MECH 513
Assignment 4
Nick Speal

Originally submitted on Nov 25, 2013. Modified on Nov 6, 2014.

Notes:

An excerpt from this assignment is included below as a part of my portfolio. This document begins with a copy of the full assignment, followed by the solution to question 3. For each part, the question is repeated, followed by the code, the console output, and then any plots and discussion.

The MATLAB code can be found at github.com/nickspeal.

Assignment 4
Control Systems, MECH 513, Fall 2013
(due November 25, IN CLASS)

The following problems are all from Williams and Lawrence 'Linear State-Space Control Systems'

1. Problems from Chapter 7:
   CME7.3c
2. Problems from Chapter 8:
   NE8.2d, NE8.3d, CME8.3, CE8.3 (choose one of three cases to plot)

The following problem must also be solved:

3. Consider the system

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 \\ a_{21} & 1 \end{bmatrix}\mathbf{x} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}u$$
$$y = \begin{bmatrix} 1 & 0 \end{bmatrix}\mathbf{x}$$

which is to track a constant reference, $r = 2$. Nominally $a_{21} = 2$. For any simulations, let $\mathbf{x}(0) = [1\ 1]^T$ and $\hat{\mathbf{x}} = [0\ 0]^T$ and simulate from zero to 15 seconds.

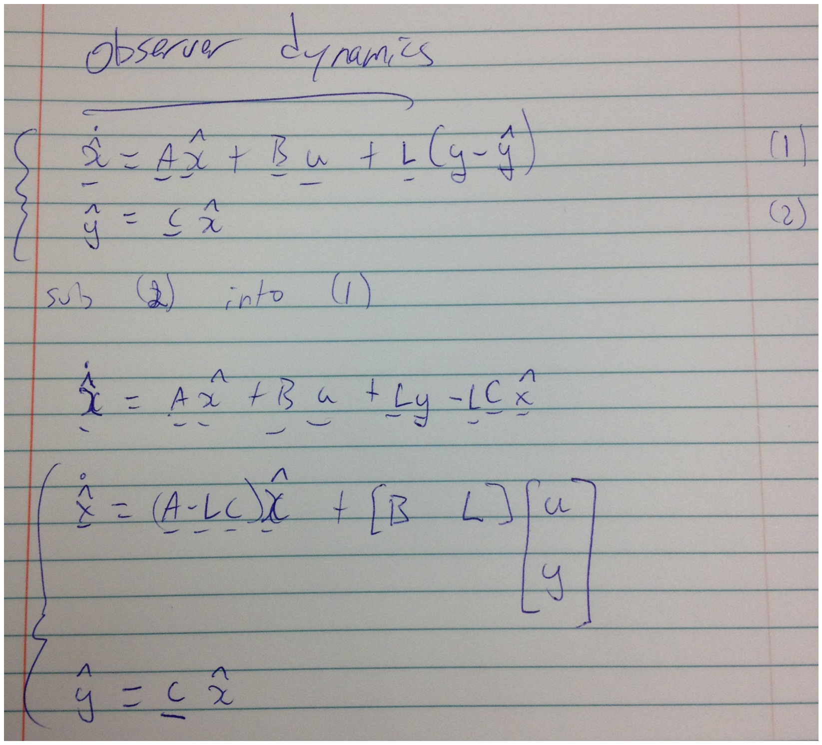
a) Letting u = -**Kx** + Gr, find **K** and G such that the closed-loop eigenvalues are at -1 and -2, and asymptotic tracking is achieved. Verify that the closed-loop DC gain of the system is one.
b) Next design an observer, placing the observer eigenvalues at -4 and -5. Write out the observer dynamics equation explicitly.
c) Combine the results from part a) and b) and verify that the DC gain is one. Simulate the system response; specifically, plot the state variables and their estimated values and the control input, *u*.
d) Now, assume that the real value for $a_{21} = 2.1$ (initial model contained a small error). Simulate the closed-loop response of the system and plot the state variables and their estimated values and the control input, *u*. Comment on the results.
e) Augment the original nominal system with integral action and design a feedback control law with closed-loop eigenvalues at -1, -2, -3. Find the transfer function from *r* to *y* and confirm that asymptotic tracking is achieved.
f) Considering the perturbed system where $a_{21} = 2.1$, and the same controller as in e), verify that the closed-loop system is still asymptotically stable, and that asymptotic tracking is still achieved despite the perturbation.
g) Using the same observer as in b), find the DC gain of the closed-loop transfer function. Verify that asymptotic tracking is still achieved and simulate the response with and without the perturbation; specifically, plot the state variables and their estimated values and the control input, *u*. Comment on the results. Is the controller robust?

a) Letting u = -Kx + Gr, find K and G such that the closed-loop eigenvalues are at -1 and -2, and asymptotic tracking is achieved. Verify that the closed-loop DC gain of the system is one.

```
% Assignmenet Question 3
% Nicholas Speal
% Assignment 4

clear
clc
close all

%% General System Parameters
a21 = 2;
r = 2;
A = [0 1; a21 1];
B = [0;1];
C = [1 0];
D = 0;

x0 = [1;1];
x0_est = [0;0];
t = [0:.01:15]';
r = 2*ones(size(t));

n = size(A);

%% A
disp('PART A')
des_eigs = [-1 -2];
K = place(A,B,des_eigs)
G = -inv(C*inv(A-B*K)*B)

A_cl = A-B*K;
B_cl = B*G;
sys_cl = ss(A_cl,B_cl,C,D);

[y t x] = lsim(sys_cl,r,t,x0);

DC_gain_for_part_a = y(end)/r(end)
```

```
PART A
K =
   4   4
G =
   2
DC_gain_for_part_a =
   1.0000
```

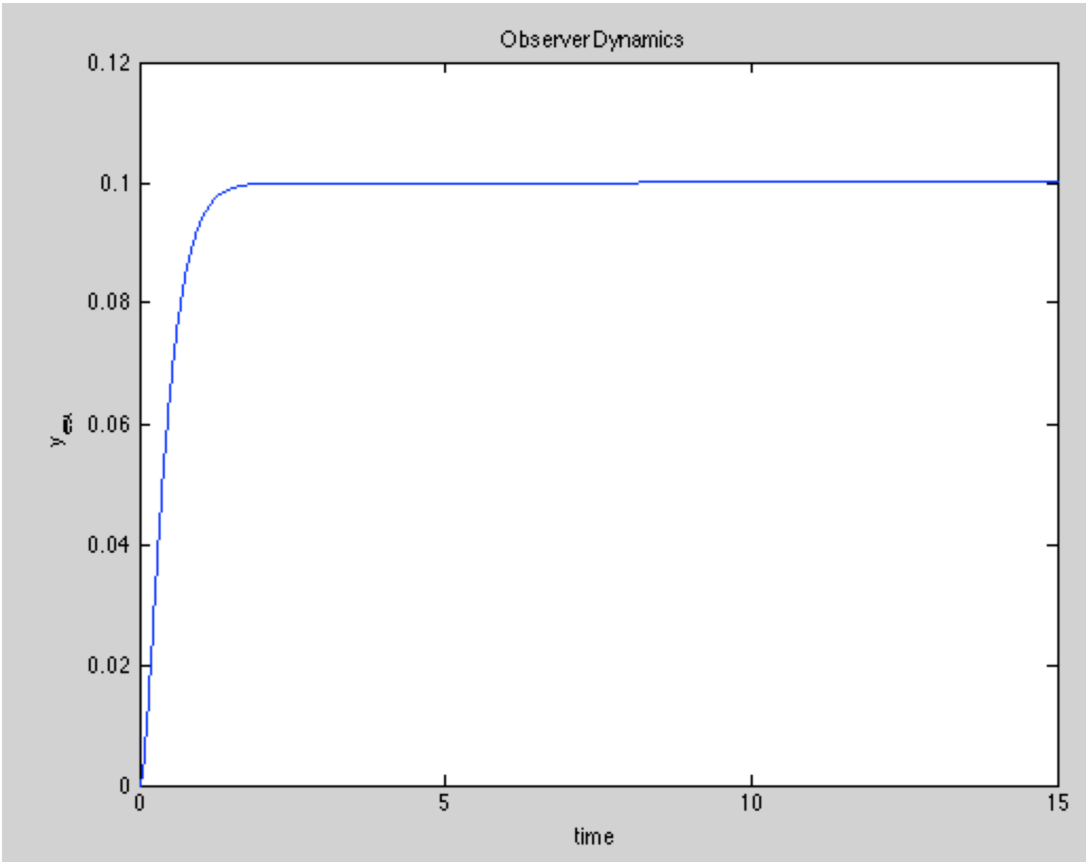
b) Next design an observer, placing the observer eigenvalues at -4 and -5. Write out the observer dynamics equation explicitly.




```
%% Part B
disp('PART B')
des_eig_obs = [-4 -5];
L = place(A',C',des_eig_obs)'

% observer dynamics equation:
A_obs = A-L*C;
B_obs = [B L];
U_obs = [r zeros(size(t))];

observer_system = ss(A_obs,B_obs,C,D);
[y_est t x_est] = lsim(observer_system, U_obs, t, x0_est);

figure
plot(t, y_est); xlabel('time'); ylabel('y_{est}');
title('Observer Dynamics');
```

PART B

L =
  10.0000
  32.0000

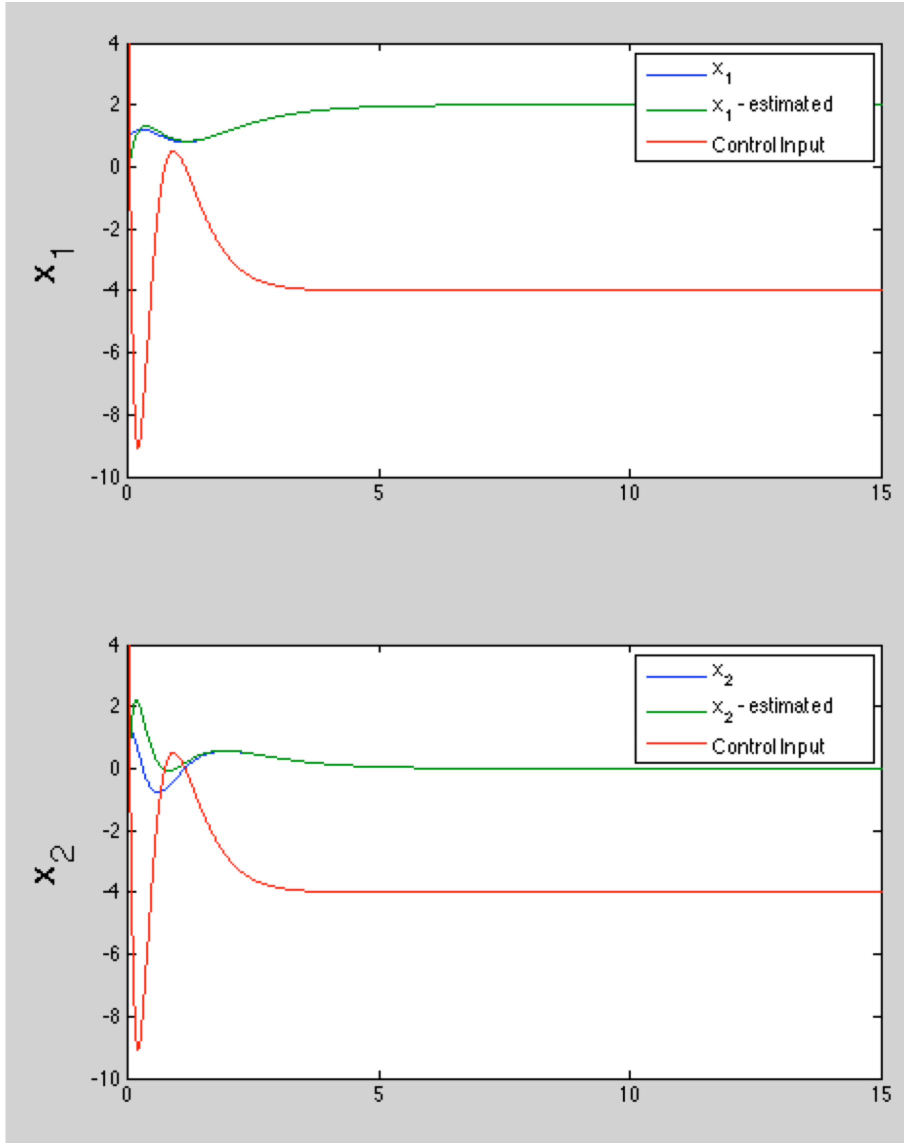

Observer Dynamics
0.12
0.1
0.08
0.06
0.04
0.02
0
5
10
15
time

c) Combine the results from part a) and b) and verify that the DC gain is one. Simulate the system response; specifically, plot the state variables and their estimated values and the control input, *u*.

```
%% PART C
disp('PART C')
% Full state feedback system with observer
A_obs_fb = [A -B*K; L*C A-B*K-L*C];
B_obs_fb = [G*B; G*B];
C_obs_fb = [C zeros(1,2)];
D_obs_fb = 0;
sys_obs_fb = ss(A_obs_fb, B_obs_fb, C_obs_fb, D_obs_fb)

[y, t, x] = lsim(sys_obs_fb, r, t, [x0; x0_est]);

DC_gain_for_part_C = y(end)/r(end)

%control input for each variable:
U = -K*x(:,3:4)' +G*r';      %transposes introduced to make dimensions
agree
%ctrlInput = B_obs_fb*U; %not sure...
ctrlInput = [1;1]*U;


figure
subplot(211)
    plot(t,x(:,1),t,x(:,3),t,ctrlInput(1,:))
    ylabel('x_1','fontsize',20),legend('x_1','x_1 - estimated','Control
Input')
subplot(212)
    plot(t,x(:,2),t,x(:,4),t,ctrlInput(2,:))
    ylabel('x_2','fontsize',20),legend('x_2','x_2 - estimated','Control
Input')
```

PART C

```
sys_obs_fb =

  a =
       x1   x2   x3   x4
   x1    0    1    0    0
   x2    2    1   -4   -4
   x3   10    0  -10    1
   x4   32    0  -34   -3

  b =
       u1
   x1   0
   x2   2
   x3   0
   x4   2

  c =
       x1  x2  x3  x4
   y1   1   0   0   0

  d =
       u1
   y1   0
```



Continuous-time state-space model.

```
DC_gain_for_part_C =

    1.0000
```

d) Now, assume that the real value for a_{21} = 2.1 (initial model contained a small error). Simulate the closed-loop response of the system and plot the state variables and their estimated values and the control input, *u*. Comment on the results.

```
%% PART D
disp('PART D')
a21 = 2.1;
A = [0 1; a21 1];

% Full state feedback system with observer
A_obs_fb = [A -B*K; L*C A-B*K-L*C];
B_obs_fb = [G*B; G*B];
C_obs_fb = [C zeros(1,2)];
D_obs_fb = 0;
sys_obs_fb = ss(A_obs_fb, B_obs_fb, C_obs_fb, D_obs_fb)

[y, t, x] = lsim(sys_obs_fb, r, t, [x0; x0_est]);

DC_gain_for_part_D = y(end)/r(end)

%control input for each variable:
U = -K*x(:,3:4)' +G*r';      %transposes introduced to make dimensions
agree
ctrlInput = [1;1]*U;
%ctrlInput = B_obs_fb*U; %not sure...



figure
subplot(211)
    plot(t,x(:,1),t,x(:,3),t,ctrlInput(1,:))
    ylabel('x_1','fontsize',20),legend('x_1','x_1 - estimated','Control
Input')
subplot(212)
    plot(t,x(:,2),t,x(:,4),t,ctrlInput(2,:))
    ylabel('x_2','fontsize',20),legend('x_2','x_2 - estimated','Control
Input')
```

PART D

sys_obs_fb =

a =

	x1	x2	x3	x4
x1	0	1	0	0
x2	2.1	1	-4	-4
x3	10	0	-10	1
x4	32	0	-33.9	-3

b =

	u1
x1	0
x2	2
x3	0
x4	2

c =

	x1	x2	x3	x4
y1	1	0	0	0

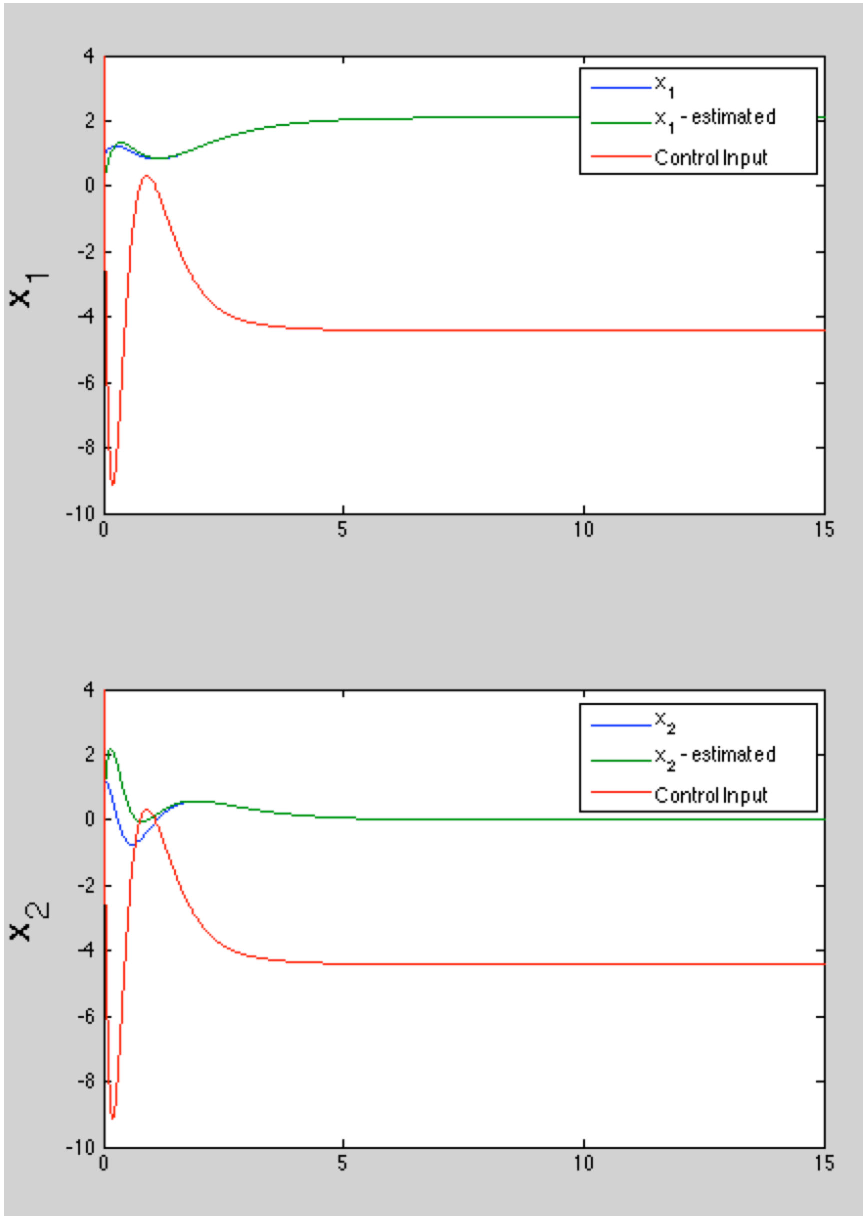
d =

	u1
y1	0

Continuous-time state-space model.

DC_gain_for_part_D =

1.0526



COMMENTS FOR PART D

The plots in part D are very similar to those in part C, indicating that the controller is robust to model perturbations.

There is an initial difference between the estimated and actual state, but it eventually converges because the observer is asymptotically stable.

e) Augment the original nominal system with integral action and design a feedback control law with closed-loop eigenvalues at -1, -2, -3. Find the transfer function from *r* to *y* and confirm that asymptotic tracking is achieved.

```
%% PART E
disp('PART E')
%restore nominal system
a21 = 2;
A = [0 1; a21 1];

%augment with integral action as a servomechanism
A_aug = [A zeros(n,1); -C 0];
B_aug = [B; 0];
DesEigs = [-1 -2 -3];
K_total = place(A_aug,B_aug, DesEigs)
    k = K_total(1:end-1);
    ki = -K_total(end);

A_servo = A_aug-B_aug*K_total;
B_servo = [zeros(n,1);1];
C_servo = [C 0];
sys_servo = ss(A_servo, B_servo, C_servo, D);

[num,den]=ss2tf(A_servo,B_servo,C_servo,D);
closed_loop_poles_part_E = roots(den)
disp('negative real poles mean asymptotic stability')
tf(num,den)
DC_gain_for_part_E = num(end)/den(end)
```

```
PART E
K_total =
  13.0000    7.0000   -6.0000

closed_loop_poles_part_E =
  -3.0000
  -2.0000
  -1.0000

negative real poles mean asymptotic stability
ans =
        6
 ----------------------
 s^3 + 6 s^2 + 11 s + 6

Continuous-time transfer function.

DC_gain_for_part_E =
   1.0000
```

f) Considering the perturbed system where $a_{21} = 2.1$, and the same controller as in e), verify that the closed-loop system is still asymptotically stable, and that asymptotic tracking is still achieved despite the perturbation.

```
%% PRT F
disp('PART F')
%purturb system
a21 = 2.1;
A = [0 1; a21 1];

A_aug = [A zeros(n,1); -C 0];
B_aug = [B; 0];

%augment with integral action as a servomechanism
A_servo = A_aug-B_aug*K_total;  %use same K_total as in part E
B_servo = [zeros(n,1);1];
C_servo = [C 0];
sys_servo = ss(A_servo, B_servo, C_servo, D);

[num,den]=ss2tf(A_servo,B_servo,C_servo,D);
closed_loop_poles_part_F = roots(den)
disp('negative real poles mean asymptotic stability')
tf(num,den)
DC_gain_for_part_F = num(end)/den(end)
```

```
PART F
closed_loop_poles_part_F =

  -3.1300
  -1.8122
  -1.0578

negative real poles mean asymptotic stability

ans =

         6
  -----------------------
  s^3 + 6 s^2 + 10.9 s + 6

Continuous-time transfer function.

DC_gain_for_part_F =
   1.0000
```

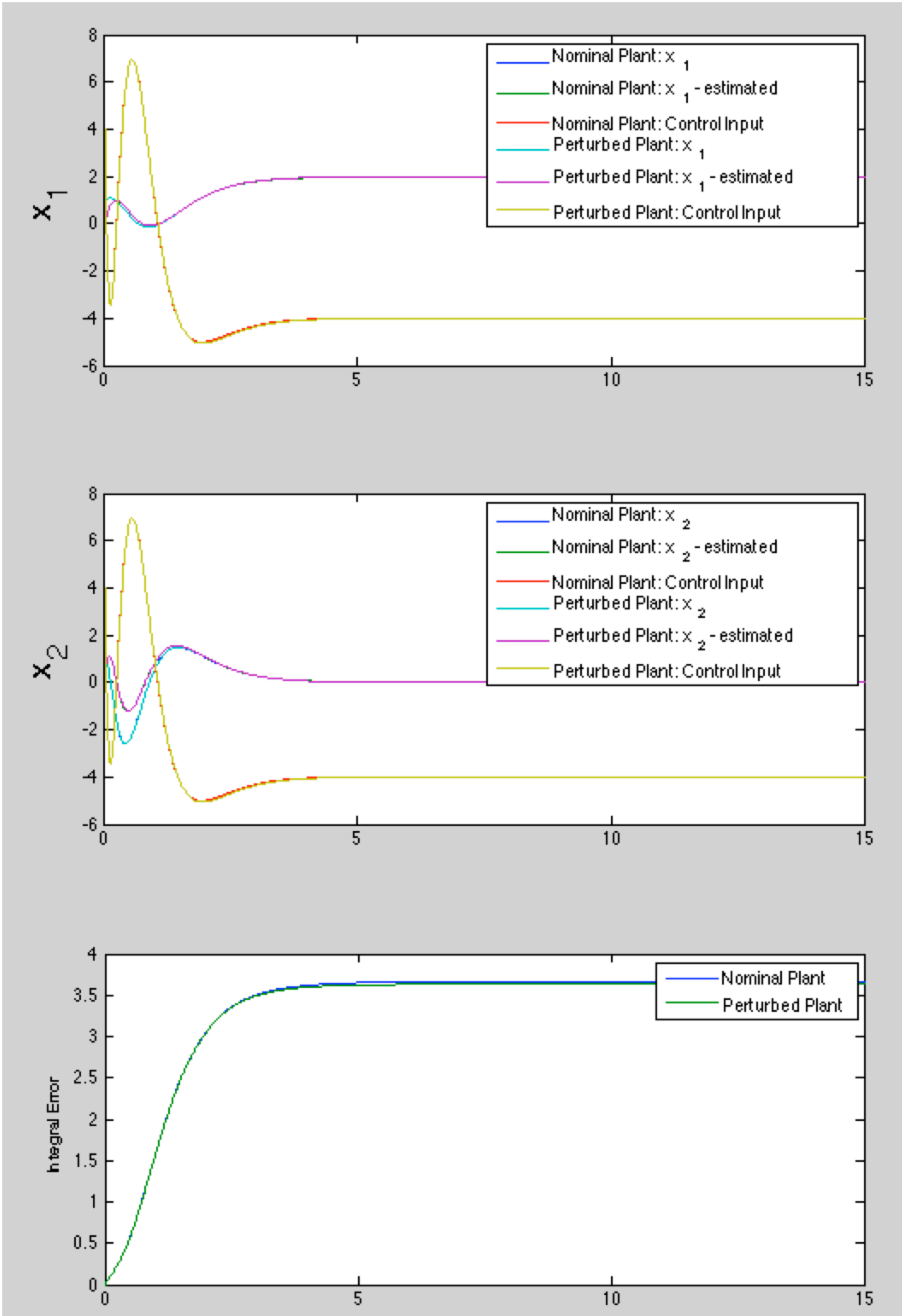
g) Using the same observer as in b), find the DC gain of the closed-loop transfer function. Verify that asymptotic tracking is still achieved and simulate the response with and without the perturbation; specifically, plot the state variables and their estimated values and the control input, *u*. Comment on the results. Is the controller robust?

```
%% PART G
disp('PART G')

%--- PERTURBED

% create big system with servomechanism, observer, and perturbed plant
% use K and L from parts E and B respectively

A_BigSys_pert = [A B*ki -B*k; -C 0 zeros(1,n); L*C B*ki A-B*k-L*C];
B_BigSys = [zeros(n,1); 1; zeros(n,1)];
C_BigSys = [C 0 zeros(1,n)];

sys_pert = ss(A_BigSys_pert, B_BigSys, C_BigSys, 0);

%--- NOMINAL

%recreate big system with servo, observer, and nominal plant
%use same K and L
a21 = 2.0;
A = [0 1; a21 1];

A_BigSys_Nominal = [A B*ki -B*k; -C 0 zeros(1,n); L*C B*ki A-B*k-L*C];
%B,C,D unchanged
sys_nominal = ss(A_BigSys_Nominal, B_BigSys, C_BigSys, 0);

%--- COMPARISON

%DC Gain Analysis
[num,den]=ss2tf(A_BigSys_pert, B_BigSys, C_BigSys, 0)
DC_gain_perturbed = num(end)/den(end)

[num,den]=ss2tf(A_BigSys_Nominal, B_BigSys, C_BigSys, 0)
DC_gain_nominal = num(end)/den(end)

%Simulate
[y_pert, t, x_pert] = lsim(sys_pert, r, t, [x0; 0; x0_est]);
[y_nom, t, x_nom] = lsim(sys_nominal, r, t, [x0; 0; x0_est]);

%control input for each variable:
U_nom = -K*x_nom(:,4:5)' +G*r';      %transposes introduced to make
dimensions agree
U_pert = -K*x_pert(:,4:5)' +G*r';      %transposes introduced to make
dimensions agree
```

```
figure
subplot(311)

plot(t,x_nom(:,1),t,x_nom(:,4),t,U_nom,t,x_pert(:,1),t,x_pert(:,4),t,U_
pert)
    ylabel('x_1','fontsize',20),legend('Nominal Plant: x_1','Nominal
Plant: x_1 - estimated','Nominal Plant: Control Input', 'Perturbed
Plant: x_1','Perturbed Plant: x_1 - estimated','Perturbed Plant:
Control Input')
subplot(312)

plot(t,x_nom(:,2),t,x_nom(:,5),t,U_nom,t,x_pert(:,2),t,x_pert(:,5),t,U_
pert)
    ylabel('x_2','fontsize',20),legend('Nominal Plant: x_2','Nominal
Plant: x_2 - estimated','Nominal Plant: Control Input', 'Perturbed
Plant: x_2','Perturbed Plant: x_2 - estimated','Perturbed Plant:
Control Input')
subplot(313)
    plot(t,x_nom(:,3),t,x_pert(:,3))
    ylabel('Integral Error'),legend('Nominal Plant', 'Perturbed Plant')
```

PART G

```
num =
     0         0         0    6.0000   54.0000  119.4000

den =
    1.0000   15.0000   84.8000  223.5000  270.9100  119.4000


DC_gain_perturbed =
    1.0000

num =
     0         0         0    6.0000   54.0000  120.0000

den =
    1.0000   15.0000   85.0000  225.0000  274.0000  120.0000


DC_gain_nominal =
    1.0000
>>
```



Nominal Plant: x 1
Nominal Plant: x 1 - estimated
Nominal Plant: Control Input
Perturbed Plant: x 1
Perturbed Plant: x 1 - estimated
Perturbed Plant: Control Input
x1
8
6
4
2
0
-2
-4
-6
0
5
10
15
Nominal Plant: x 2
Nominal Plant: x 2 - estimated
Nominal Plant: Control Input
Perturbed Plant: x 2
Perturbed Plant: x 2 - estimated
Perturbed Plant: Control Input
x2
8
6
4
2
0
-2
-4
-6
0
5
10
15
Nominal Plant
Perturbed Plant
Integral Error
4
3.5
3
2.5
2
1.5
1
0.5
0
0
5
10
15

Comments on Part G

There is very little difference between the results for the nominal and perturbed plants. In fact, it is difficult to see because the curves are just about right on top of each other. This indicates that the controller is robust to disturbances in the plant. In all cases, the integral error converges such that there is no steady state error and the state x1 tracks the reference input well.