# EE49001: Control and Electronic System Design

Assignment-5 (Compensatory)

Submitted By:

21EE30004: Anirvan Krishna

## Section-I: Networked Control Systems

### Plotting State Trajectory for Lossless transmission

An NCS plant is characterized by the following linear discrete time system

$$x(t+1) = Ax(t) + Bu(t), x(0) = \begin{bmatrix} 2 & 5 & 8 \end{bmatrix}^T$$
$$u(t) = Kx(t)$$

| | | | |
|---|---|---|---|
| $A = \begin{bmatrix} -2 & -13 & 9 \\ -5 & -10 & 9 \\ -10 & -11 & 12 \end{bmatrix}$ | $B = \begin{bmatrix} 1 \\ 4 \\ 7 \end{bmatrix}$ | $K = \begin{bmatrix} 2.2225 & -10.44 & 5.5944 \end{bmatrix}$ | |

The system is simulated for 15-time units (following script) the system trajectory is plotted.

MATLAB Code

```
clc; clear; close all;

A = [
-2 -13 9;
-5 -10 9
-10 -11 12;
];

B = [1;4;7];
K = [2.2225 -10.44 5.5944];
x0 = [2; 5; 8];
t = 0:15;
xt = zeros(3,16); xt(:,1) = x0;
ut = zeros(1,16); ut(1) = K*x0;

for i = 2:16
xt(:,i) = A*xt(:,i-1) + B*ut(i-1);
ut(i) = K*xt(:,i);
end
```

```
xt = gen_xt(15,A,B,K,x0,0);

fig = figure; fig.Position(3) = 1000; fig.Position(4) = 1000; movegui('center');
sgtitle('State Trajectory', Interpreter='latex');

hold on;
plot(t,xt(1,:));
plot(t,xt(2,:));
plot(t,xt(3,:));
hold off;
xlabel('$t$', Interpreter='latex');
ylabel('$x(t)$', Interpreter='latex');
legend('$x_1$', '$x_2$', '$x_3$',Interpreter='latex');
```

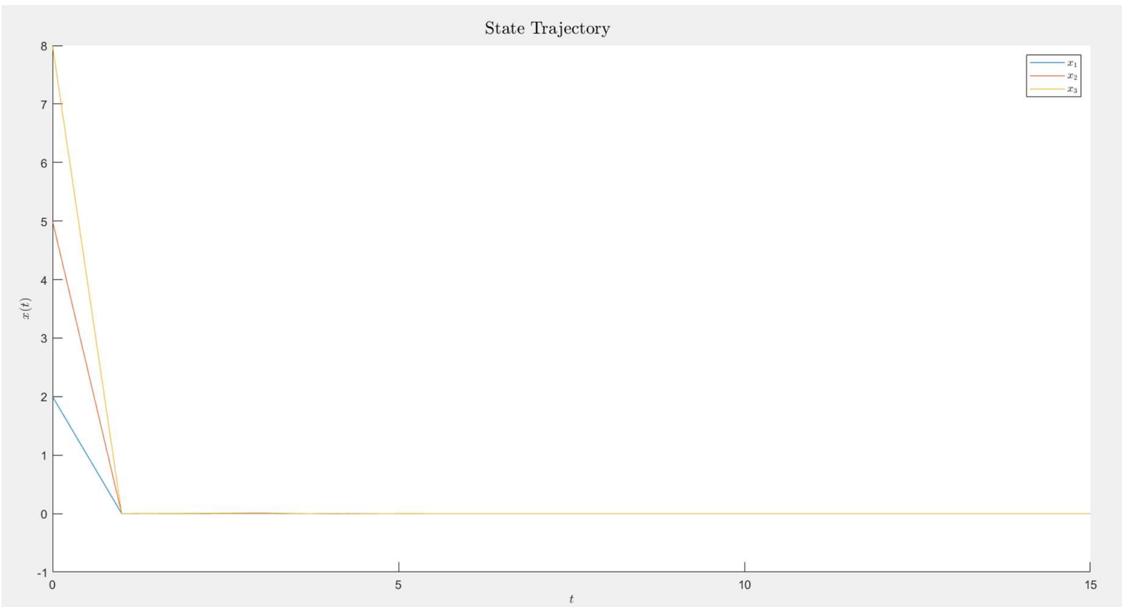



Fig. Trajectory of States $x_1, x_2$ and $x_3$

## Lossy Transmission (Fixed Loss $\kappa$)

Let the loss signal $\kappa(t)$ be given as:

$$\kappa(t) = \{0,0,1,0,0,0,1,0,1,0,0,1,0,0,0,0\}$$

After the losses, the input: $u(t) \coloneqq \kappa(t) \cdot u(t)$

MATLAB Code

```
clc; clear; close all;

A = [
    -2 -13 9;
    -5 -10 9
    -10 -11 12;
    ];

B = [1;4;7;];
```

```
K = [2.2225 -10.44 5.5944];
k = [0 0 1 0 0 0 1 0 1 0 0 1 0 0 0 0];
x0 = [2; 5; 8]; t = 0:15;
xt = zeros(3,16); xt(:,1) = x0;
ut = zeros(1,16); ut(1) = K*x0*k(1);

for i = 2:16
    xt(:,i) = A*xt(:,i-1) + B*ut(i-1);
    ut(i) = K*xt(:,i)*k(i);
end

fig = figure; fig.Position(3) = 1000; fig.Position(4) = 500; movegui('center');
sgtitle('State Trajectory', Interpreter='latex');

hold on;
plot(t,xt(1,:));
plot(t,xt(2,:));
plot(t,xt(3,:));
hold off;
xlabel('$t$', Interpreter='latex');
ylabel('$x(t)$', Interpreter='latex');
legend('$x_1$', '$x_2$', '$x_3$',Interpreter='latex');
```

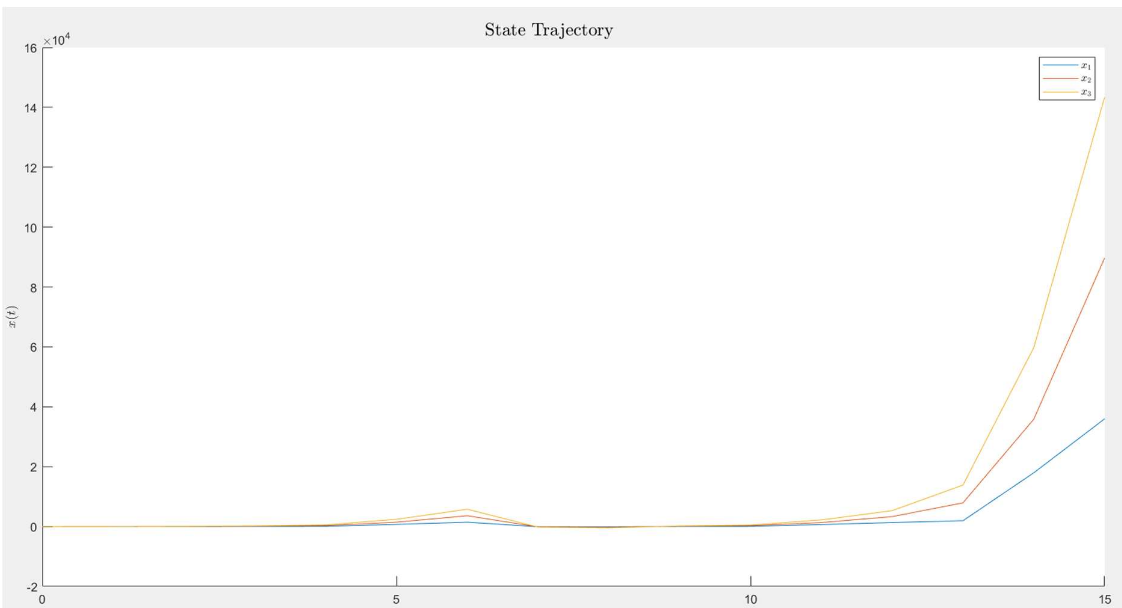



Fig. State trajectories with given values of $\kappa(t)$

In the lossy medium, we observe that the state trajectories start from $0$ and then start increasing. Meanwhile in the lossless medium, the states are observed to decrease from some initial value to $0$, As the time increased

## Lossy Medium (Random Loss)

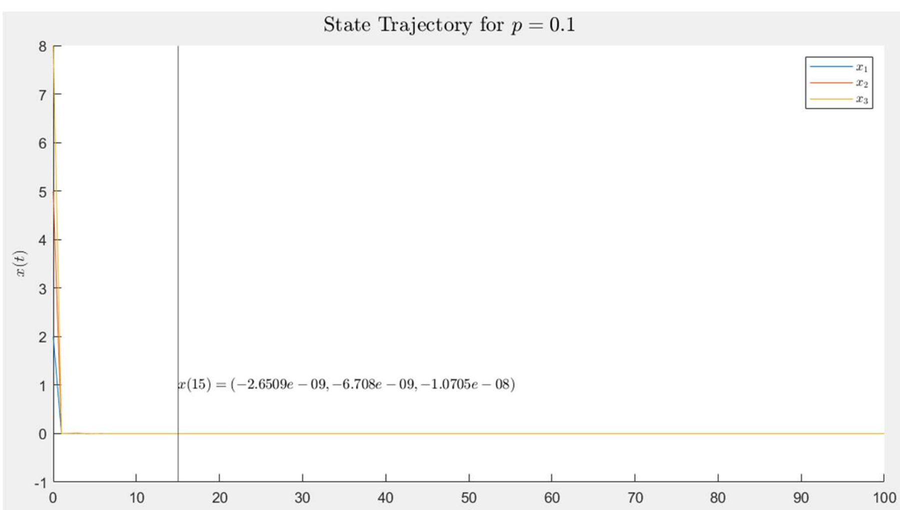
The same NCS plant is simulated for 100 units time (script following), this time the data loss signal is generated randomly.

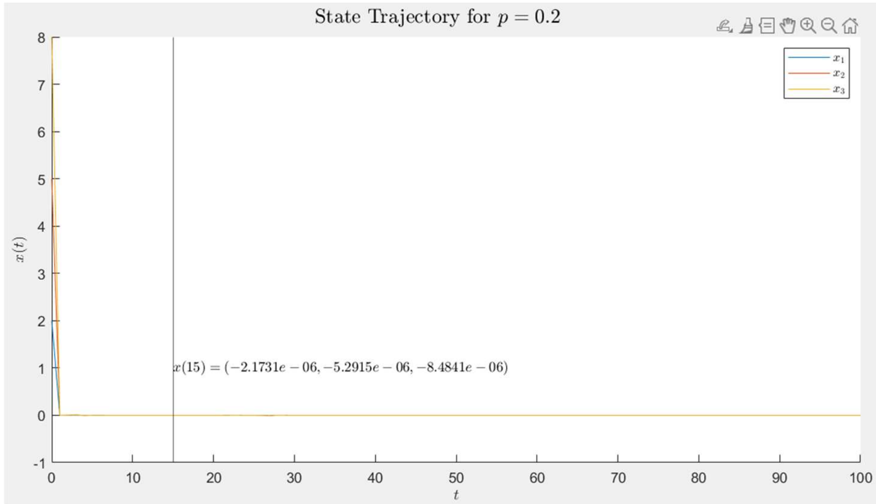
$$\kappa(t) = \begin{cases} 0, & \text{with probability } p \\ 1, & \text{with probability } 1-p \end{cases}$$

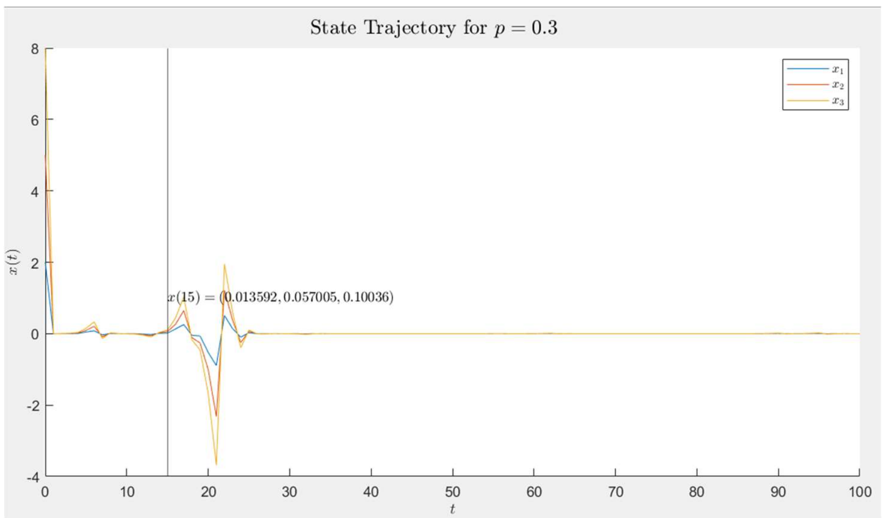
## MATLAB Code

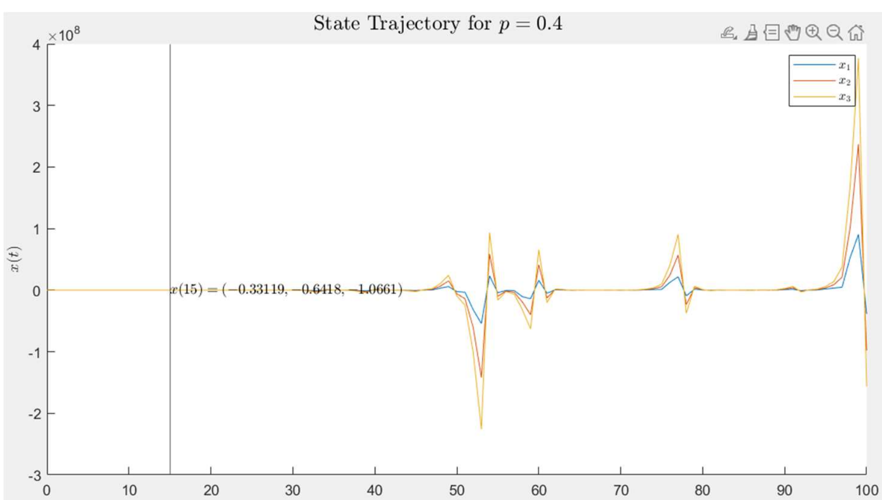
```
clc; clear; close all;

A = [
-2 -13 9;
-5 -10 9
-10 -11 12;
];

B = [1;4;7;];
K = [2.2225 -10.44 5.5944];

for p = 1:9

xt = gen_xt(100,A,B,K,x0,p/10); t = 0:100;
fig = figure; fig.Position(3) = 1000; fig.Position(4) = 500; movegui('center');
sgtitle(['State Trajectory for $p=',num2str(p/10),'$'], Interpreter='latex');

hold on;
plot(t,xt(1,:));
plot(t,xt(2,:));
plot(t,xt(3,:));
xline(15);
text(15,1,['$x(15)=(',num2str(xt(1,16)),',',num2str(xt(2,16)),',',num2str(xt(3,16)),')$'], Interpreter='latex')

hold off;
xlabel('$t$', Interpreter='latex');
ylabel('$x(t)$', Interpreter='latex');
legend('$x_1$', '$x_2$', '$x_3$',Interpreter='latex');

end
```

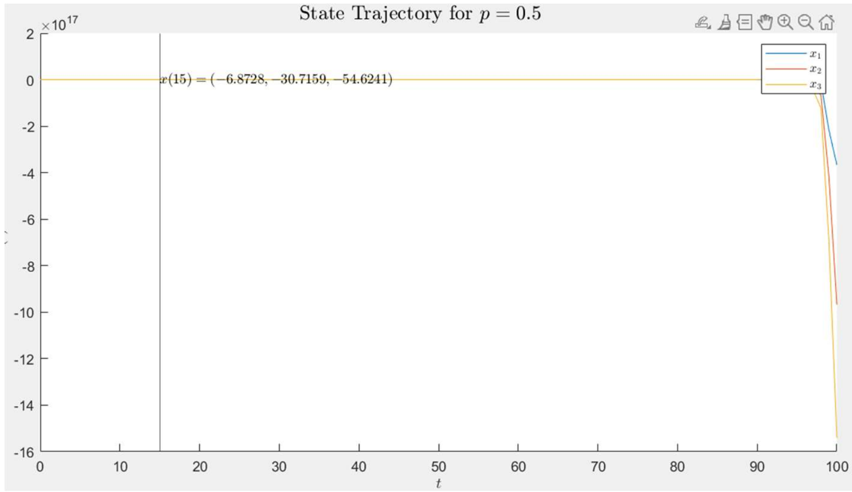
```
function xt = gen_xt(n, A, B, K, x0, p)

  xt = zeros(length(x0),n+1); ut = zeros(height(K),n+1);
  k = (rand(1,n+1)>p);
  xt(:,1) = x0; ut(:,1) = K*x0*k(1);

  for i = 2:n+1

    xt(:,i) = A*xt(:,i-1) + B*ut(:,i-1);
    ut(:,i) = K*xt(:,i)*k(i);

  end
end
```

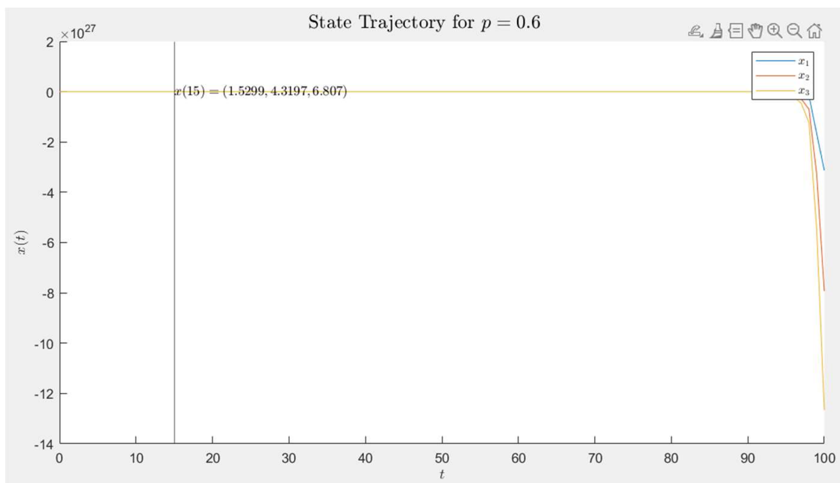


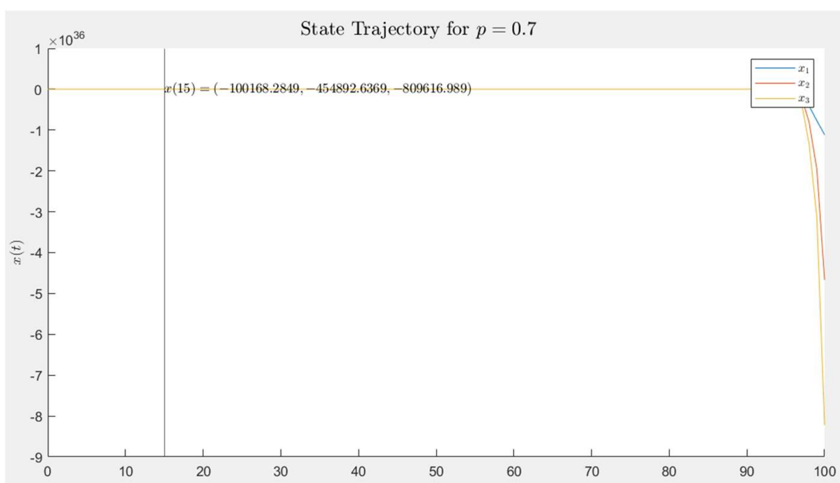

State Trajectory for $p = 0.2$
$x(t)$
$t$
$x_1$
$x_2$
$x_3$
$x(15) = (-2.1731e-06, -5.2915e-06, -8.4841e-06)$

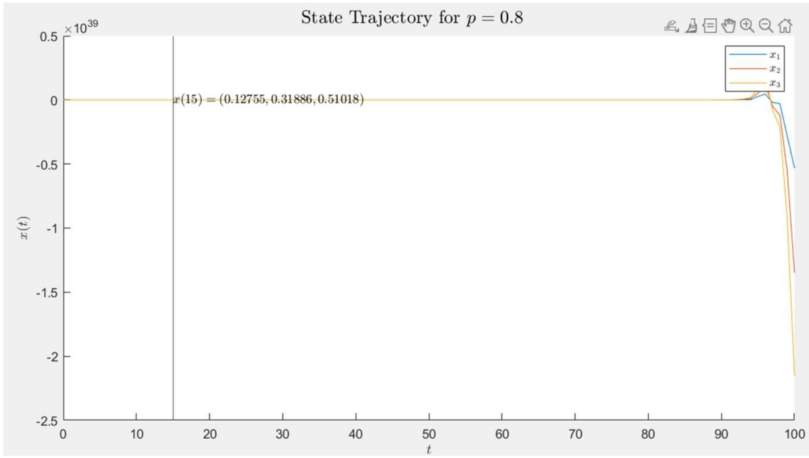

State Trajectory for $p = 0.3$
$x(t)$
$t$
$x_1$
$x_2$
$x_3$
$x(15) = (0.013592, 0.057005, 0.10036)$


State Trajectory for $p = 0.4$
$\times 10^8$
$x(t)$
$t$
$x_1$
$x_2$
$x_3$
$x(15) = (-0.33119, -0.6418, -1.0661)$


State Trajectory for $p = 0.5$
$\times 10^{17}$
$x(15) = (-6.8728, -30.7159, -54.6241)$
$x_1$
$x_2$
$x_3$
$t$


State Trajectory for $p = 0.6$
$\times 10^{27}$
$x(15) = (1.5299, 4.3197, 6.807)$
$x_1$
$x_2$
$x_3$
$x(t)$
$t$


State Trajectory for $p = 0.7$
$\times 10^{36}$
$x(15) = (-100168.2849, -454892.6369, -809616.989)$
$x_1$
$x_2$
$x_3$
$x(t)$
$t$



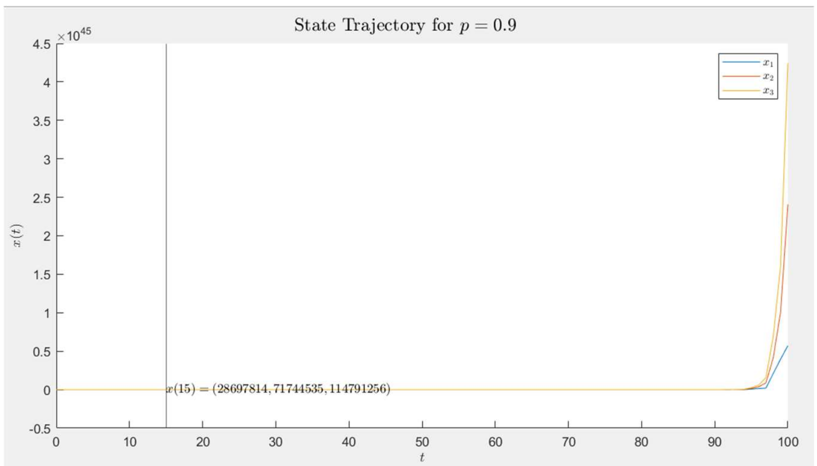




**Fig.** Lossy Media with different probabilities for losses

As can be observed $x(15) \approx 0$ for $p \leq 0.3$.

# Section-II: Switched Systems

## Trajectory of the Switched Systems

A switched system is characterized by

$$x(t+1) = A_{\sigma(t)}x(t), x(0) = [-1 \quad 1]^T$$

Where,

| | |
|---|---|
| $A_1 = \begin{bmatrix} 0.47 & 0.12 \\ -3.90 & 0.19 \end{bmatrix}$ | $A_2 = \begin{bmatrix} -0.03 & 0.78 \\ 0.60 & 0.47 \end{bmatrix}$ |

We have: $\sigma_1(t) = \{1, 1, 2, 2, 2, 2, 1, 1, 2, 2, 2, 2, 1, 1, 2\}$

$\sigma_2(t) = \{1, 1, 1, 2, 2, 1, 1, 1, 2, 2, 1, 1, 1, 2, 2\}$

The system is simulated for 15 time units (following script) for two different random noise signal and the response is plotted.

MATLAB Code

```
clear; close all; clc;

A1 = [
0.47, 0.12;
-3.90, 0.19
];

A2 = [
-0.03, 0.78;
0.60, 0.47;
];

A = zeros(2,2,2);
A(:,:,1) = A1;
A(:,:,2) = A2;

x0 = [-1;1];

s1 = [1,1,2,2,2,2,1,1,2,2,2,2,1,1,2];
s2 = [1,1,1,2,2,1,1,1,2,2,1,1,1,2,2];

n=15;
xt = zeros(length(x0), n+1);
xt(:,1) = x0;

for i = 1:n
xt(:,i+1) = A(:,:,s1(i))*xt(:,i);
end

t = 0:15;
xt1 = gen_xt1(15,A,s1,x0);
xt2 = gen_xt1(15,A,s2,x0);
```

```
fig = figure; fig.Position(3) = 2000; fig.Position(4) = 1000; movegui('center');
sgtitle('State Trajectory', Interpreter='latex');

subplot(1,2,1);
hold on;
plot(t,xt1(1,:));
plot(t,xt1(2,:));
hold off;
xlabel('$t$', Interpreter='latex');
ylabel('$x^{(1)}(t)$', Interpreter='latex');
legend('$x_1$', '$x_2$',Interpreter='latex');

subplot(1,2,2);
hold on;
plot(t,xt2(1,:));
plot(t,xt2(2,:));
hold off;
xlabel('$t$', Interpreter='latex');
ylabel('$x^{(2)}(t)$', Interpreter='latex');
legend('$x_1$', '$x_2$',Interpreter='latex');

```

```
function xt = gen_xt1(n,A,s,x0)
xt = zeros(length(x0), n+1);
xt(:,1) = x0;
for i = 1:n
    a.  xt(:,i+1) = A(:,:,s(i))*xt(:,i);
end
end
```

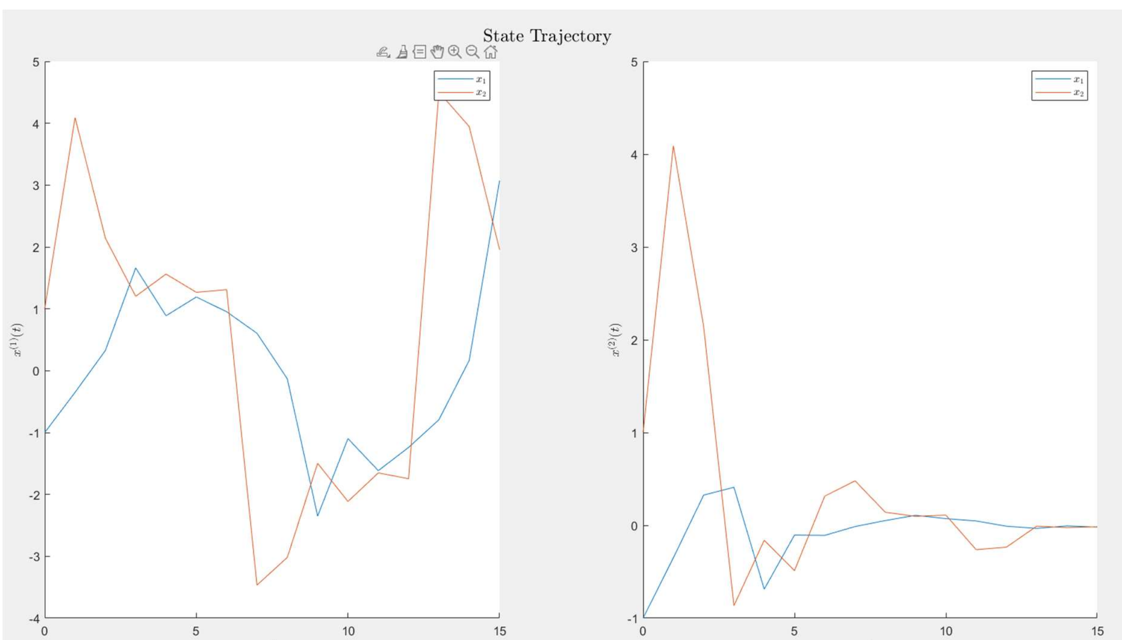



Fig. State trajectories corresponding to different values of $\sigma(t)$

As can be seen that the second system is stable. Thus, random switching doesn't always result in a stable response.