Control System Design Project 3

Joseph Shenouda

December 2019

1 Introduction

In this project we worked with Linear Quadratic Controller to find the optimal full-state feedback gain matrix for our system based off the integral quadratic performance criterion optimization problem and a constraint using two penalty matrices that were given to us. We were able to regulate the system down to zero because the system was obtained by linearization about a nominal point. In part 1b of the project we also created a non-zero set-point controller by looking to regulate one state variable to a desired constant r value. In part 2a of the project we investigate the case when we do not have all the states availiable for feedback and calculate the optimal performance loss cause by using an observer to estimate our state variables. Finally in part 2b we worked with Kalman filters to observe the effect they have in de-noising signals.

2 Problem Setup

We used the following state space equation for part 1a,1b and 2a of the project.

$$A = \begin{pmatrix} -0.01357 & -32.2 & -46.3 & 0 \\ 0.00012 & 0 & 1.214 & 0 \\ -0.0001212 & 0 & -1.214 & 1 \\ 0.00057 & 0 & -9.1 & -0.6696 \end{pmatrix}$$

$$B = \begin{pmatrix} -0.433 \\ 0.1394 \\ -0.1394 \\ -0.1577 \end{pmatrix}$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

And our D matrix is just zeros. We can easily see that our system only outputs all of the state variables which allows us to perform full state feedback on the system to get a desired output.

3 Part 1a Zero Set-point Linear Quadratic Controller

Our goal is to design a full state feedback controller that minimizes the following expression with respect to the system input.

$$J = \frac{1}{2}\int_0^\infty (x^T(t)R_1x(t) + u^T(t)R_2u(t))dt, R_1 = R_1^T > 0, R_2 = R_2^T > 0$$

We wish to obtain an optimal feedback gain $-F^{opt}$ such that $u^{opt}(t) = -F^{opt}x(t)$ using $R_1 = C^TCandR_2 = I_m$ and we can find the actual value of our optimization function by $J_{opt} = 0.5 * x_0^TPx_0$ we used the Matlab function "lqr" to obtain F^{opt} and P. The results of the system dynamics with optimal feedback is presented.

Our Simulink model of the optimal full state feedback system is

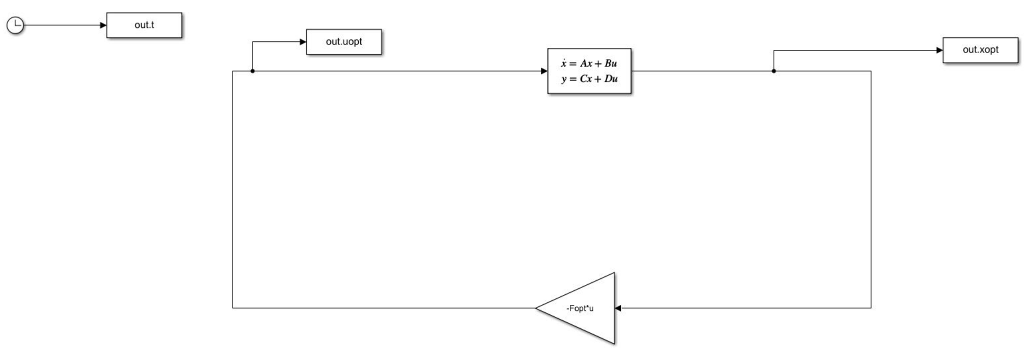
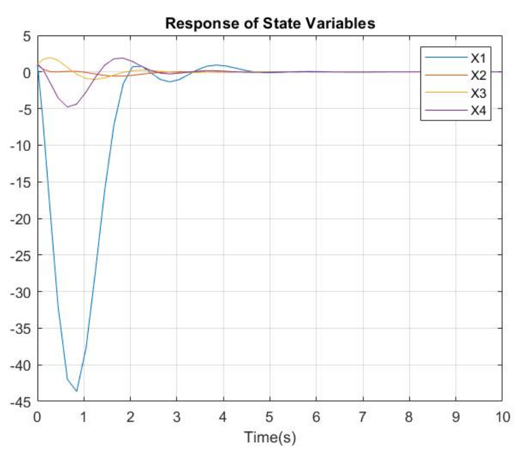
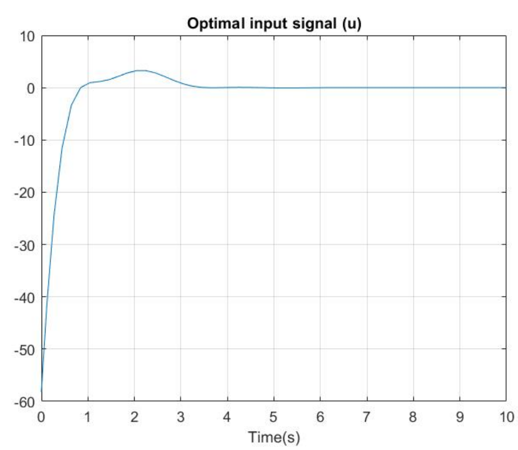


Figure 1: Simulink Model for full state optimal feedback

After running this simulation we plotted the state variables that the system input $u(t)$



(a) Plot of state variables



(b) Plot of u

As we can see from the figures provided our settling time was pretty quick and we did not need to really adjust for high oscillations. We also checked for controllability and observability in our system before simulating. Finally we acheived a $Jopt = 0.00102$.

4 Part 1b Non-Zero Set Point LQ Controller

This part of the project was very similar to part 1a only this time we were looking to acheive a constant steady state value for our system. I desired to set the first state variable X1 to be 10 for my system. We choose a C_z such that $z(t) = C_z x(t)$ got us to a constant value r in the steady state. We know that in the steady state we have

$$0 = Ax_{ss} + Bu_{ss}$$

so we set up the following problem and solve:

$$\begin{pmatrix} A & B \\ C_z & 0 \end{pmatrix} \begin{pmatrix} x_{ss} \\ u_{ss} \end{pmatrix} = \begin{pmatrix} 0 \\ r \end{pmatrix}$$

After solving our system we find that we are looking for the following $u_{opt}(t) = -F_{opt}(t)x_{opt}(t) + Fx_{ss} + u_{ss}$ from our results we got $J_{opt} = 0.001015$

The Simulink model of the non-zero set point LQ controller:

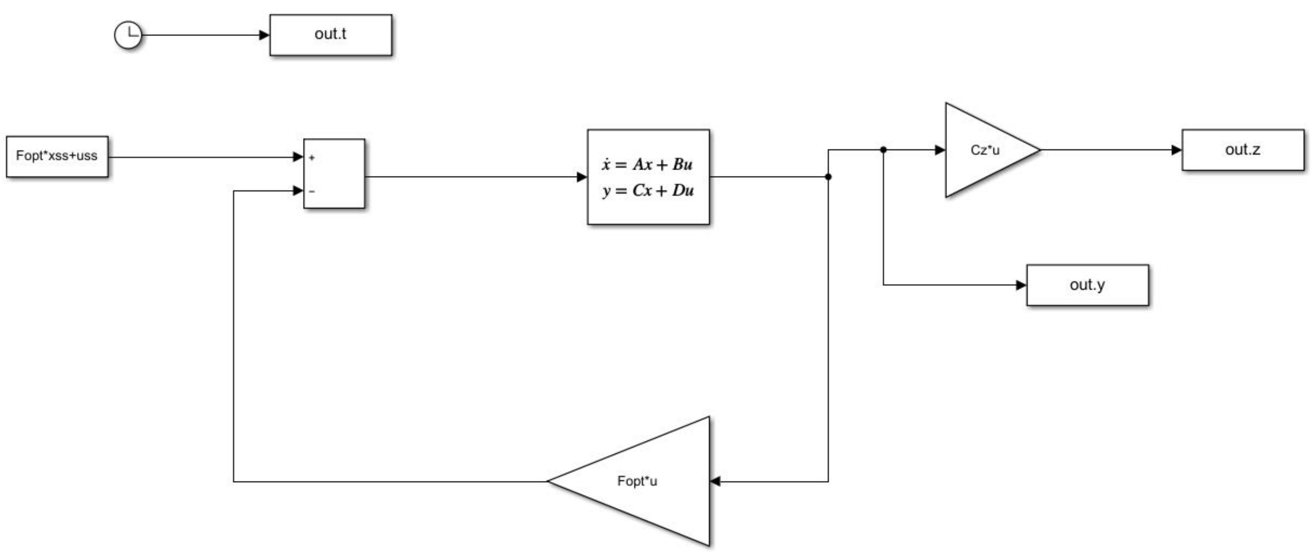


Figure 3: Simulink Model

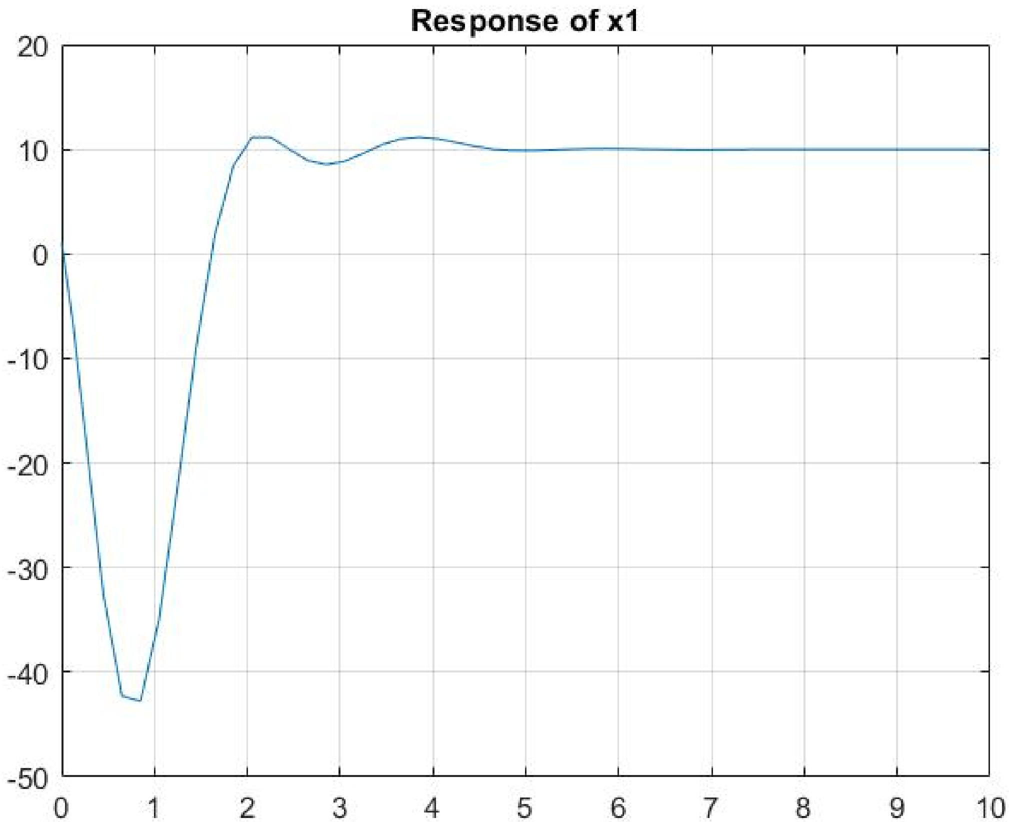


Figure 4: Plot of x1 part 1b

5 Part 2a Observer effects

For this part of the project we looked at the loss we obtain when we don't have all state variables for feedback and need to use an observer. We found our J_{loss} by finding the observer gain K that places our closed loop eigenvalues 5 times further left than the optimal feedback gain. Then we used Lyapunov function from Matlab to get our new $P22$ matrix which we used to find $J_{loss} = 0.5 * e_0^T * P22 * e_0$ after running our code we got $J_{loss} = 0.0142$ which is very small so the observer did not have that great of an effect.

6 Part 2b Kalman Filter

For this section we looked at the stochastic linear systems and the effects of the Kalman filter in reducing white Gaussian noise in our system. We observed the simulation implementing the Kalman filter and present the plots obtained with and without the Kalman filter below. As we can see the Kalman filter does a great job at eliminating teh gaussian white noise being injected into our system.

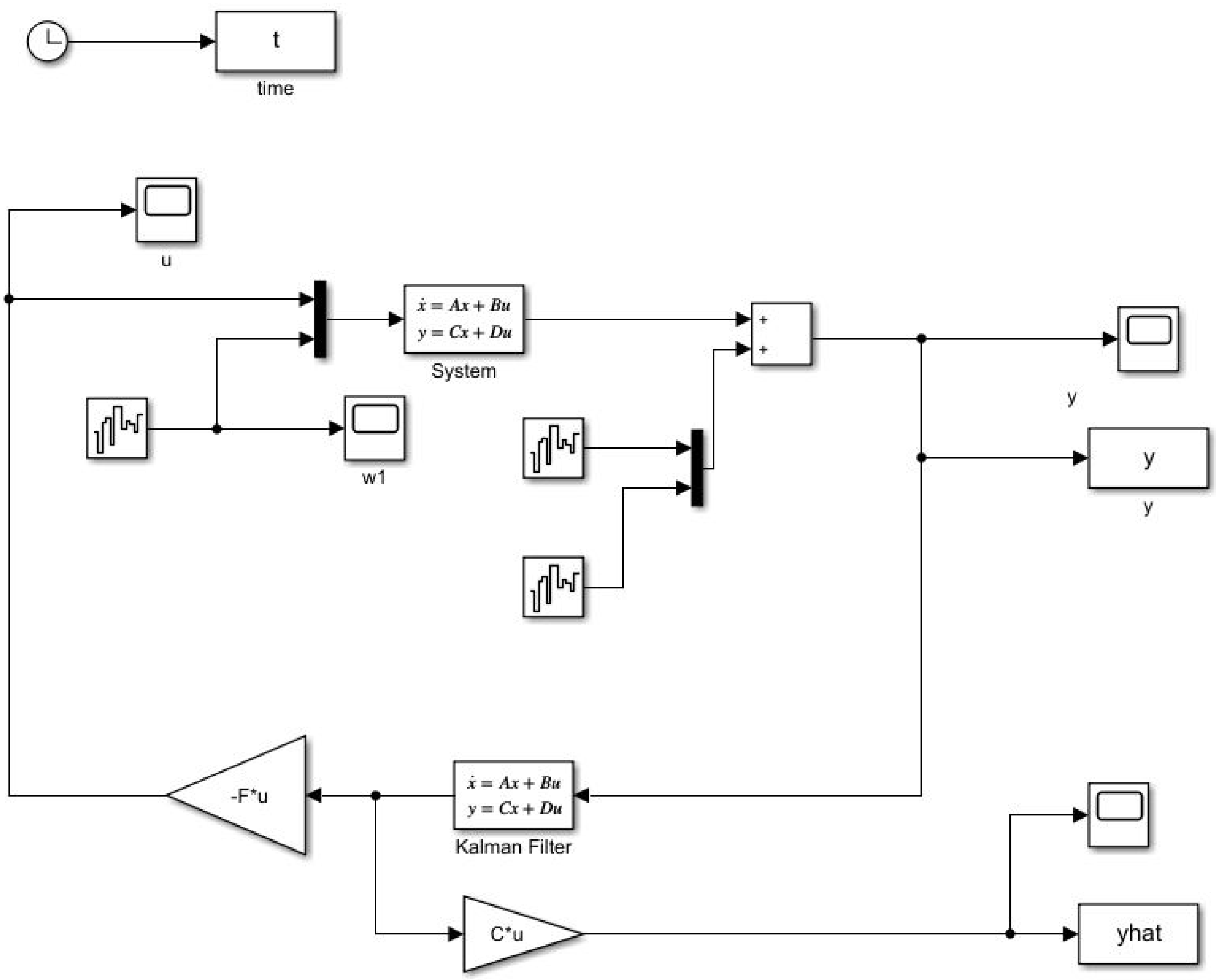
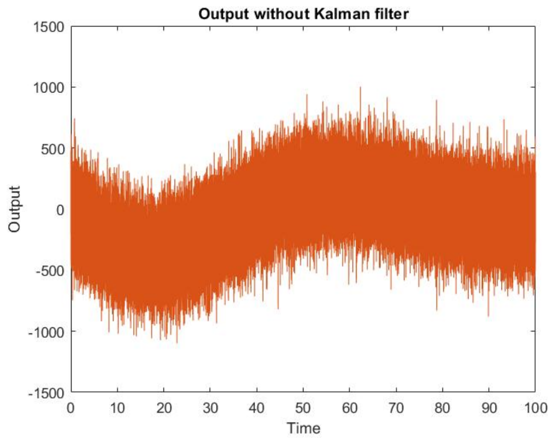
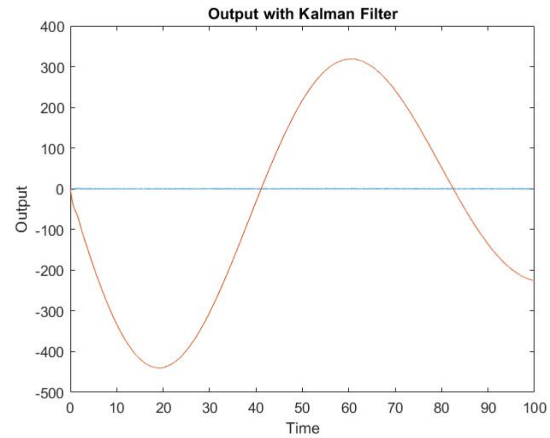


Figure 5: Simulink Model for Kalman Filter



(a) Plot of output without Kalman filter



(b) Plot of output with Kalman filter

7 Discussion

In this project we observed the power of the Linear Quadratic Controller in both the zero set point case as well as the non-zero set point. In both we saw how we could use Matlab to solve the Riccati equation and produce for us the desired optimal feedback controller. We also saw how much loss we get from the optimization problem when we try to use an observer to estimate all the states. Finally we got a glimpse at how the Kalman filter works and its potential to be used in real world systems were we must take into account stochastic processes that may effect our control system.

8 MATLAB Code

8.1 Part 1a Zero set point LQ Controller

```
% Dimensions of state variables
n = 4;

% Dimensions of output
m = 1;

% Dimensions of controler
l = 1;

A = [-0.01357 -32.2 -46.3 0;
     0.00012 0 1.214 0;
     -0.0001212 0 -1.214 1;
     0.00057 0 -9.1 -0.6696];
B = [-0.433;0.1394;-0.1394;-0.1577];
C = eye(n);
```

```
D = zeros(n,1);

R1 = C'*C;
R2 = 1;

CM = ctrb(A,B); rank(CM); %controllability test
OB = obsv(A,C); rank(OB); %observability test

x0 = [1;1;1;1];

[Fopt, P] = lqr(A,B,R1,R2);

Jopt = 0.5*x0'*P*x0;

out = sim('part1aSimulink');

run('plots1a');
```

8.2 Part 1b Non-zero setpoint LQ controller

```
% n = number of state variables; m = number of inputs
% l  = number of outputs;
n = 4; m = 1;
A = [-0.01357 -32.2 -46.3 0;
     0.00012 0 1.214 0;
     -0.0001212 0 -1.214 1;
     0.00057 0 -9.1 -0.6696];
B = [-0.433;0.1394;-0.1394;-0.1577];
C = eye(n);
D = zeros(n,1);

%Our goal is to regulate x1 of our state variables to 10
r = 10; Cz = [1 0 0 0];

M = [A B; Cz 0];

sol = inv(M)*[zeros(4,1);r];

xss = sol(1:4);
uss = sol(5);

R1 = Cz'*Cz;
R2 = 1;

CM = ctrb(A,B); rank(CM); %controllability test
OB = obsv(A,C); rank(OB); %observability test

```

```
x0 = [1;1;1;1];

[Fopt, P] = lqr(A,B,R1,R2);
ev_cl = eig(A-B*Fopt);
Jopt = 0.5*x0'*P*x0;

out1b = sim('part1bSimulink');
run('plots1b');
```

8.3 Part 2a Effects of Observer

```
% n = number of state variables; m = number of inputs
% l  = number of outputs;
n=4;
A = [-0.01357 -32.2 -46.3 0;
     0.00012 0 1.214 0;
     -0.0001212 0 -1.214 1;
     0.00057 0 -9.1 -0.6696];
B = [-0.433;0.1394;-0.1394;-0.1577];
C = eye(n);
D = zeros(n,1);

R1 = C'*C;
R2 = 1;

x0 = [1;1;1;1];
x0hat = [1.5;1.5;1.5;1.5];

e0 = x0-x0hat;

[Fopt, P] = lqr(A,B,R1,R2);

eig_observer = 5*eig(A-B*Fopt);

K = place(A,B,eig_observer);

P22=lyap((A-K*C)',P*B*inv(R2)*B'*P);

Jloss = 0.5*e0'*P22*e0;
```

8.4 Part 2b Kalman Filter

```
%
% F-8 Aircraft LQG Optimal Control
%
clear all
n=4
```

```
m1=1
m2=1
r=2
%
A=[-0.01357   -32.2   -46.3   0;
     0.00012     0      1.214   0;
    -0.0001212   0     -1.214   1;
    0.00057      0     -9.01   -0.6696]
B=[-0.433; 0.1394; -0.1394; -0.1577]
G=[-46.3; 1.214; -1.214; -9.01]
W1=[0.000315]
W2=[0.000686 0;
    0 40]
C=[0 0 0 1;
    1 0 0 0];
D=zeros(2,1)
x0=[1 1 1 1]'
R1=diag([0.001,0, 3260, 3260])
R2=3260
% Optimal control
[F,P]=lqr(A,B,R1,R2);
[K,PF]=lqe(A,G,C,W1,W2)
Jopt=trace(P*G*W1*G'+PF*F'*R2*F)
```