Briefly explain *your* design process, submitting whatever plots *you* feel are necessary to justify it. Also submit your `ControlLoop` function and a snapshot of the figure for your final design.

NOTE: In this problem, if you like and if it helps your design process, you are also welcome to play around with the diagonal entries of Q_c and R_o. (Doing so may be essential in this problem, if you want to meet the performance specifications.)

4. In this problem, you will apply the Hamilton-Jacobi-Bellman equation to derive an optimal observer for a scalar state-space system. As you have seen in class, this observer can be found by solving the following optimal control problem:

$$\begin{array}{ll} \underset{d,x(t_1)}{\text{minimize}} & s\left(cx(t_0)-y(t_0)\right)^2+\int_{t_0}^{t_1}\left(q(cx-y)^2+rd^2\right)dt \\ \text{subject to} & \dot{x}=ax+bu+d, \end{array} \tag{1}$$

where $q \geq 0$, $r > 0$, and $s > 0$. The interpretation of this problem is as follows. The current time is t_1. You have taken measurements $y(t)$ over the time interval $[t_0, t_1]$. You are looking for disturbance inputs $d(t)$ over this same time interval and for the state estimate $x(t_1)$ that best explain these measurements. You will find these things in a number of steps. Completing these steps is the hardest bit of theory that I will ask of you this semester. Understanding this theory makes you very dangerous.

The optimal control problem (1) has the general form

$$\begin{array}{ll} \underset{d,x(t_1)}{\text{minimize}} & h(x(t_0))+\int_{t_0}^{t_1} g(x,d)dt \\ \text{subject to} & \dot{x}=f(x,d). \end{array} \tag{2}$$

The Hamilton-Jacobi-Bellman equation for a problem of this form is

$$\frac{\partial v}{\partial t}=\min_{d}\left\{g-\left(\frac{\partial v}{\partial x}\right)f\right\}, \tag{3}$$

where $v(t,x)$ is the *value function*. The reason this expression looks slightly different from what you saw in HW4 for optimal control design is that, in this case, $v(t,x)$ is a "cost-to-go" running *backward in time*, not forward in time. Note that $h(x(t_0))$ looks like

$$p(t_0)x(t_0)^2+2m(t_0)x(t_0)+n(t_0) \tag{4}$$

for an appropriate choice of $p(t_0)$, $m(t_0)$, and $n(t_0)$, so it makes sense to guess a value function of the form

$$v(t,x) = px^2 + 2mx + n, \tag{5}$$

where p, m, and n are functions of time. If this guess is correct, then the optimal choice of state estimate $x(t_1)$ is simply the one that minimizes $v(t_1, x(t_1))$:

$$x(t_1) = -p(t_1)^{-1}m(t_1), \tag{6}$$

as you could verify by completing the square. (Here, we have assumed that $p(t_1) > 0$. This assumption turns out to be true.) What remains is to find p and m as functions of time.

Proceed as follows:

(a) Find f, g, and h by comparing (1) with (2).

(b) Find $\partial v/\partial t$ and $\partial v/\partial x$ by taking partial derivatives of (5).

(c) Plug f, g, h, $\partial v/\partial t$, and $\partial v/\partial x$ into (3).

(d) Minimize the right-hand side of (3) with respect to d, either by "completing the square" or by applying the "first and second derivative test."

(e) Equate coefficients of x^2, x, and 1 in the result. Doing so will produce three first-order ordinary differential equations:

$$\dot{p} = \cdots \qquad \dot{m} = \cdots \qquad \dot{n} = \cdots$$

Write the boundary conditions $p(t_0)$, $m(t_0)$, and $n(t_0)$ for these ODEs by equating coefficients of (4) with $h(t_0)$.

(f) Define $\hat{x} = -p^{-1}m$. Using your answer from part (e), show that

$$\dot{\hat{x}} = a\hat{x} + bu - \ell(c\hat{x} - y) \tag{7}$$

for an appropriate choice of ℓ. Because of (6), $\hat{x}(t_1)$ is the optimal state estimate, and so (7) is an optimal observer.

CONGRATULATIONS! THIS IS A HUGE RESULT.

(g) If t_0 is finite, then the gain ℓ in (7) is time-varying. Suppose $t_0 \to -\infty$. This corresponds to an assumption that we have been running our optimal observer for a long time. It is a remarkable fact that p—hence, ℓ—tends to a steady-state value. Write an equation that characterizes this steady-state value. (There is no need to solve this equation yet.)

(h) The equation you found in part (g) should be quadratic in p. Rewrite this equation in terms of $\bar{p} = p^{-1}$. It should now look very familiar to you. (Compare it to the continuous algebraic Riccati equation that corresponds to LQR.) Express ℓ in terms of $\bar{p}$. You have now found the "steady-state" or "infinite-horizon" optimal observer.

(i) What happens to ℓ as $(q/r) \to 0$ and how will the observer behave?

(j) What happens to ℓ as $(q/r) \to \infty$ and how will the observer behave?

5. **EXTRA CREDIT: OBSERVER FOR PARAMETER IDENTIFICATION**

You owe it to yourself to solve this problem. It is the awesomest thing ever in the world.

An optimal observer for a linear system. Consider the system

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t)\\ y(t) &= Cx(t).\end{aligned}$$

As we have discussed in class, a finite-horizon optimal observer for this system has the form

$$\dot{\widehat{x}}(t) = A\widehat{x}(t) + Bu(t) - L(t)\left(C\widehat{x}(t) - y(t)\right)$$

for some initial guess

$$\widehat{x}(0) = \widehat{x}_0$$

where

$$L(t) = P(t)C^T Q_o$$

and where $P(t)$ is found by solving

$$\dot{P}(t) = R_o^{-1} + AP(t) + P(t)A^T - P(t)C^T Q_o C P(t)$$

forward in time with the initial condition

$$P(0) = P_0.$$

This observer is often referred to as a "continuous-time Kalman filter (KF)."

An "optimal" observer for a nonlinear system. Consider the system

$$\begin{aligned}\dot{x}(t) &= f(x(t), u(t))\\ y(t) &= g(x(t))\end{aligned}$$

where f and g are functions, perhaps nonlinear. It turns out that a finite-horizon "optimal" observer for this system ("optimal" is in quotes because it's not really optimal any more, but it's still good) has the form

$$\dot{\widehat{x}}(t) = f(\widehat{x}(t), u(t)) - L(t)\left(g(\widehat{x}(t)) - y(t)\right)$$

for some initial guess

$$\widehat{x}(0) = \widehat{x}_0$$

where

$$L(t) = P(t)G^T Q_o$$

and where $P(t)$ is found by solving

$$\dot{P}(t) = R_o^{-1} + F(\widehat{x}(t), u(t))P(t) + P(t)F(\widehat{x}(t), u(t))^T - P(t)G(\widehat{x}(t))^T Q_o G(\widehat{x}(t))P(t)$$

forward in time with initial condition

$$P(0) = P_0.$$

In these expressions,

$$F(x,u) = \nabla_x f(x,u) \qquad \text{and} \qquad G(x) = \nabla_x g(x)$$

are gradients of f and g with respect to x. This observer is often referred to as a "continuous-time extended Kalman filter (EKF)."

Both the "KF" and the "EKF" are super-famous, and have been implemented in almost every single aerospace system that has been flown since the 1960's.

Parameter identification as observer design for a nonlinear system. Consider the scalar state-space system

$$\begin{aligned} \dot{z} &= az + bu \\ y &= z \end{aligned} \tag{8}$$

You will recognize this system as describing the hardware that I often bring to class (recall the Millennium Falcon flying across the room). Two weeks ago, we discussed the problem of parameter identification, which was to guess the value of a and b from data. At that time, we took an approach that worked, but that was not very systematic. Now, we will derive a different approach that is more systematic.

The trick is to treat a and b as *variables*. In particular, consider the *nonlinear* system

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t)) \\ y(t) &= g(x(t)) \end{aligned} \tag{9}$$

where

$$f(x,u) = \begin{bmatrix} x_2 x_1 + x_3 u \\ 0 \\ 0 \end{bmatrix} \qquad\qquad g(x) = x_1$$

and, as usual, our notation means

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}.$$

In this system:

- the variable x_1 plays the role of the angular velocity z, which behaves as in (8);
- the variable x_2 plays the role of the parameter a, which is constant;
- the variable x_3 plays the role of the parameter b, which is also constant.

Note that, if

$$x(0) = \begin{bmatrix} z(0) \\ a \\ b \end{bmatrix},$$

then the output of (9) will be exactly the same as the output of (8). Suppose you were to apply the "EKF" that is described above to estimate x given y. You would recover *both* an estimate $\widehat{x}_1$ of the angular velocity z and estimates $\widehat{x}_2$ and $\widehat{x}_3$ of the parameters a and b.

What you need to do.

- The files `datasec01.txt` (for the 11AM section) and `datasec02.txt` (for the 12PM section) contain the experimental data that *you* generated in class with the hardware, in the week between HW4 and HW5. Download the data file for your section.
- The script `hw6prob05.m` opens the data file. Change the name in Line 7 of the code so that it opens the right data file. The result is that you are given an array of times `t`, as well as the inputs `u` and outputs `y` at those times.
- Implement the "EKF" described above to find $\widehat{x}$ at the array of times, given

$$\widehat{x}(0) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \qquad\qquad P(0) = I.$$

  You'll note that this initial guess is completely wrong—in class, we found that $a \approx -1$ and $b \approx 100$. You'll likely need to iterate a little bit to find values of Q and R that work. **HINT:** remember that

$$\dot{\widehat{x}}(t) \approx \frac{\widehat{x}(t+\Delta t) - \widehat{x}(t)}{\Delta t}$$

  and

$$\dot{P}(t) \approx \frac{P(t+\Delta t) - P(t)}{\Delta t}.$$

  These approximations should allow you to integrate to find $\widehat{x}(t)$ and $P(t)$ just like you did in the "implementation" part of Problems 1-4.
- Create four plots:
  - $u(t)$
  - $y(t)$ and $\widehat{x}_1(t)$
  - $\widehat{x}_2(t)$, i.e., your estimate of a
  - $\widehat{x}_3(t)$, i.e., your estimate of b

  Once you see these results, you will no longer regret doing this problem. AWESOME!