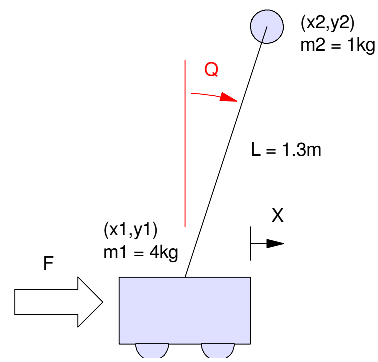
ECE 463/663 - Homework #6

Pole Placement. Due Monday, February 26th
Please submit as a hard copy, email to jacob.glower@ndsu.edu, or submit on BlackBoard

Problem 1) (30pt) Use the dynamics of a Cart and Pendulum System from homework set #4:

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



(10pt) Design a feedback control law of the form

U = Kr * R - Kx * X

so that the closed-loop system has

- A 2% settling time of 5 seconds, and
- 5% overshoot for a step input

Start with translating the requirements to pole location:

2% settling time of 5 seconds

The real part of the dominant pole should be at -0.8

5% overshoot for a step input

The damping ratio should be 0.6901 (2nd-order approximations)

The angle of the dominant pole should be 46.36 degrees $(\zeta = \cos\theta)$

The dominant pole should be at s = -0.8 + j0.8390

Place the poles of the closed-loop system at s = {-0.8 + j0.8390, -0.8 - j0.8390, -5, -6}

The last two poles are arbitrary - they don't affect the step response

In Matlab: Input the system dynamics:

```
>> A = [0,0,1,0;0,0,0,1;0,-2.45,0,0;0,9.42,0,0]

         0         0    1.0000         0
         0         0         0    1.0000
         0   -2.4500         0         0
         0    9.4200         0         0

>> B = [0;0;0.25;-0.1923]
         0
         0
    0.2500
   -0.1923
```

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

Find Kx to place the closed-loop poles

```
>> P = [-0.8 + j*0.8390, -0.8 - j*0.8390, -5, -6];
>> Kx = ppl(A, B, P)
```

Kx = -21.4015 -331.3277 -33.3268 -108.8492

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

  -6.0000
  -5.0000
  -0.8000 + 0.8390i
  -0.8000 - 0.8390i
```

Find Kr to set the DC gain to 1.000

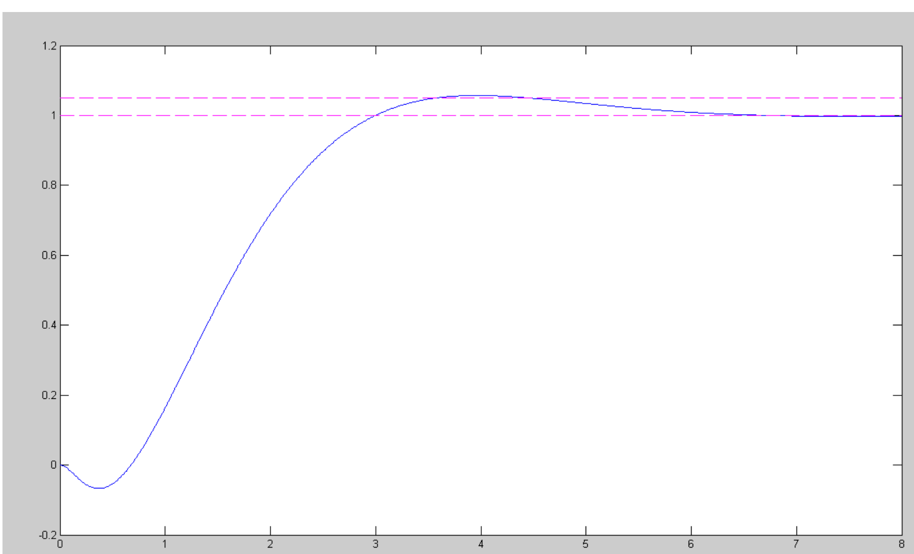
```
>> DC = -C*inv(A-B*Kx)*B

DC =   -0.0467

>> Kr = 1/DC
```

Kr = -21.4015

(10pt) Check the step response of the linear system in Matlab

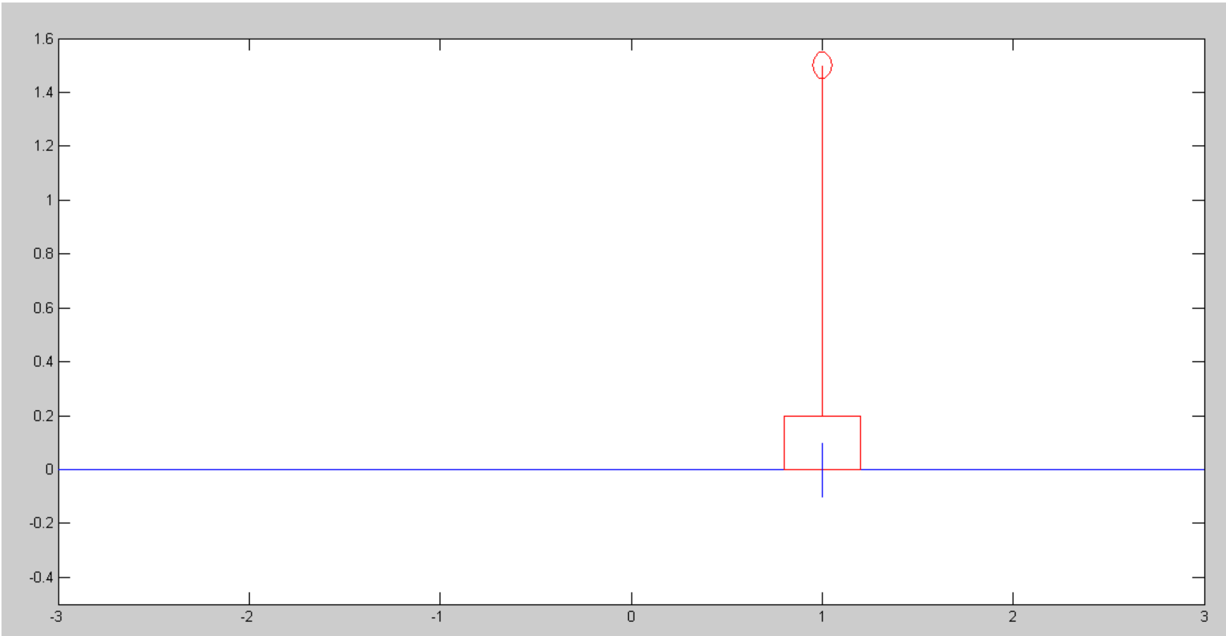
```
>> Gcl = ss(A-B*Kx, B*Kr, C, D);
>> t = [0:0.01:8]';
>> y = step(Gcl, t);
>> max(y)

ans =     1.0572

>> plot(t,y,t,0*y+1,'m--',t,0*y+1.05,'m--')
```

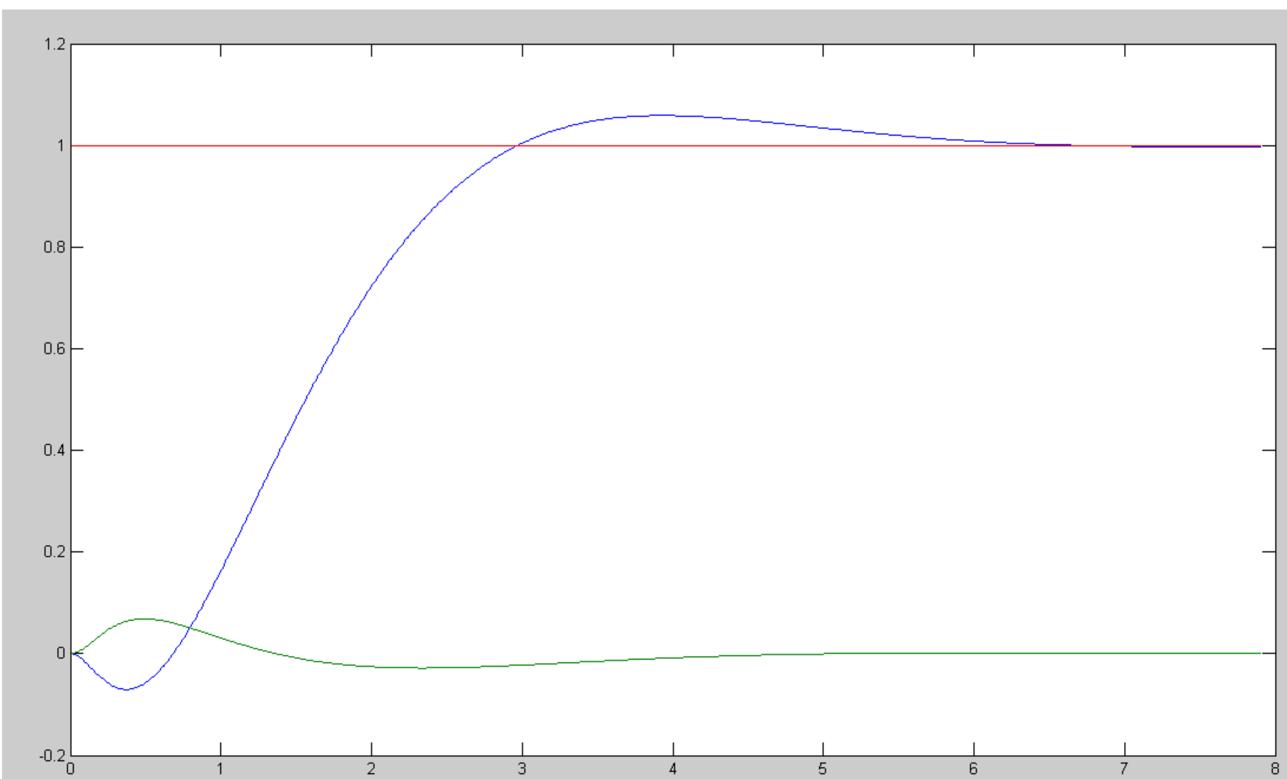
y(t) has 5.72% overshoot (should be 5.00%)



(10pt) Check the step response of the nonlinear system

- The results are almost identical to the linear system's response



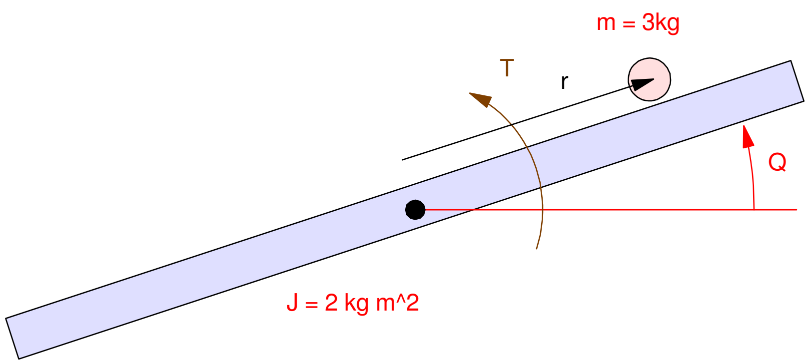


Matlab Code:

```
% Cart and Pendulum
% Homework #6: Pole Placememt

X = [0,0,0,0]';
Ref = 1;
dt = 0.01;
t = 0;
n = 0;
y = [];

Kx = [-21.4015 -331.3277  -33.3268 -108.8492];
Kr = -21.4015;


while(t < 15)
   Ref = 1;
   U = Kr*Ref - Kx*X;

   dX = CartDynamics(X, U);

   X = X + dX * dt;
   t = t + dt;

   n = mod(n+1, 5);
   if(n == 0)
      CartDisplay(X, Ref);
     end
   y = [y ; X(1), X(2), Ref];
   end

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

Problem 2) (30pt) Use the dynamics for the Ball and Beam system from homework set #4.

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



(10pt) Design a feedback control law so that the closed-loop system has

- A 2% settling time of 5 seconds, and
- 5% overshoot for a step input

Similar to problem #1, input the dynamics:

```
>> A = [0,0,1,0;0,0,0,1;0,-7,0,0;-5.88,0,0,0]

         0         0    1.0000         0
         0         0         0    1.0000
         0   -7.0000         0         0
   -5.8800         0         0         0

>> B = [0;0;0;0.2]

         0
         0
         0
    0.2000

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

Find Kx to place the closed-loop poles

```
>> P = [-0.8 + j*0.8390, -0.8 - j*0.8390, -5, -6];
>> Kx = ppl(A, B, P)

Kx =  -58.1983  244.7196  -44.8451   63.0000
```

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

  -6.0000
  -5.0000
  -0.8000 + 0.8390i
  -0.8000 - 0.8390i
```

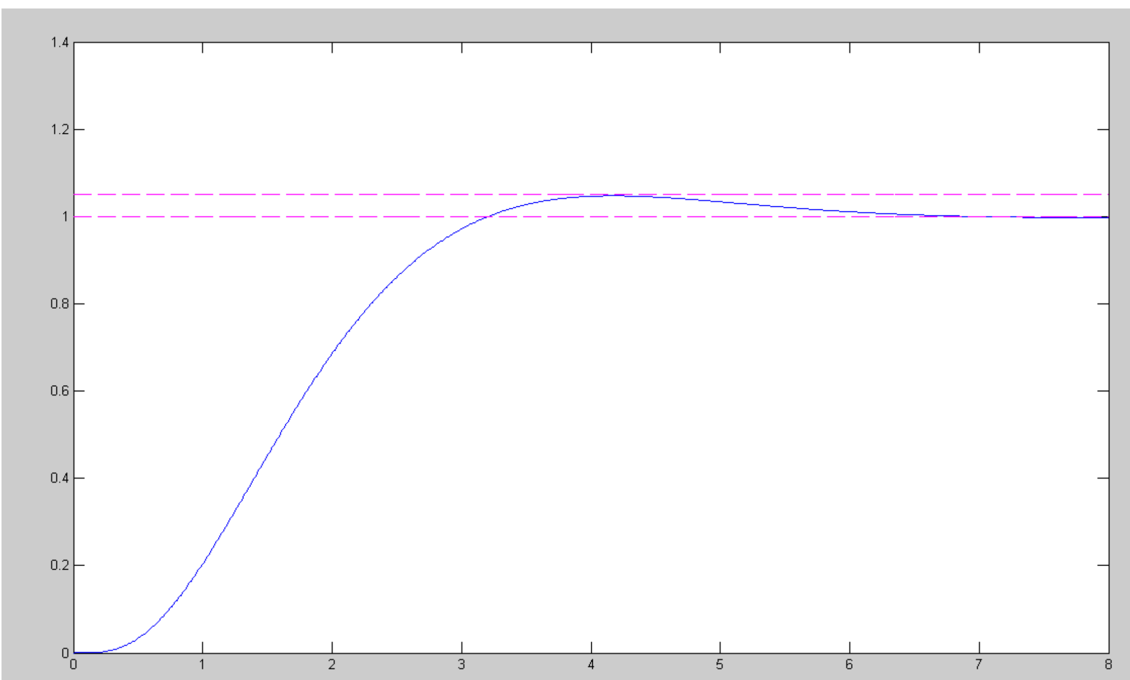
Find Kr to set the DC gain to 1.000

```
>> DC = -C*inv(A-B*Kx)*B

DC =   -0.0347

>> Kr = 1/DC

Kr =  -28.7983
```

>> (10pt) Check the step response of the linear system in Matlab

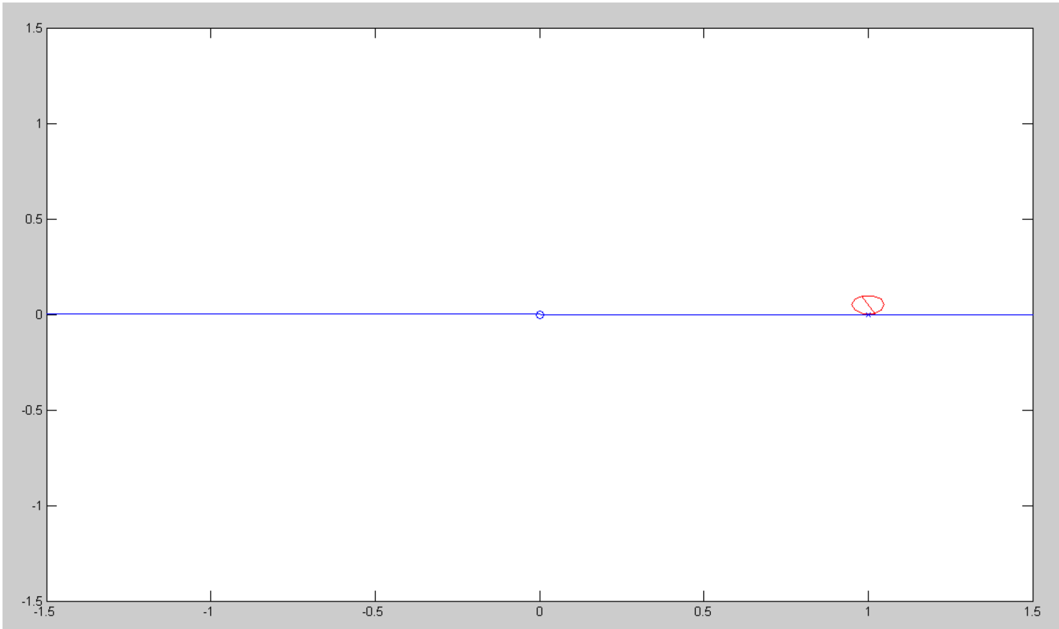
```
>> Gcl = ss(A-B*Kx, B*Kr, C, D);
>> t = [0:0.01:8]';
>> y = step(Gcl, t);
>> plot(t,y,t,0*y+1,'m--',t,0*y+1.05,'m--')
>> max(y)

ans =    1.0473
```

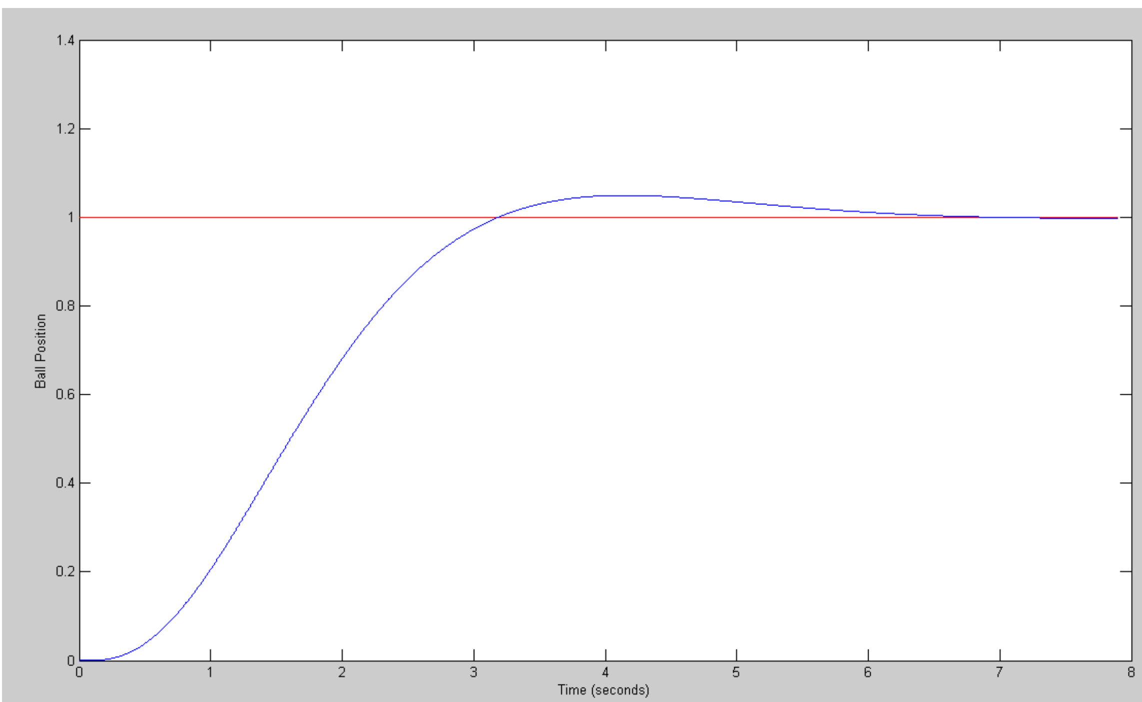
y(t) has 4.73% overshoot (should be 5.00%)



(10pt) Check the step response of the nonlinear system

- The step response is almost identical to the linear system
- It's pretty slow - the nonlinearities have little impact at this speed

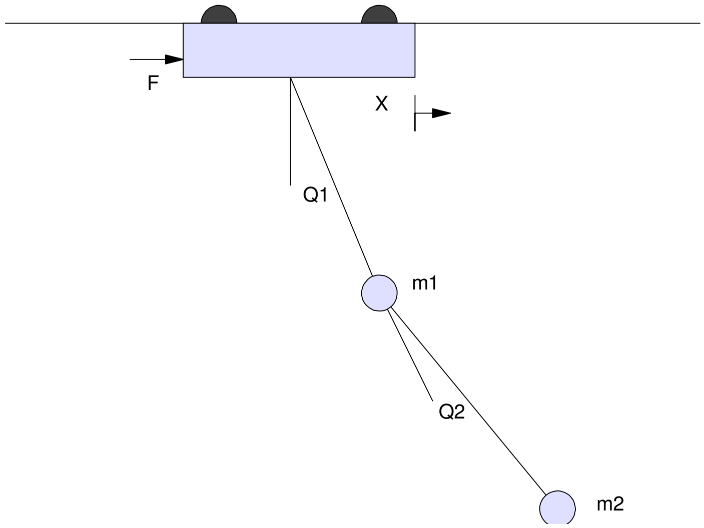




Matlab Code

```
% Ball & Beam System
% Spring 2024
% Homework #6

X = [0, 0, 0, 0]';
dt = 0.002;
t = 0;
n = 0;
y = [];

Kx = [  -58.1983  244.7196  -44.8451   63.0000];
Kr = [-28.7983];


while(t < 15)
   Ref = 1;
   U = Kr*Ref - Kx*X;
   dX = BeamDynamics(X, U);

   X = X + dX * dt;
   t = t + dt;

   y = [y ; Ref, X(1)];
   n = mod(n+1,5);
   if(n == 0)
      BeamDisplay(X, Ref);
      end
   end

t = [1:length(y)]' * dt;


plot(t,y(:,1),'r',t,y(:,2),'b');
xlabel('Time (seconds)');
ylabel('Ball Position');
```

Problem #3 (30pt): The dynamics of a double gantry (Gantry2) are



$$s\begin{bmatrix} x \\ \theta_1 \\ \theta_2 \\ \dot{x} \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 2g & 0 & 0 & 0 & 0 \\ 0 & -3g & g & 0 & 0 & 0 \\ 0 & 3g & -3g & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \theta_1 \\ \theta_2 \\ \dot{x} \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ -1 \\ 1 \end{bmatrix} F$$

(10pt) Design a feedback control law of the form

U = Kr * R - Kx * X

so that the closed-loop system has

- A 2% settling time of 10 seconds, and
- 5% overshoot for a step input

```
>> Z = zeros(3,3);
>> I = eye(3,3);
>> g = 9.8;
>> K = [0,2,0;0,-3,1;0,3,-3]*g;
>> A = [Z,I ; K,Z]

         0         0         0    1.0000         0         0
         0         0         0         0    1.0000         0
         0         0         0         0         0    1.0000
         0   19.6000         0         0         0         0
         0  -29.4000    9.8000         0         0         0
         0   29.4000  -29.4000         0         0         0
```

```
>> eig(A)

        0
        0
   0.0000 + 6.8099i
   0.0000 - 6.8099i
   0.0000 + 3.5250i
   0.0000 - 3.5250i

>> B = [0;0;0;1;-1;1]

     0
     0
     0
     1
    -1
     1

>> rank([B,A*B,A^2*B,A^3*B,A^4*B,A^5*B])

ans =     6

>> P = [-0.4+j*0.42,-0.4-j*0.42,-2,-3,-4,-5];
>> Kx = ppl(A, B, P)
```

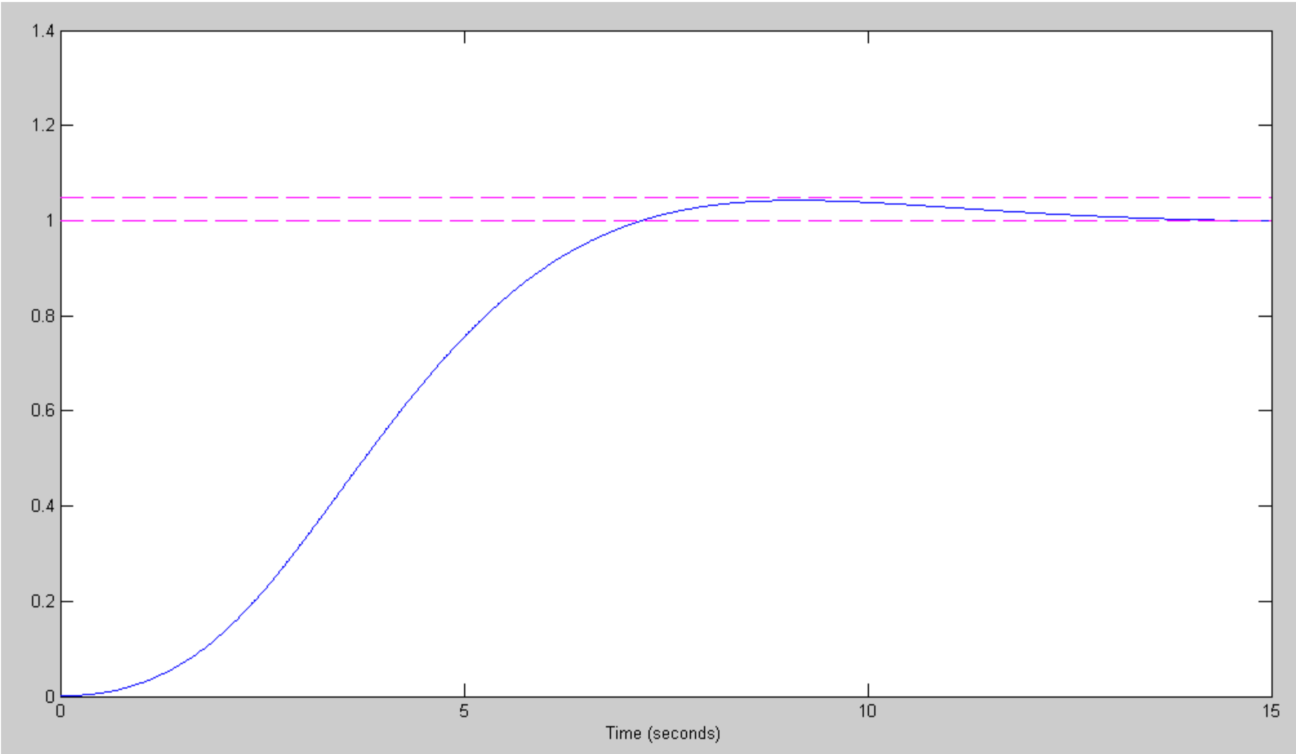
Kx =   0.2102   16.1936   39.7198   0.7695   -9.4564   4.5741

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

  -5.0000
  -4.0000
  -3.0000
  -2.0000
  -0.4000 + 0.4200i
  -0.4000 - 0.4200i

>> C = [1,0,0,0,0,0];
>> D = 0;
>> DC = -C*inv(A - B*Kx)*B;
>> Kr = 1/DC
```

Kr =   0.2102

(10pt) Determine the step response of the linear system in Matlab

```
>> Gcl = ss(A-B*Kx, B*Kr, C, D);
>> t = [0:0.01:15]';
>> y = step(Gcl,t);
>> plot(t,y,t,0*t+1,'m--',t,0*t+1.05,'m--')
>> xlabel('Time (seconds)');
```




```
>>
```

(10pt) Determine the step response of the nonlinear system

```
X = [0, 0, 0, 0, 0, 0]';
Ref = 1;
dt = 0.01;
U = 0;
t = 0;
Kx = [ 0.2102   16.1936   39.7198    0.7695   -9.4564    4.5741];
Kr = 0.2102;
y = [];
n = 0;
while(t < 15)
   Ref = 1;
   U = Kr*Ref - Kx*X;
   dX = Gantry2Dynamics(X, U);
   X = X + dX * dt;
   t = t + dt;
   n = mod(n+1,5);
   if(n == 0)
      Gantry2Display(X, Ref);
      plot([Ref, Ref],[-0.1,0.1],'b');
      end
   y = [y ; Ref, X(1), X(2), X(3)];
end

pause(2);
t = [1:length(y)]' * dt;
plot(t,y);
xlabel('Time (seconds)');
```