ECE 463/663 - Homework #10

LQG Control. Due Monday, April 8th
Please submit as a hard copy, email to jacob.glower@ndsu.edu, or submit on BlackBoard

LQG Control

1) Cart & Pendulum (HW #4 & HW#6):

$$s\begin{bmatrix} x \\ \theta \\ \dot{x} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -2.45 & 0 & 0 \\ 0 & 9.42 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \theta \\ \dot{x} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0.25 \\ -0.1923 \end{bmatrix} F$$

Design a full-state feedback control law of the form

$$F = U = K_r R - K_x X$$

for the cart and pendulum system from homework #4 using LQG control so that

- The DC gain is 1.00
- The 2% settling time is 8 seconds, and
- There is less than 5% overshoot for a step input.

The desired response is

$$x = \left(\frac{0.5249}{(s+0.5+j0.5243)(s+0.5-j0.5242)} \right) = \left(\frac{0.5249}{s^2+s+0.5249} \right)$$

Compare your results with homework #6

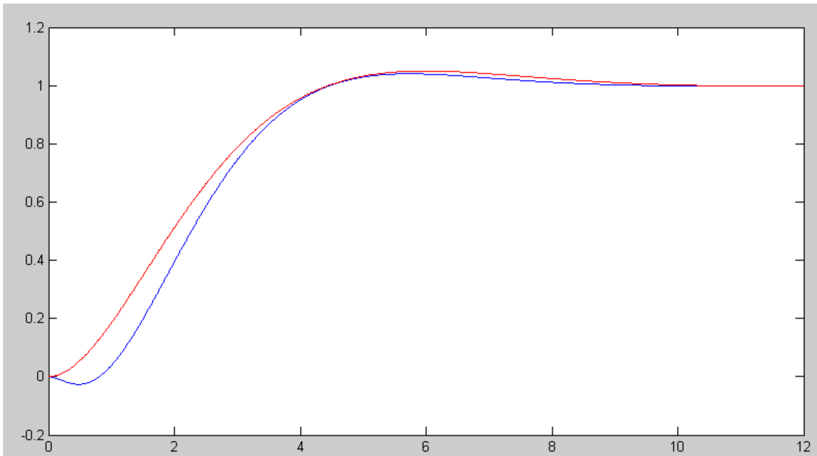
Start with a script that lets you adjust the gains on x^2 and $\dot{x}^2$

Play with the gains until the step response is clos to the desired response

```
A = [0,0,1,0;0,0,0,1;0,-2.45,0,0;0,9.42,0,0];
B = [0;0;0.25;-0.1923];
C = [1,0,0,0];

Gd = tf(0.5249, [1,1,0.5249]);
t = [0:0.01:12]';
Yd = step(Gd, t);

Qy = C'*C;
Qv = (C*A)' * (C*A);

Kx = lqr(A, B, 15*Qy + 0*Qv, 1);
DC = -C*inv(A-B*Kx)*B;
Kr = 1/DC;
Gcl = ss(A-B*Kx, B*Kr, C, 0);
Y = step(Gcl, t);

plot(t,Y,'b',t,Yd,'r');
```

After some trial and error, the result that looks pretty good is

$$Q = 15 \cdot (C^T C) + 0\left((CA)^T (CA)\right)$$



Homework #10 (LQR)	Homework #6 (pole placement)

`>> eig(A - B*Kx)`
 	
`-3.0668 + 0.0314i`
`-3.0668 - 0.0314i`
`-0.6291 + 0.6164i`
`-0.6291 - 0.6164i`
	`>> eig(A - B*Kx)`
	
	`-6.0000`
	`-5.0000`
	`-0.8000 + 0.8390i`
	`-0.8000 - 0.8390i`

`Kx' =`
`-3.8730`
`-147.1007`
`-8.8078`
`-49.8895`
	`Kx' =`
	`-21.4015`
	`-331.3277`
	`-33.3268`
	`-108.8492`

Gains are about half when using LQR

2) Cart & Pendulum with Multiple Inputs

Assume a torque on the beam is also allowed

$$s\begin{bmatrix} x \\ \theta \\ \dot{x} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -2.45 & 0 & 0 \\ 0 & 9.42 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \theta \\ \dot{x} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0.25 \\ -0.1923 \end{bmatrix} F + \begin{bmatrix} 0 \\ 0 \\ -0.1923 \\ 0.7396 \end{bmatrix} T$$

Design a full-state feedback control law of the form

$$\begin{bmatrix} F \\ T \end{bmatrix} = K_r R - K_x X$$

where Kx is a 2x4 matrix using LQG techniques so that

- The DC gain from R to x is 1.00
- The 2% settling time is 8 seconds, and
- There is less than 5% overshoot for a step input.

This took some effort to get the dimensions to work out... In matrix form

$$sX = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -2.45 & 0 & 0 \\ 0 & 9.42 & 0 & 0 \end{bmatrix} X + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.25 & -0.1923 \\ -0.1923 & 0.7396 \end{bmatrix} U$$

Assume R drives both inputs the same

$$K_r = \begin{bmatrix} k \\ k \end{bmatrix}$$

so that the dimensions work out:

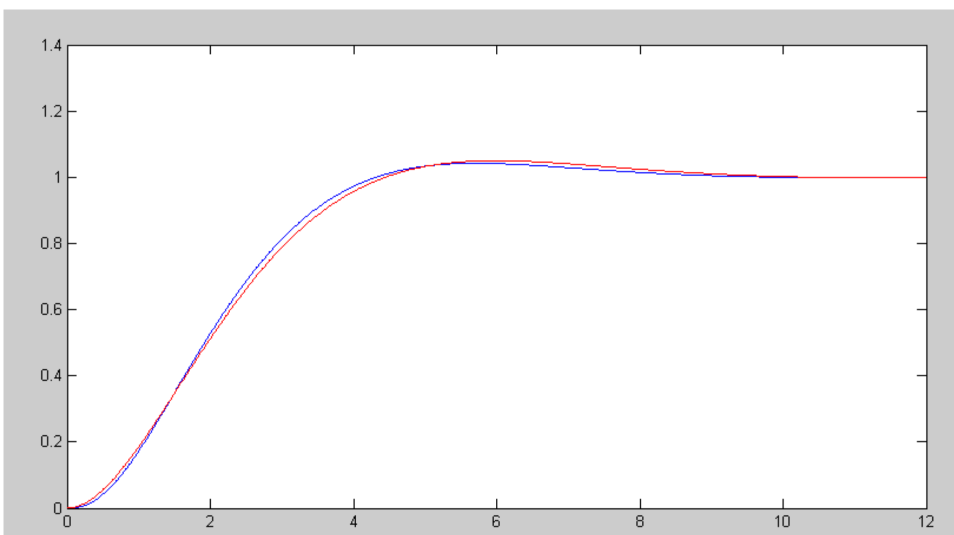
$$sX_{4x1} = A_{4x4} X_{4x1} + B_{4x2} K_{r2x1} R_{1x1}$$

Start with a script

```
A = [0,0,1,0;0,0,0,1;0,-2.45,0,0;0,9.42,0,0];
Bf = [0;0;0.25;-0.1923];
Bt = [0;0;-0.1923;0.7396];
B = [Bf,Bt];
C = [1,0,0,0];
D = [0,0];

Gd = tf(0.5249, [1,1,0.5249]);
t = [0:0.01:12]';
Yd = step(Gd, t);

R = [1,0;0,1];

Qy = C'*C;
Qv = (C*A)' * (C*A);

Kx = lqr(A, B, 10*Qy + 0*Qv, R);
Kr = [1;1];
DC = -C*inv(A-B*Kx)*B*Kr;
Kr = Kr / DC;
Gcl = ss(A-B*Kx, B*Kr, C, 0);
Y = step(Gcl, t);

plot(t,Y,'b',t,Yd,'r');
```

Some trial and error resulted in

$$Q = 10(C^T C) + 0\left((CA)^T (CA)\right)$$



The resulting control law is

```
>> Kx

    3.0198   -3.7837    5.2438   -0.1115
    0.9386   24.8974    2.2674    8.6202

>> Kr

    2.5263
    2.5263
```

The resulting closed-loop poles:

```
>> eig(A - B*Kx)

  -0.5668 + 0.5572i
  -0.5668 - 0.5572i
  -3.0692 + 0.1023i
  -3.0692 - 0.1023i
```

With LQR, you can handle multiple inputs...

3) Ball and Beam (HW #4 & HW#6):

$$s\begin{bmatrix} r \\ \theta \\ \dot{r} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -7 & 0 & 0 \\ -5.88 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} r \\ \theta \\ \dot{r} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0.2 \end{bmatrix} T$$

Design a full-state feedback control law of the form

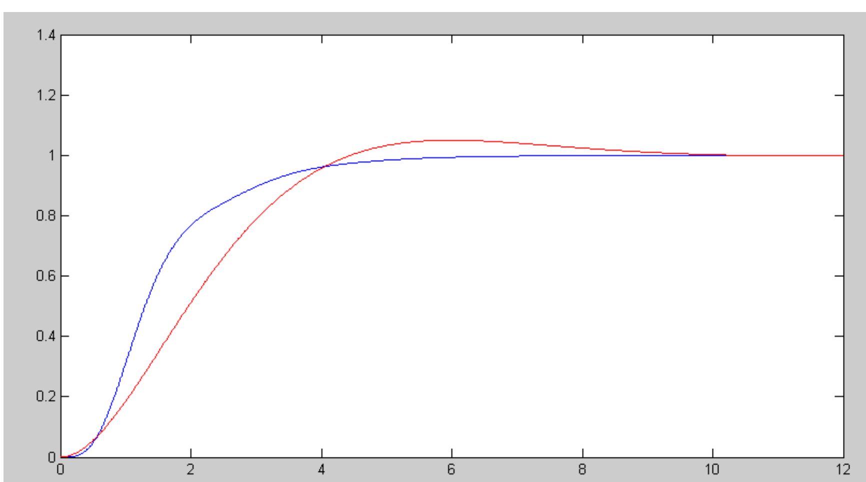
$$T = U = K_r R - K_x X$$

for the ball and beam system from homework #4 using LQG control so that

- The DC gain is 1.00
- The 2% settling time is 8 seconds, and
- There is less than 5% overshoot for a step input.

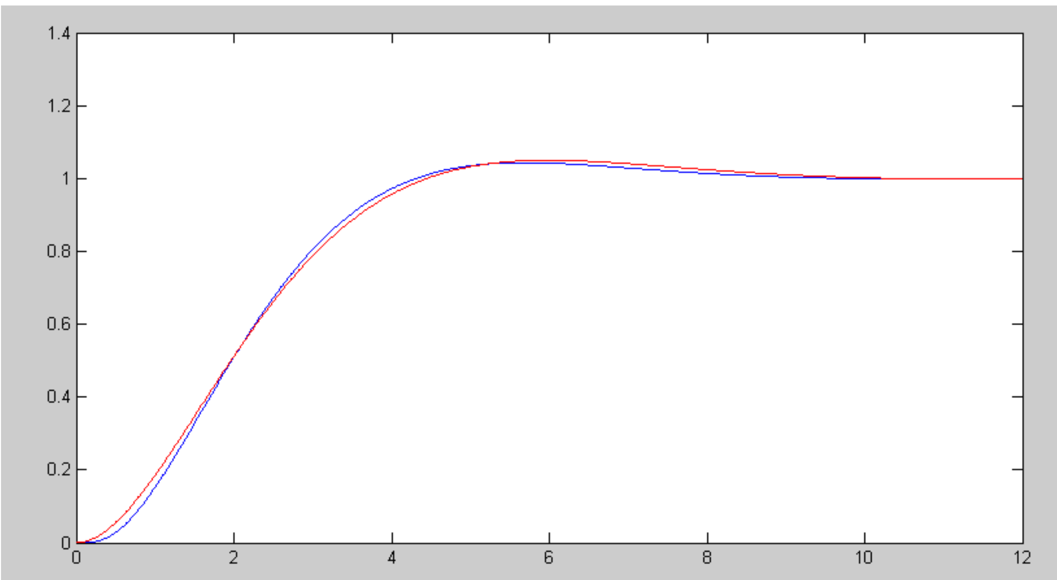
Weighting x and x' just isn't working very well

```
A = [0,0,1,0;0,0,0,1;0,-7,0,0;-5.88,0,0,0];
B = [0;0;0;0.2];
C = [1,0,0,0];

Gd = tf(0.5249, [1,1,0.5249]);
t = [0:0.01:12]';
Yd = step(Gd, t);

Qy = C'*C;
Qv = (C*A)' * (C*A);

Kx = lqr(A, B, 0.1*Qy + 1000*Qv, 1);
DC = -C*inv(A-B*Kx)*B;
Kr = 1/DC;
Gcl = ss(A-B*Kx, B*Kr, C, 0);
Y = step(Gcl, t);

plot(t,Y,'b',t,Yd,'r');
```



So, free up Q to weight each term:

```
A = [0,0,1,0;0,0,0,1;0,-7,0,0;-5.88,0,0,0];
B = [0;0;0;0.2];
C = [1,0,0,0];

Gd = tf(0.5249, [1,1,0.5249]);
t = [0:0.01:12]';
Yd = step(Gd, t);

Q = diag([1,100000,1,1000]);

Kx = lqr(A, B, Q, 1);
DC = -C*inv(A-B*Kx)*B;
Kr = 1/DC;
Gcl = ss(A-B*Kx, B*Kr, C, 0);
Y = step(Gcl, t);

plot(t,Y,'b',t,Yd,'r');
```



The resulting feedback gains:

```
>> Kx

Kx =  -58.8170  389.8451  -57.2503   69.9889

>> Kr

Kr =  -29.4170
```

The resulting closed-loop poles

```
>> eig(A - B*Kx)

  -6.4262 + 4.6145i
  -6.4262 - 4.6145i
  -0.5727 + 0.5744i
  -0.5727 - 0.5744i
```

<table>
<tr><th>Homework #10 (LQR)</th><th>Homework #6 (pole placement)</th></tr>
<tr><td><pre>>> eig(A - B*Kx)

  -6.4262 + 4.6145i
  -6.4262 - 4.6145i
  -0.5727 + 0.5744i
  -0.5727 - 0.5744i</pre></td><td><pre>>> eig(A - B*Kx)

  -6.0000
  -5.0000
  -0.8000 + 0.8390i
  -0.8000 - 0.8390i</pre></td></tr>
<tr><td><pre>Kx' =
 -58.8170
 389.8451
 -57.2503
  69.9889</pre></td><td><pre>Kx' =
 -58.1983
 244.7196
 -44.8451
  63.0000</pre></td></tr>
</table>

The feedback gains are abou tthe same - maybe a little smaller with pole placement

The optimal poles location is similar to pole placement - but easier to come up with