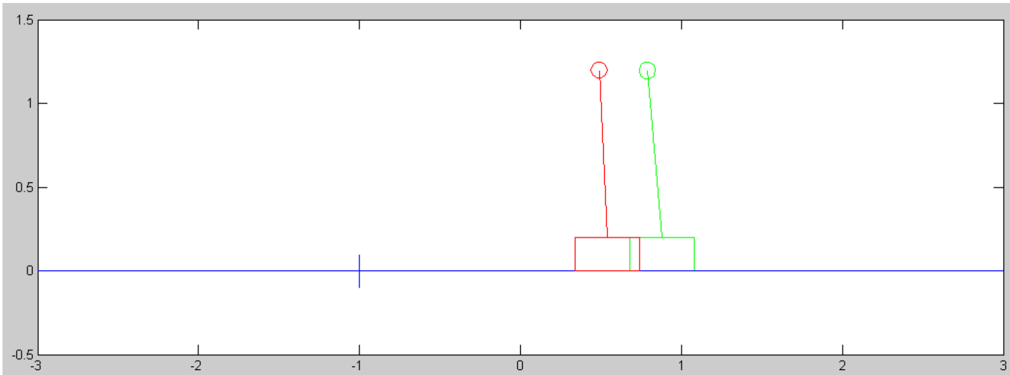
ECE 463: Homework #8

Linear Observers. Due Monday March 21st



Cart and Pendulum from homework #4 with a state estimator (green)

Use the dynamics for the cart and pendulum from homework set #4

$$s\begin{bmatrix} x \\ \theta \\ sx \\ s\theta \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -14.7 & 0 & 0 \\ 0 & 24.5 & 0 & 0 \end{bmatrix}\begin{bmatrix} x \\ \theta \\ sx \\ s\theta \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0.5 \\ -0.5 \end{bmatrix}F$$

1) Design a full-state feedback control law of the form

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

so that the closed-loop system has

- A 2% settling time of 10 seconds, and
- 10% overshoot for a step input.

Plot the step response of the linarized system in Matlab.

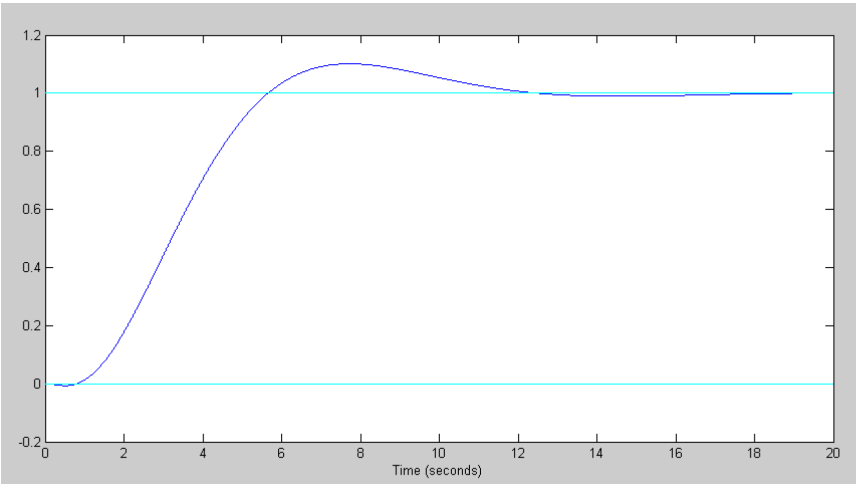
```
>> A = [0,0,1,0;0,0,0,1;0,-14.7,0,0;0,24.5,0,0]

         0         0    1.0000         0
         0         0         0    1.0000
         0  -14.7000         0         0
         0   24.5000         0         0

>> B = [0;0;0.5;-0.5]

         0
         0
    0.5000
   -0.5000

>> C = [1,0,0,0];
>> D = 0;

>> Kx = ppl(A, B, [-0.33+j*0.46,-0.33-j*0.46,-2,-3])
```

Kx =   -0.3924   -68.6334   -1.1352   -12.4552

```
>> DC = -C*inv(A-B*Kx)*B;
>> Kr = 1/DC
```

Kr =   -0.3924

```
>> t = [0:0.01:20]';
>> R = 0*t + 1;
>> X0 = zeros(4,1);
>> y = step3(A-B*Kx, B*Kr, C, D, t, X0, R);
>> plot(t,y)
>> xlabel('Time (seconds)');
```



Assume you can only measure the cart position and beam angle.

2) Design a full-order observer to estimate all four states so that the observer is 2-5 times faster than the plant. You may use either cart position or beam angle (or both) as measurements.

```
>> H = ppl(A', C', [-1,-1.2,-1.4,-1.6])

H =

    5.2000   -9.2468   34.5400  -57.7495

>> H = ppl(A', C', [-1,-1.2,-1.4,-1.6])'

H =

    5.2000
   -9.2468
   34.5400
  -57.7495

>> eig(A - H*C)

ans =

   -1.6000
   -1.4000
   -1.2000
   -1.0000

>>
```

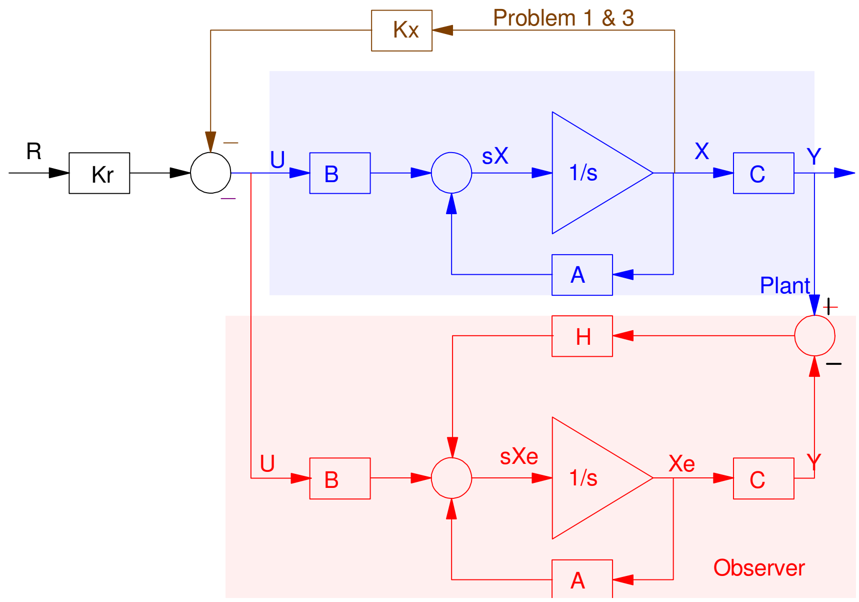
3) Give the state-space model of the closde loop system using the actual states:

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

and plot the step response with initial conditions of

$X(0) = [0, 0, 0, 0]'$ $\qquad X_{observer}(0) = [0.1, 0.1, 0.1, 0.1]'$

(note: use the function step3)

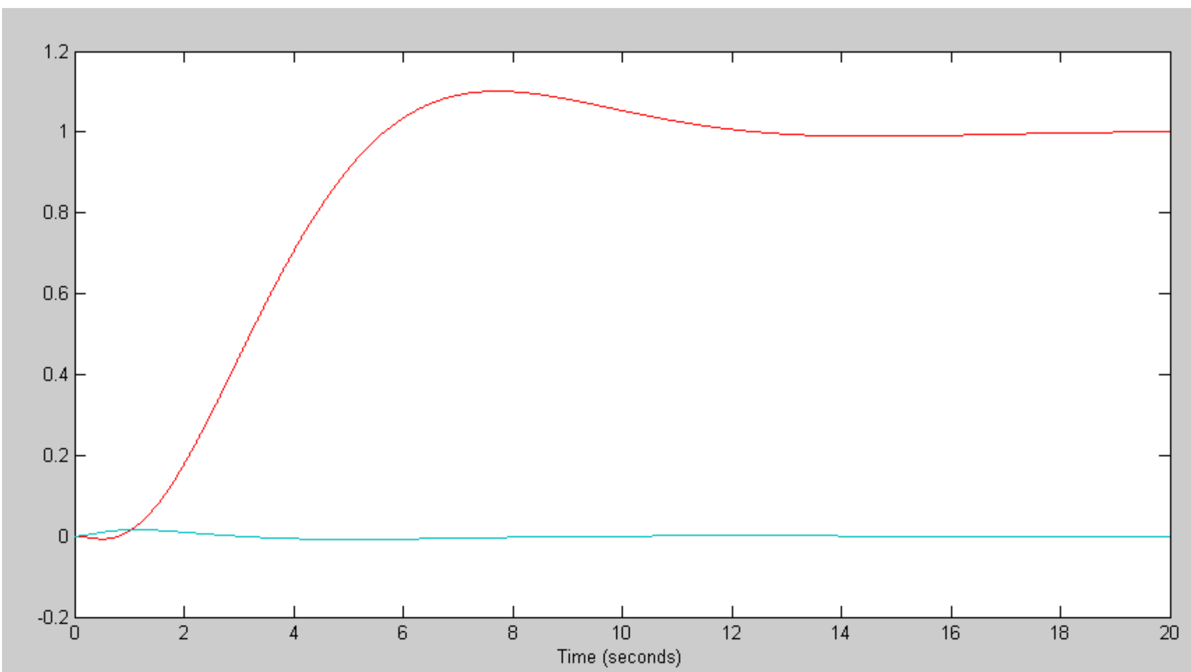


$$\begin{bmatrix} sX \\ sX_e \end{bmatrix} = \begin{bmatrix} A - BK_x & 0 \\ HC - BK_x & A - HC \end{bmatrix} \begin{bmatrix} X \\ X_e \end{bmatrix} + \begin{bmatrix} BK_r \\ BK_r \end{bmatrix} R$$

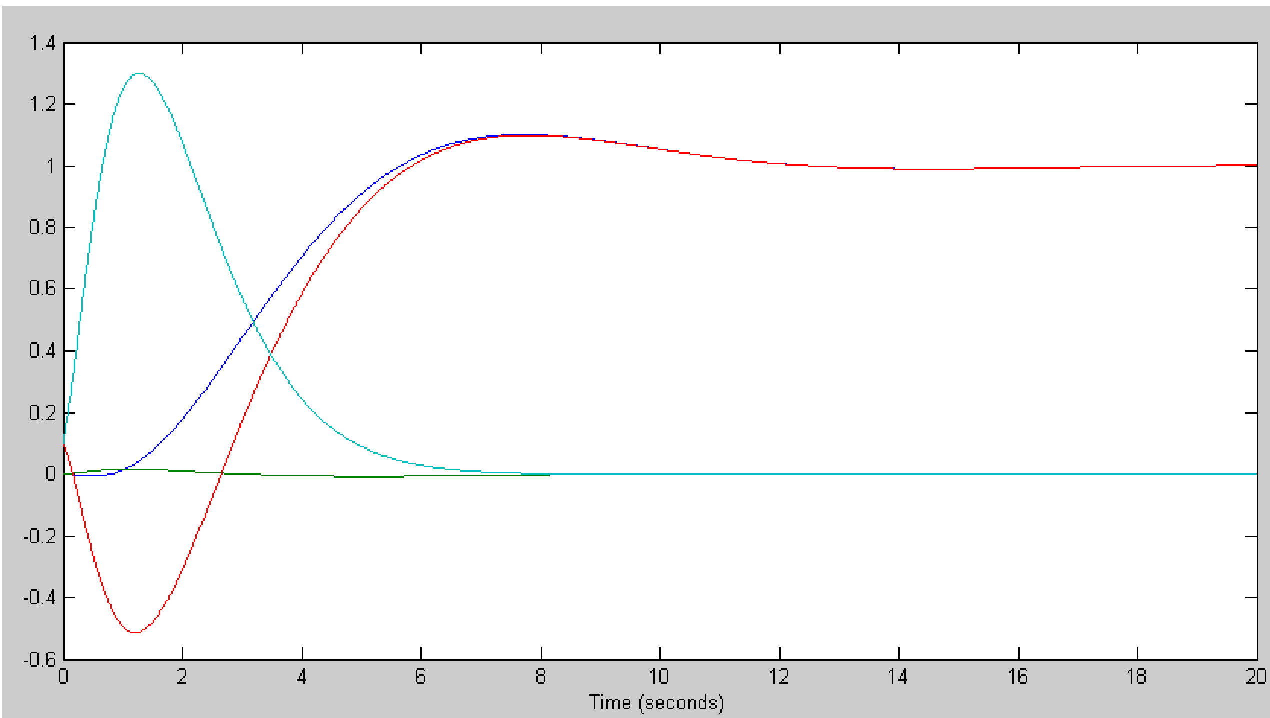
Plotting the step response

```
>> A8 = [A-B*Kx, zeros(4,4) ; H*C - B*Kx, A-H*C]

         0         0    1.0000         0         0         0         0         0
         0         0         0    1.0000         0         0         0         0
    0.1962   19.6167    0.5676    6.2276         0         0         0         0
   -0.1962   -9.8167   -0.5676   -6.2276         0         0         0         0
    5.2000         0         0         0   -5.2000         0    1.0000         0
   -9.2468         0         0         0    9.2468         0         0    1.0000
   34.7362   34.3167    0.5676    6.2276  -34.5400  -14.7000         0         0
  -57.9457  -34.3167   -0.5676   -6.2276   57.7495   24.5000         0         0
```

```
>> B8 = [B*Kr;B*Kr]

         0
         0
   -0.1962
    0.1962
         0
         0
   -0.1962
    0.1962

>> C8 = [1,0,0,0,0,0,0,0;0,1,0,0,0,0,0,0;0,0,0,0,1,0,0,0;0,0,0,0,0,1,0,0]

x       1      0      0      0      0      0      0      0
q       0      1      0      0      0      0      0      0
xe      0      0      0      0      1      0      0      0
qe      0      0      0      0      0      1      0      0

>> D8 = [0;0;0;0];
>> X0 = zeros(8,1);
>> R  = 0*t + 1;
>> y = step3(A8, B8, C8, D8, t, X0, R);
>> plot(t,y)
>> xlabel('Time (seconds)');
```



```
>> X0 = [0;0;0;0;0.1;0.1;0.1;0.1]

         0
         0
         0
         0
    0.1000
    0.1000
    0.1000
    0.1000

>> y = step3(A8, B8, C8, D8, t, X0, R);
>> plot(t,y)
>> xlabel('Time (seconds)');
```

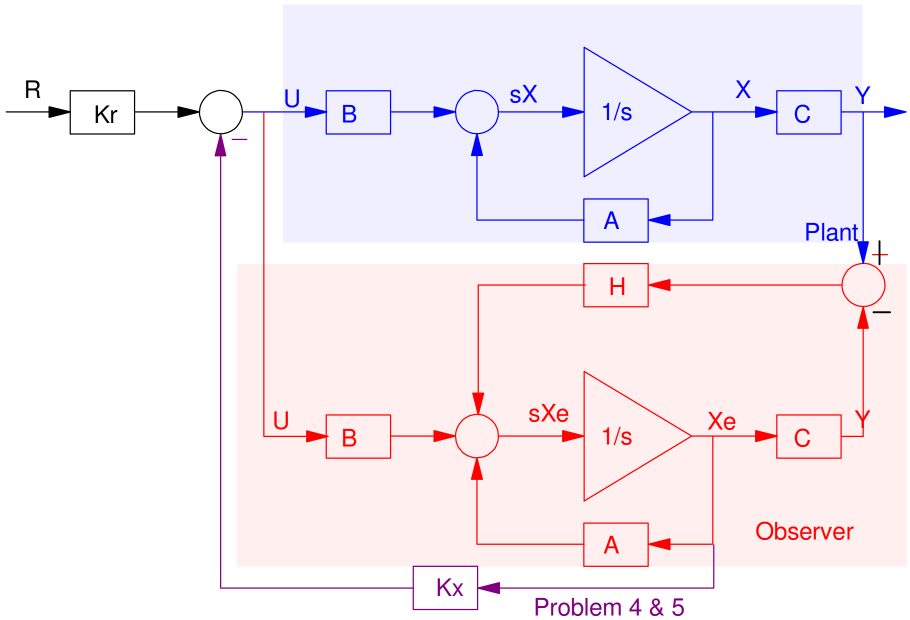


4) Give the state-space model of the closed loop system using the state estimates:

$$U = K_r R - K_x X_{observer}$$

and plot the step response with initial conditions of

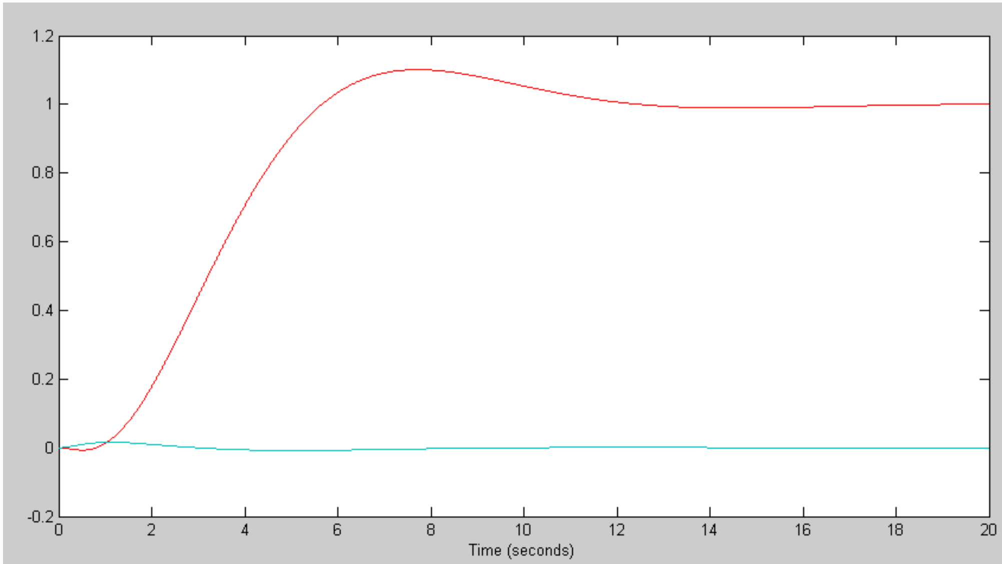
$X(0) = [0, 0, 0, 0]'$ $\quad X_{observer}(0) = [0.1, 0.1, 0.1, 0.1]'$



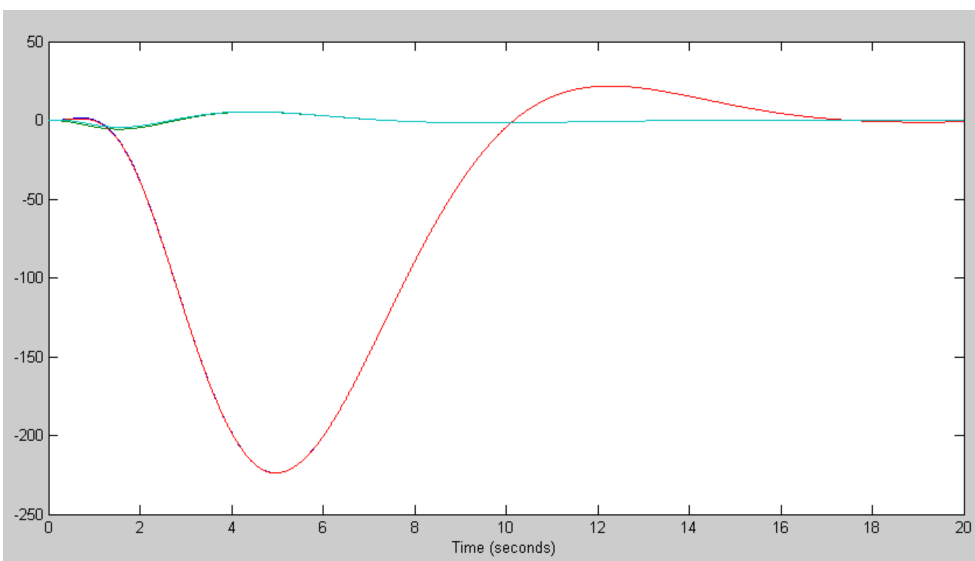
$$\begin{bmatrix} sX \\ sX_e \end{bmatrix} = \begin{bmatrix} A & -BK_x \\ HC & A-HC-BK_x \end{bmatrix} \begin{bmatrix} X \\ X_e \end{bmatrix} + \begin{bmatrix} BK_r \\ BK_r \end{bmatrix} R$$

In Matlab

```
>> A8 = [A, -B*Kx ; H*C , A-H*C-B*Kx]

         0         0    1.0000         0         0         0         0         0
         0         0         0    1.0000         0         0         0         0
         0  -14.7000         0         0    0.1962   34.3167    0.5676    6.2276
         0   24.5000         0         0   -0.1962  -34.3167   -0.5676   -6.2276
    5.2000         0         0         0   -5.2000         0    1.0000         0
   -9.2468         0         0         0    9.2468         0         0    1.0000
   34.5400         0         0         0  -34.3438   19.6167    0.5676    6.2276
  -57.7495         0         0         0   57.5533   -9.8167   -0.5676   -6.2276

>> X0 = zeros(8,1);
>> y = step3(A8, B8, C8, D8, t, X0, R);
>> plot(t,y)
>> xlabel('Time (seconds)');
```




```
>> X0 = [0;0;0;0;0.1;0.1;0.1;0.1]

         0
         0
         0
         0
    0.1000
    0.1000
    0.1000
    0.1000

>> y = step3(A8, B8, C8, D8, t, X0, R);
>> plot(t,y)
>> xlabel('Time (seconds)');
```

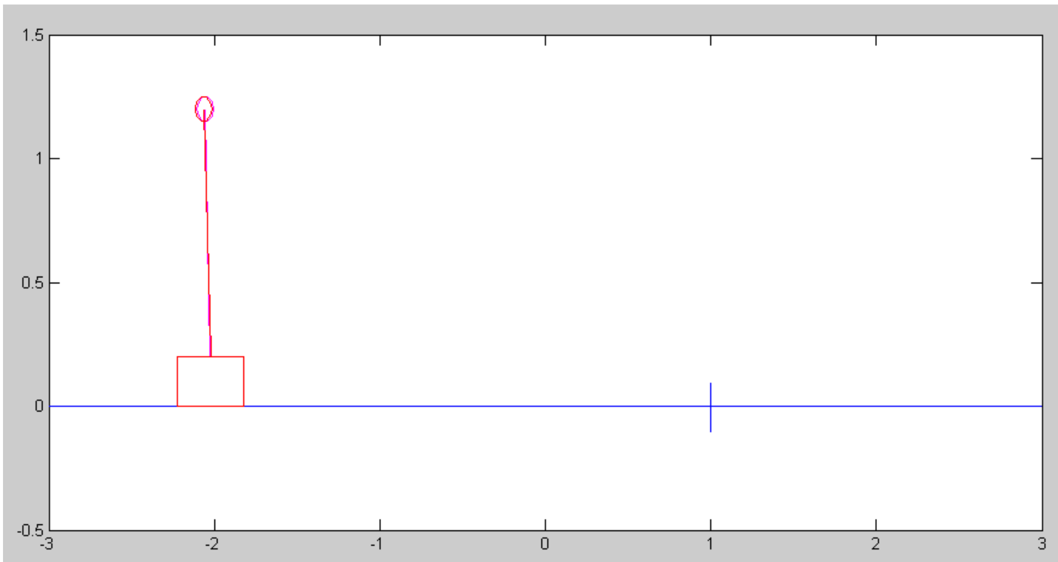


5) Modify the cart and pendulum system to include

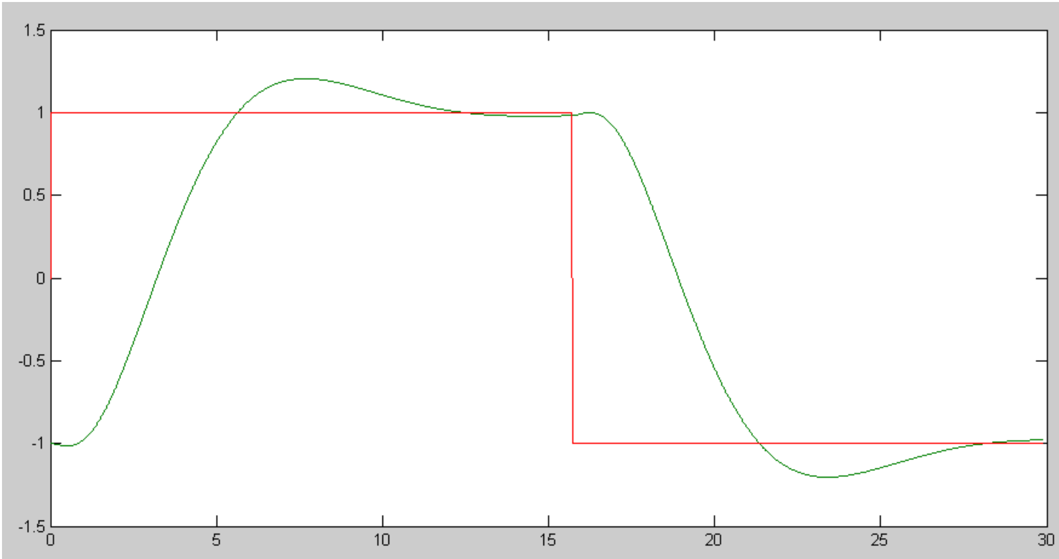
- your control law, and
- A full-order observer

Plot the step response of the nonlinear system + observer when

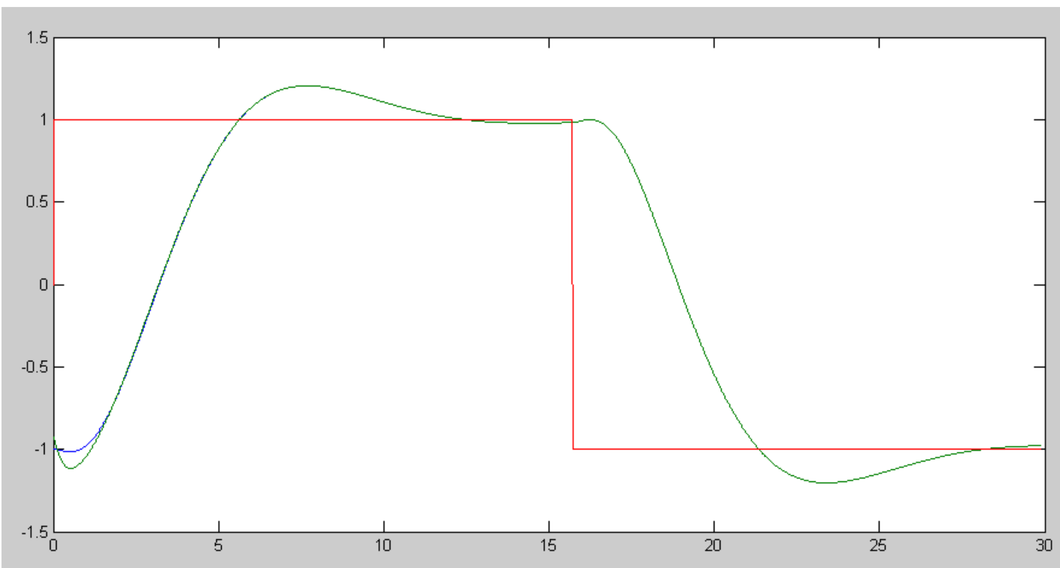
- $Xe = [0, 0, 0, 0]^T$
- $Xe = [0.1, 0.1, 0.1, 0.1]^T$



Feeding back X, initial conditions match

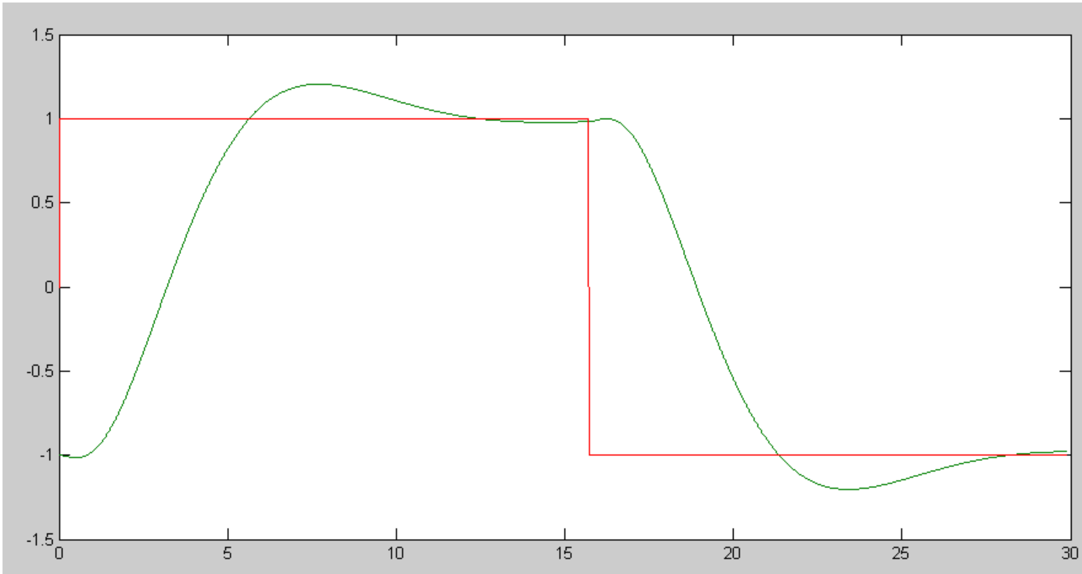


X = Xe = [0;0;0;0]

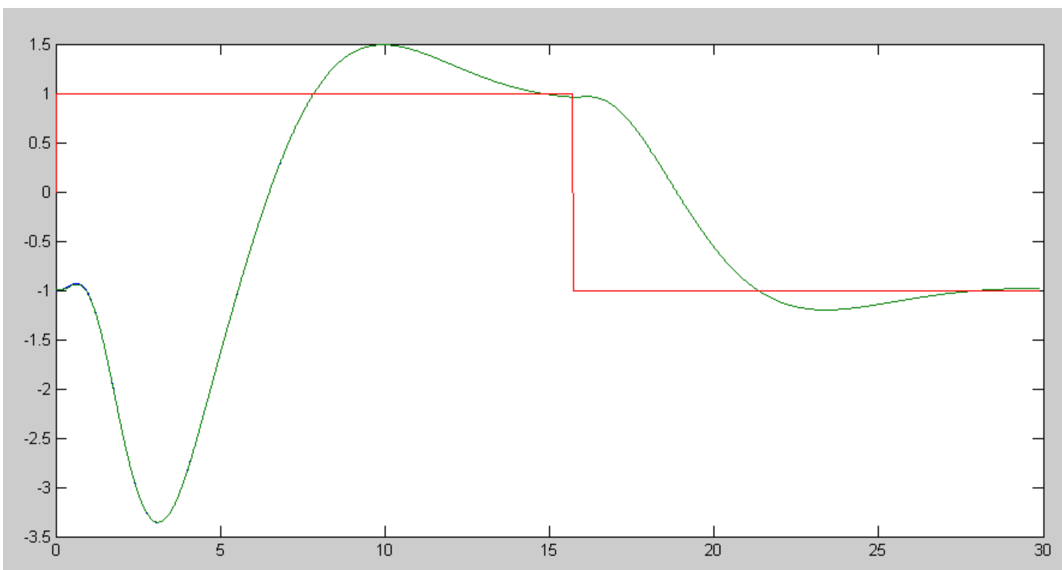


X = [-1,0,0,0]   Xe = X + 0.1*rand(4,1)

U = Kr R - Kx Xe



Xe(0) = X(0)



Xe(0) = X(0) + 0.01*rand(4,1)

```
% Cart and Pendulum
% Lecture %20
% Separation Principle

X = [-1,0,0,0]';
Ref = 1;
dt = 0.01;
t = 0;
Kx = [-0.3924  -68.6334   -1.1352  -12.4552];
Kr = -0.3924;
% Full-Order Observer
Ae = [0,0,1,0;0,0,0,1;0,-14.7,0,0;0,24.5,0,0];
Be = [0;0;0.5;-0.5];
Ce = [1,0,0,0];
H = ppl(Ae', Ce', [-1,-2,-3,-4])';
Xe = X + 0.01*rand(4,1);

n = 0;
y = [];
while(t < 29.9)
 Ref = sign(sin(0.2*t));
 U = Kr*Ref - Kx*Xe;
 dX = CartDynamics(X, U);
 dXe = Ae*Xe + Be*U + H*(X(1) - Ce*Xe);

 X = X + dX * dt;
 Xe = Xe + dXe * dt;

 t = t + dt;
 n = mod(n+1, 5);
 if(n == 0)
    CartDisplay(X, Xe, Ref);
 end
 y = [y ; X(1), Xe(1), Ref];
end

hold off;
t = [1:length(y)]' * dt;
plot(t,y);
```