ABDULLAH ALMOSALAMI

ECE 463

MODERN CONTROL

03/23/2021

TUESDAY

Using the Cart & Pendulum system from HW~~#~~4 ...

$m_{CART} = 1kg$

$m_{ball} = 4kg$

$L = 1.0m$

1) Design a full-state feedback control law:

$$u = K_r R - \underline{K_x}\, \underline{X}$$

so that $2\% T_s = 6s$, 10% MOS.

Plot the system response.

↓ I've got my linearized CartPend function...

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -39.2 & 0 & 0 \\ 0 & 49 & 0 & 0 \end{bmatrix}$$

I've got my ObtainCartFeedback function, so:

I place the poles at: $[-0.667 \pm j0.9097, -3.3333, -4.3333]$

$$K_x = [-1.8748, -76.8134, -2.9603, -11.9603] \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \\ -1 \end{bmatrix}$$

$$K_r = -1.8748$$

↑ For $x(t)$

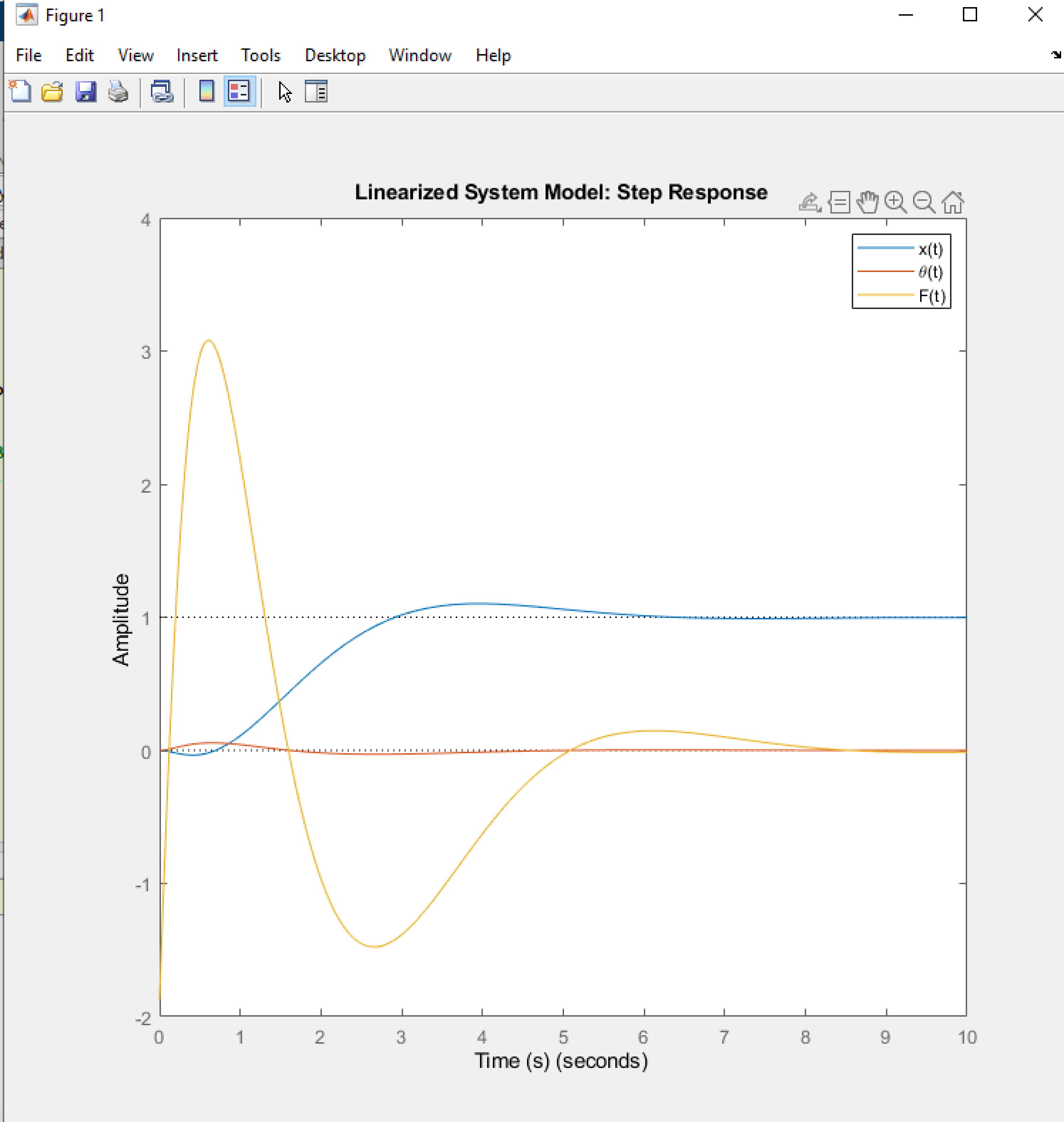


Figure 1
File Edit View Insert Tools Desktop Window Help
Linearized System Model: Step Response
x(t)
θ(t)
F(t)
Amplitude
4
3
2
1
0
-1
-2
0 1 2 3 4 5 6 7 8 9 10
Time (s) (seconds)

2) Assume you can only measure the cart position $x(t)$ and beam angle $\theta(t)$.

(a) Design a full-order observer to estimate all four states 2-5 times faster than the plant.

1) Check observability: - Let's say $x(t)$ is the output:
$$C = \begin{array}{cccc} x & \theta & \dot{x} & \dot{\theta} \\ \hline 1 & 0 & 0 & 0 \end{array}$$

Observability matrix: $[C*A^0; C*A; C*A^2; C*A^3]$

$\downarrow$

rank $\rightarrow$ 4 ✓

- $\theta(t)$: rank $\rightarrow$ 2 ; so NOT possible w/ just θ...

2) So now we Bass Gura on A^T, C^T, and H = place Poles (A^T, C^T, des_p)
To make the observer 5x faster, let's go w/ $[-3.4, -4, -5, -6]$.
So:

$$\boxed{H = [18.4, -32.4796, 174, -227.9082]}$$

If I used instead x2 faster: $[-2, -3, -4, -5]$

$\Rightarrow H = 14, -21, 120, -153\ldots$
ehh... not much better

3) Augmented system:

$$A_{aug} = \left[\begin{array}{cccc|cccc} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -39.2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 49 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 18.4 & 0 & 0 & 0 & -18.4 & 0 & 1 & 0 \\ -32.4 & 0 & 0 & 0 & 32.4 & 0 & 0 & 1 \\ 174 & 0 & 0 & 0 & -174 & -39.2 & 0 & 0 \\ -227.9 & 0 & 0 & 0 & 227.9 & 49 & 0 & 0 \end{array}\right] \quad B_{aug} = \left[\begin{array}{c} 0 \\ 0 \\ 1 \\ -1 \\ \hline 0 \\ 0 \\ 1 \\ -1 \end{array}\right]$$

(b) Give state-space model using actual states and plot the step response:

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

$$u = K_r R - K_x \underset{\approx}{X}$$

If we're using the actual states, using some algebra... (next page)

$$s\begin{bmatrix} x \\ \hat{x} \end{bmatrix} = \begin{bmatrix} A - BK_x & 0 \\ HC - BK_x & A - HC \end{bmatrix} \begin{bmatrix} x \\ \hat{x} \end{bmatrix} + \begin{bmatrix} BK_r \\ BK_r \end{bmatrix} R$$

$$sX = AX + BU$$

$$s\hat{X} = A\hat{X} + BU + H(Y - \hat{Y}) = A\hat{X} + BU + H(CX - C\hat{X})$$
$$= (A - HC)\hat{X} + BU + HCX$$

$$\boxed{U = K_r R - K_x X}$$

$$sX = AX + BK_r R - BK_x X = (A - BK_x)X + BK_r R$$

$$s\hat{X} = (A - HC)\hat{X} + BK_r R - BK_x X + HCX$$
$$= (A - HC)\hat{X} + (HC - BK_x)X + (BK_r)R$$

$$s\begin{bmatrix} X \\ \hat{X} \end{bmatrix} = \begin{bmatrix} A - BK_x & 0 \\ HC - BK_x & A - HC \end{bmatrix}\begin{bmatrix} X \\ \hat{X} \end{bmatrix} + \begin{bmatrix} BK_r \\ BK_r \end{bmatrix} R$$

$$\boxed{U = K_r R - K_x \hat{X}}$$

$$sX = AX + BK_r R - BK_x \hat{X} = AX + (-BK_x)\hat{X} + (BK_r)R$$

$$s\hat{X} = (A - HC)\hat{X} + BK_r R - BK_x \hat{X} + HCX$$
$$= (A - HC - BK_x)\hat{X} + (HC)X + (BK_r)R$$

$$s\begin{bmatrix} X \\ \hat{X} \end{bmatrix} = \begin{bmatrix} A & -BK_x \\ HC & A - HC - BK_x \end{bmatrix}\begin{bmatrix} X \\ \hat{X} \end{bmatrix} + \begin{bmatrix} BK_r \\ BK_r \end{bmatrix} R$$

And I'm interested in looking at $x(t)$ and $\hat{x}(t)$, so

$$C_{aug} = [C, 0*C ; 0*C, C]$$

$$D_{aug} = [0;0]$$

(c) Now use the observer's state: $u = K_r R - \underline{K_x} \hat{\underline{X}}$

Here,

$$s \begin{bmatrix} X \\ \hat{X} \end{bmatrix} = \begin{bmatrix} A & -BK_x \\ HC & A - HC - BK_x \end{bmatrix} \begin{bmatrix} X \\ \hat{X} \end{bmatrix} + \begin{bmatrix} BK_r \\ BK_r \end{bmatrix} R$$

Same C_{aug}, D_{aug} ... And:

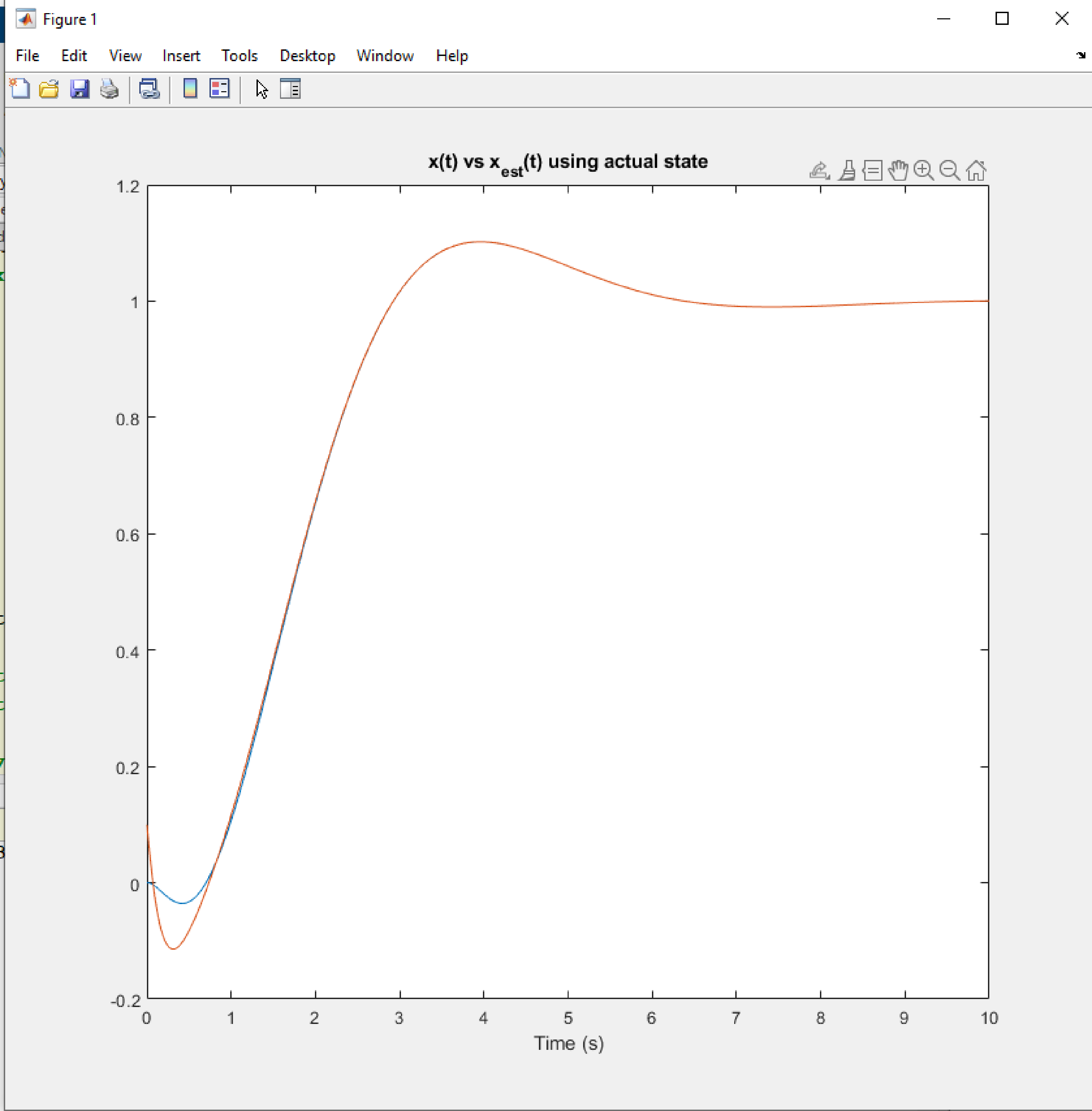


Figure 1
File Edit View Insert Tools Desktop Window Help
x(t) vs x_est(t) using actual state
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
0 1 2 3 4 5 6 7 8 9 10
Time (s)

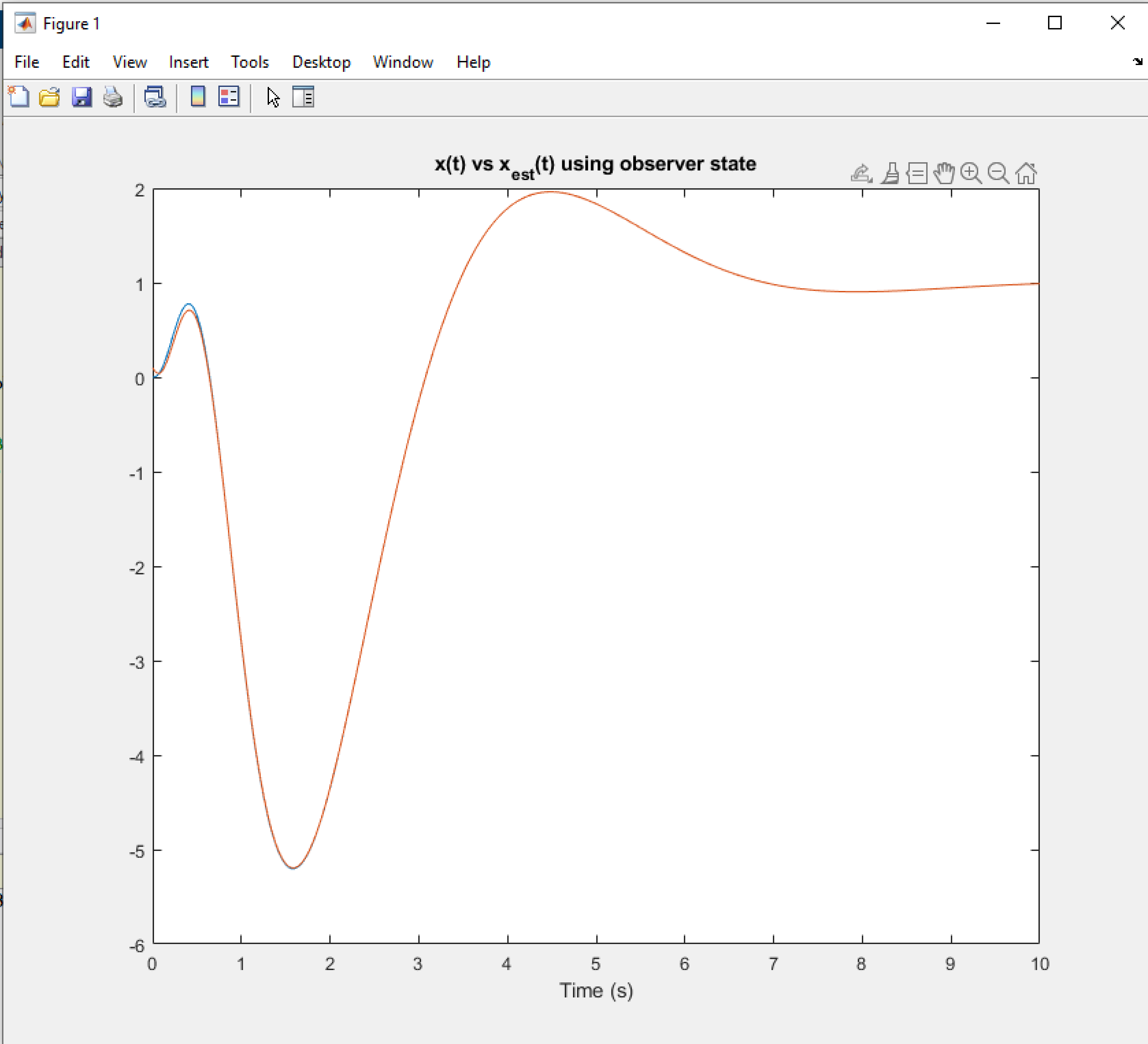


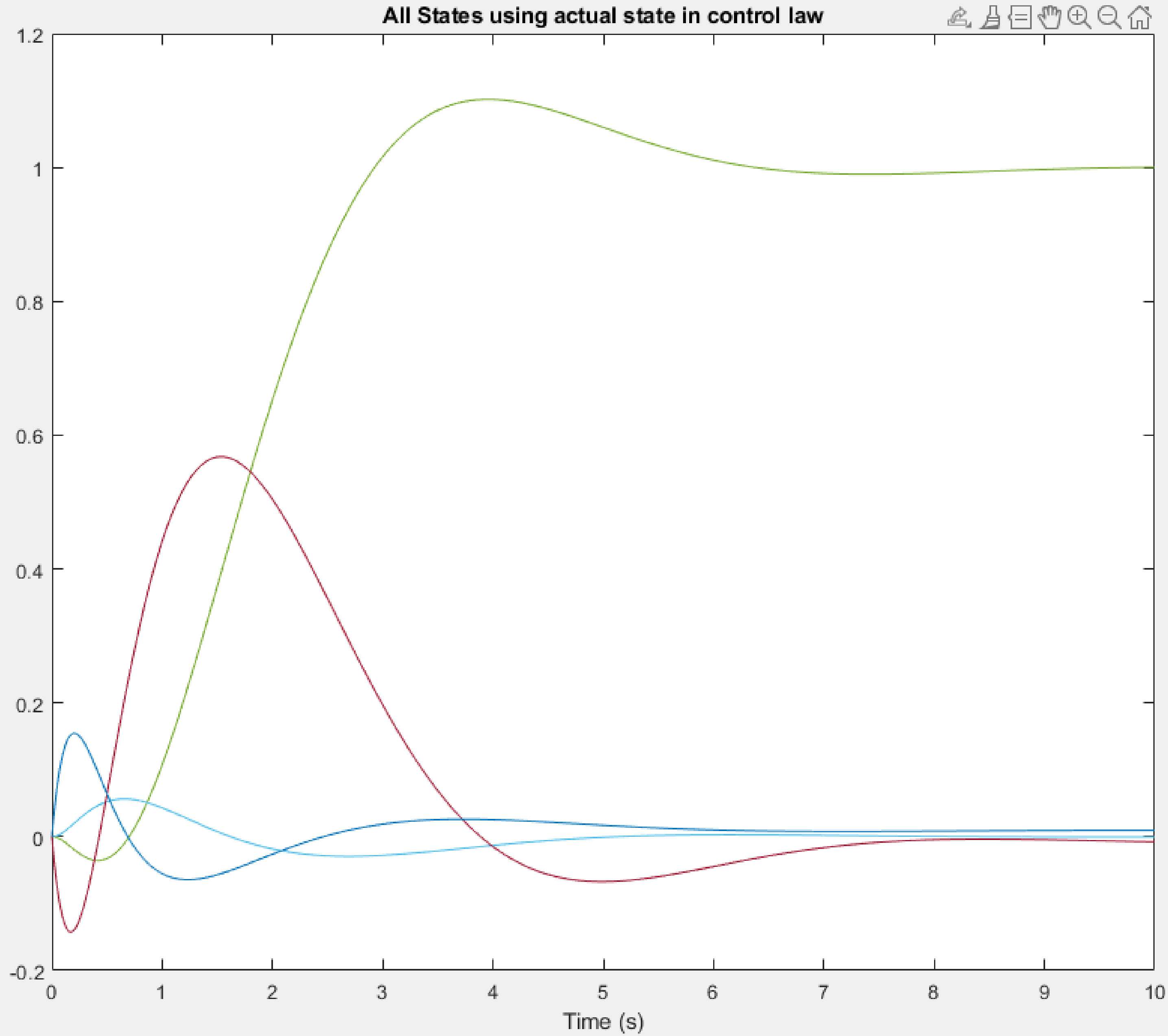
Figure 1
File Edit View Insert Tools Desktop Window Help
x(t) vs x_est(t) using observer state
2
1
0
-1
-2
-3
-4
-5
-6
0 1 2 3 4 5 6 7 8 9 10
Time (s)

3) Modify the cart & pendulum system to include your control law and a full order observer using only cart position $x(t)$ and beam angle $\theta(t)$.
Find the step response when u is $K_r R - K_x X$ and $K_r R - K_x \hat{X}$ w/ initial cond. $X(0) = [0,0,0,0]$ and $\hat{X}(0) = [0,0,0,0]$.

Not sure how this is different from 2b and 2c... Let's just have all states printed...:

Figure 1

File Edit View Insert Tools Desktop Window Help



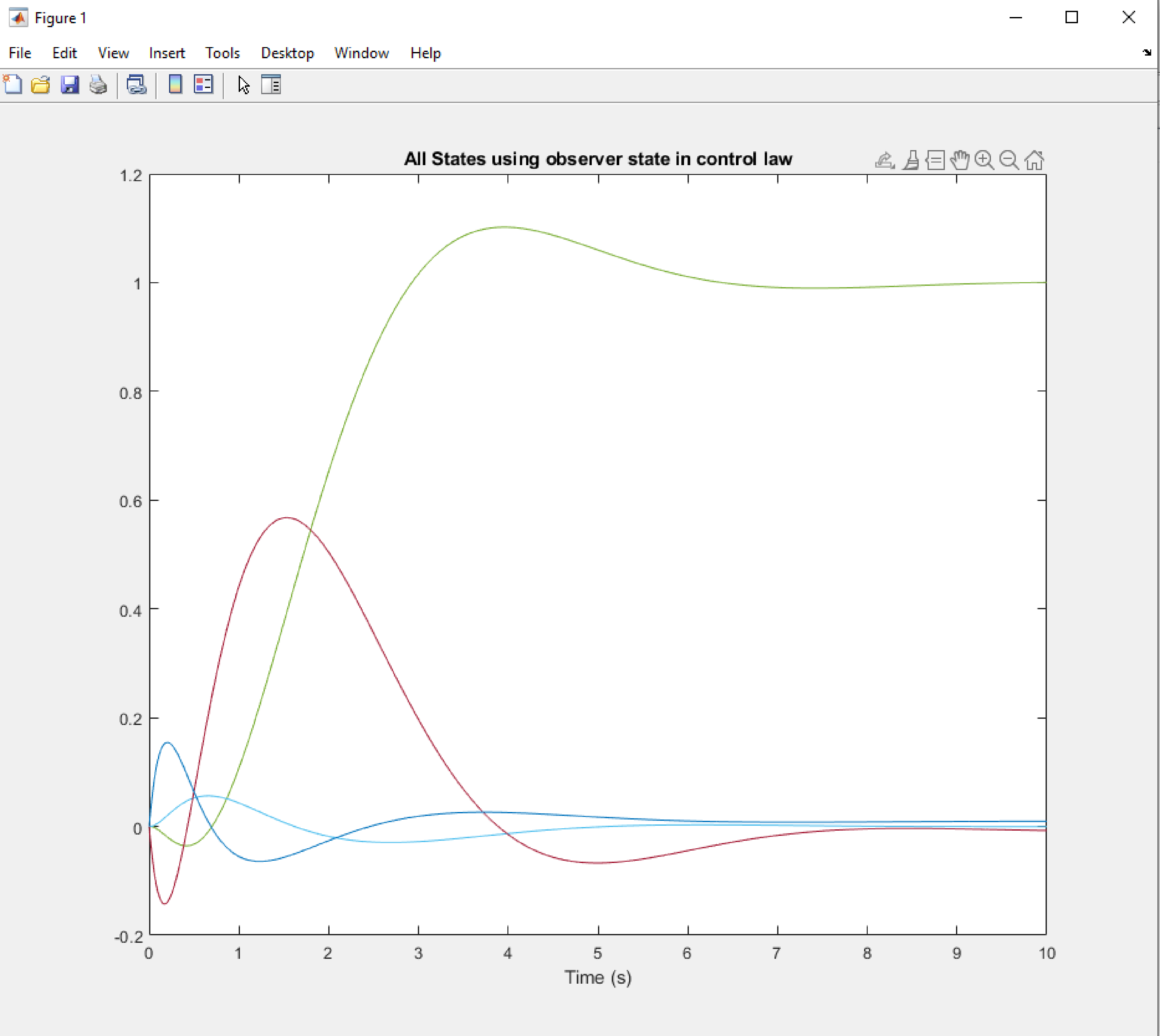


Figure 1
File
Edit
View
Insert
Tools
Desktop
Window
Help
All States using observer state in control law
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
0
1
2
3
4
5
6
7
8
9
10
Time (s)

```
function [Kx, Kr] = placePoles(A, B, C, poles)

    sz = size(A);
    N = sz(1);
    % Controllability matrix
    T1 = zeros(N,N);
    for i=0:N-1
        T1(:,i+1) = (A^i) * B;
    end

    if N <= 1
        error('System needs to have an order greater than 1.');
    elseif rank(T1) ~= N
        error('System is not controllable');
    end

    if isControllerCanonical(A)
        Pd = poly(poles);
        P = poly(eig(A));
        dP = Pd - P;
        Kx = dP([N+1:-1:2]);

    else
        P = poly(eig(A));
        % T1 is controllability matrix [B AB A^2B ... A^N-1B]
        T1 = zeros(N,N);
        for i=0:N-1
            T1(:,i+1) = (A^i) * B;
        end
        % T2 involves the characteristic polynomial coefficients
of A
        T2 = zeros(N,N);
        for i=0:N-1
            T2(i+1,:) = [zeros(1,i), P(1:end-i-1)];
        end
        % Swap rows of identity matrix to get T3
        T3 = eye(N);
        for i=1:floor(N/2)
            ii = T3(i,:);
            T3(i,:) = T3(N-i+1,:);
            T3(N-i+1,:) = ii;
        end
        % Similarity transform matrix
        T = T1*T2*T3;

        % Obtain state and input matrices of similar system
        Az = inv(T)*A*T;
```

```
        Bz = inv(T)*B;

        % Now obtain Kz needed for similar system
        Pd = poly(poles);
        dP = Pd - P;
        Kz = dP([N+1:-1:2]);

        % Perform similarity transform to obtain corresponding
Kx
        Kx = Kz*inv(T);
    end

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

end
```

```
function [Kx, Kr] = ObtainCartFeedback(mcart,mball,L, Ts, MOS)

% m1 = 1; m2 = 4; R = 1;
[A,B] = linearizedCartPend(mcart,mball,L) %#ok<NOPRT>
sig = getSigForTs(Ts);
[zeta, th, w] = getForOS(MOS,sig); %#ok<ASGLU>
dom_poles = [sig+1j*w, sig-1j*w];
des_poles = [dom_poles, 5*sig, 5*sig-1] %#ok<NOPRT>

C1 = [1 0 0 0]; % Position of cart
C2 = [0 1 0 0]; % Angle of bar
C = C1;
% olPoles = flip(eig(A));

[Kx,Kr] = placePoles(A,B,C,des_poles) %#ok<NOPRT>

G1 = ss(A-B*Kx, B*Kr, C1, 0);
G2 = ss(A-B*Kx, B*Kr, C2, 0);
G = G1;
U = ss(A-B*Kx, B*Kr, -Kx, Kr);

step(G);
hold on;
step(G2);
step(U);
legend('x(t)','\theta(t)','F(t)');
xlabel('Time (s)'); title('Linearized System Model: Step
Response');
hold off;

end
```

```
function [A,B] = linearizedCartPend(mcart, mball, L)
g = 9.8;
A = [0 0 1 0; 0 0 0 1; 0 (-mball*g/mcart) 0 0; 0
((mcart+mball)*g/mcart) 0 0];
B = [0;0;1/mcart;-1/(mcart*L)];
end
```