**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

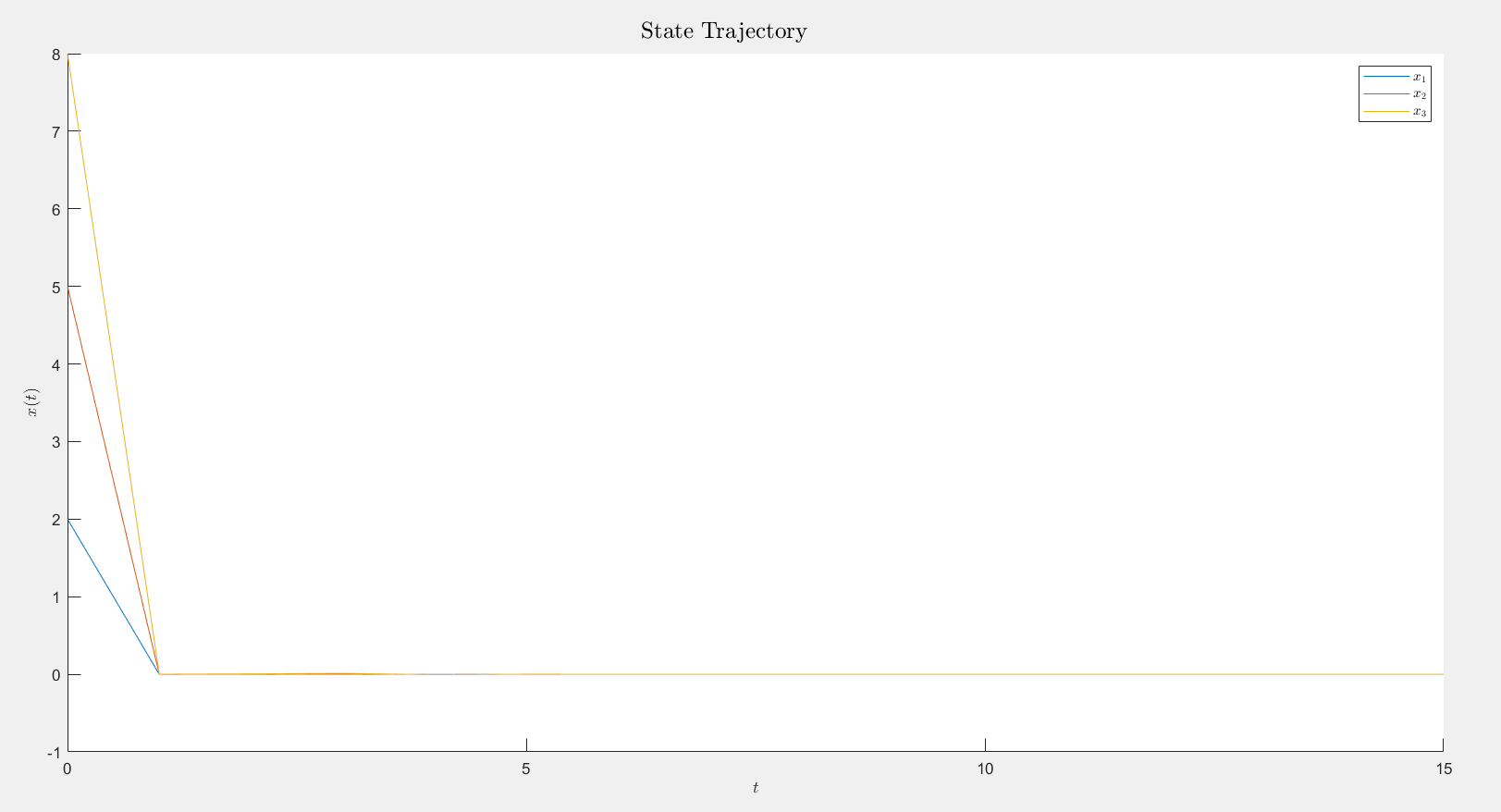
An NCS plant is characterized by the following linear discrete time system

|  |  |  |
| --- | --- | --- |
|  |  |  |

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

**MATLAB Code**

|  |
| --- |
| 1. clc; clear; close all; 2. A = [ 3. -2 -13 9; 4. -5 -10 9 5. -10 -11 12; 6. ]; 7. B = [1;4;7]; 8. K = [2.2225 -10.44 5.5944]; 9. x0 = [2; 5; 8]; 10. t = 0:15; 11. xt = zeros(3,16); xt(:,1) = x0; 12. ut = zeros(1,16); ut(1) = K\*x0; 13. for i = 2:16 14. xt(:,i) = A\*xt(:,i-1) + B\*ut(i-1); 15. ut(i) = K\*xt(:,i); 16. end 17. xt = gen\_xt(15,A,B,K,x0,0); 18. fig = figure; fig.Position(3) = 1000; fig.Position(4) = 1000; movegui('center'); 19. sgtitle('State Trajectory', Interpreter='latex'); 20. hold on; 21. plot(t,xt(1,:)); 22. plot(t,xt(2,:)); 23. plot(t,xt(3,:)); 24. hold off; 25. xlabel('$t$', Interpreter='latex'); 26. ylabel('$x(t)$', Interpreter='latex'); 27. legend('$x\_1$', '$x\_2$', '$x\_3$',Interpreter='latex'); |

****

**Fig.** Trajectory of States and

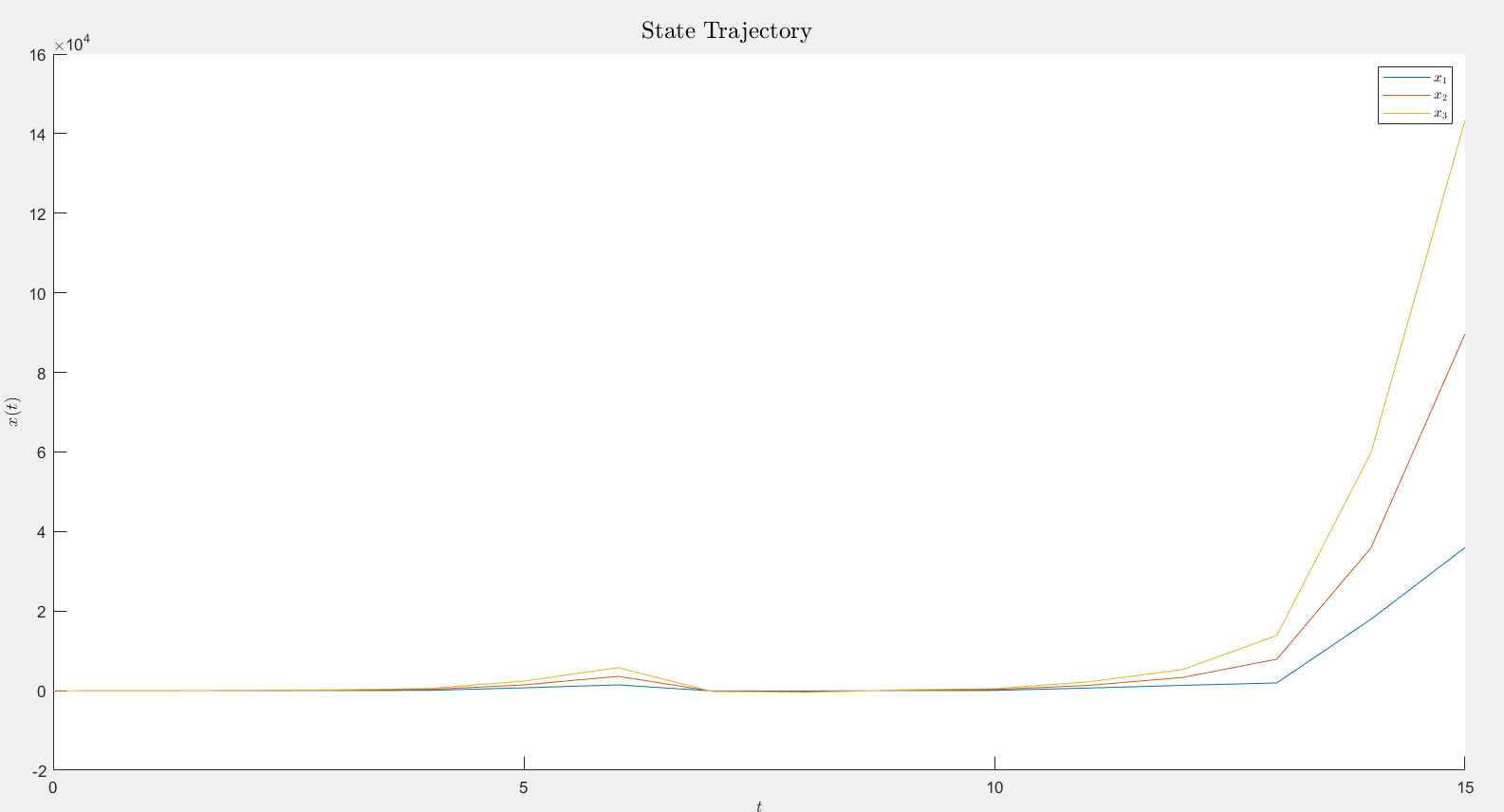
# Lossy Transmission (Fixed Loss )

Let the loss signal be given as:

After the losses, the input:

**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

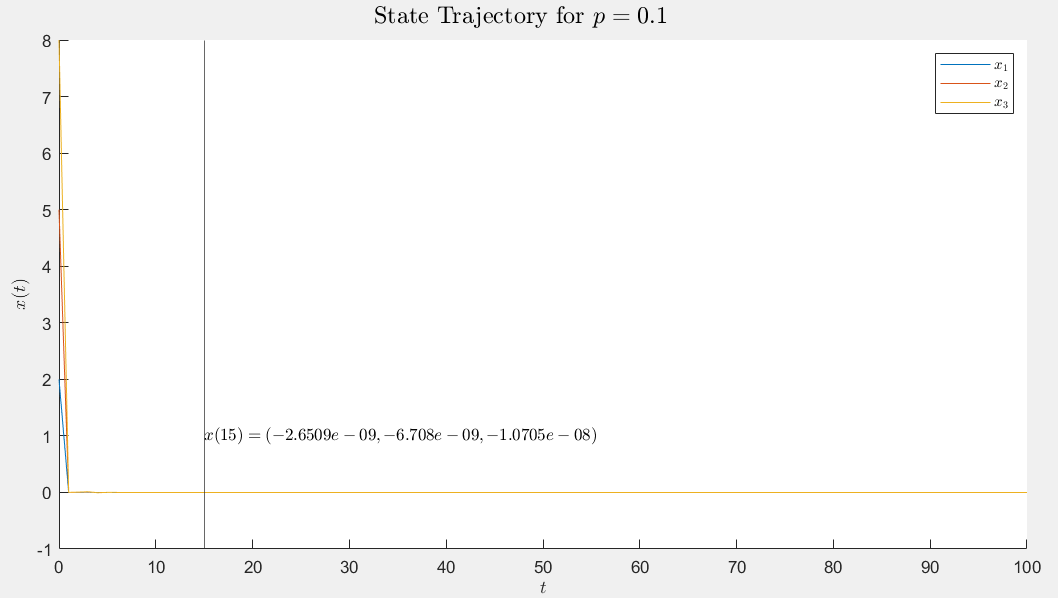
In the lossy medium, we observe that the state trajectories start from and then start increasing. Meanwhile in the lossless medium, the states are observed to decrease from some initial value to As the time increased

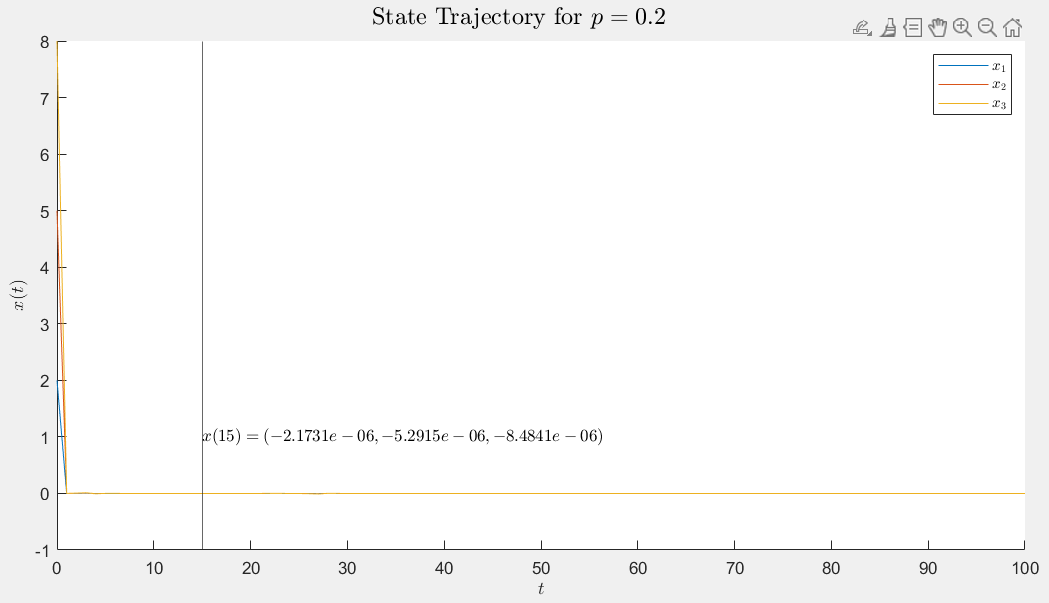
# Lossy Medium (Random Loss)

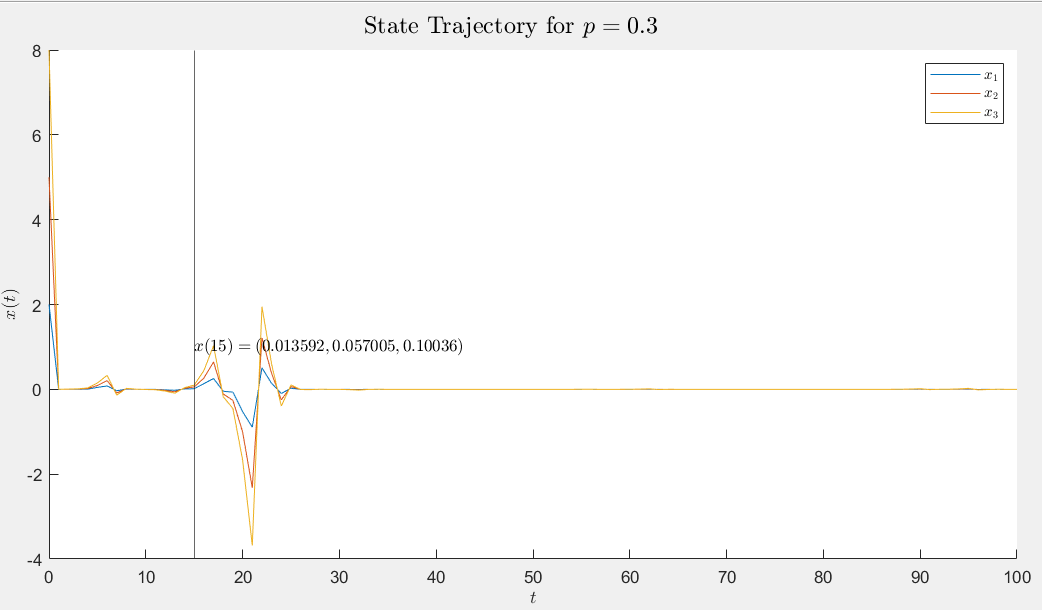
The same NCS plant is simulated for 100 units time (script following), this time the data loss signal is generated randomly.

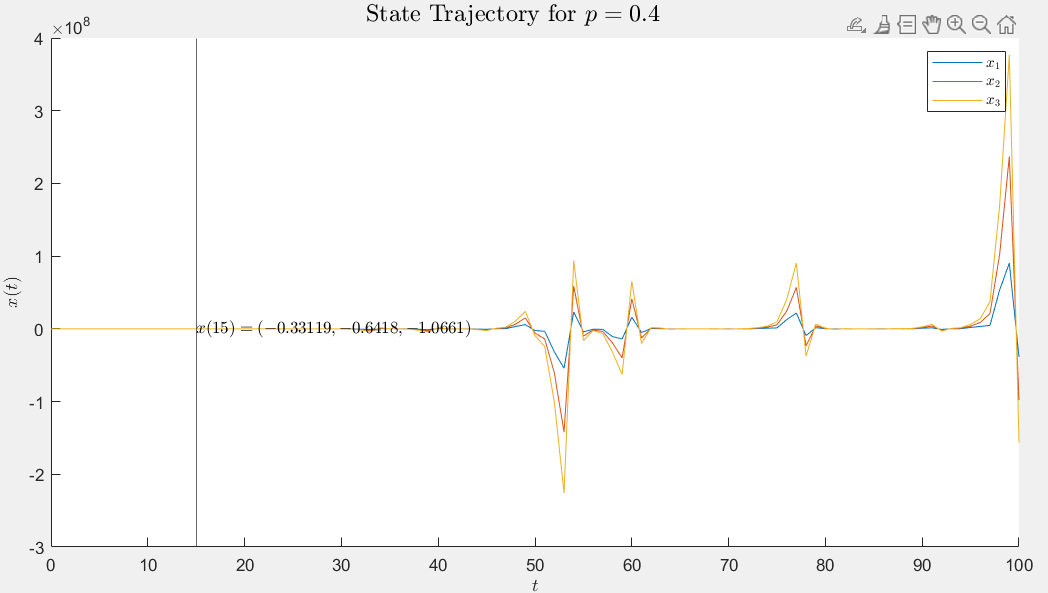
**MATLAB Code**

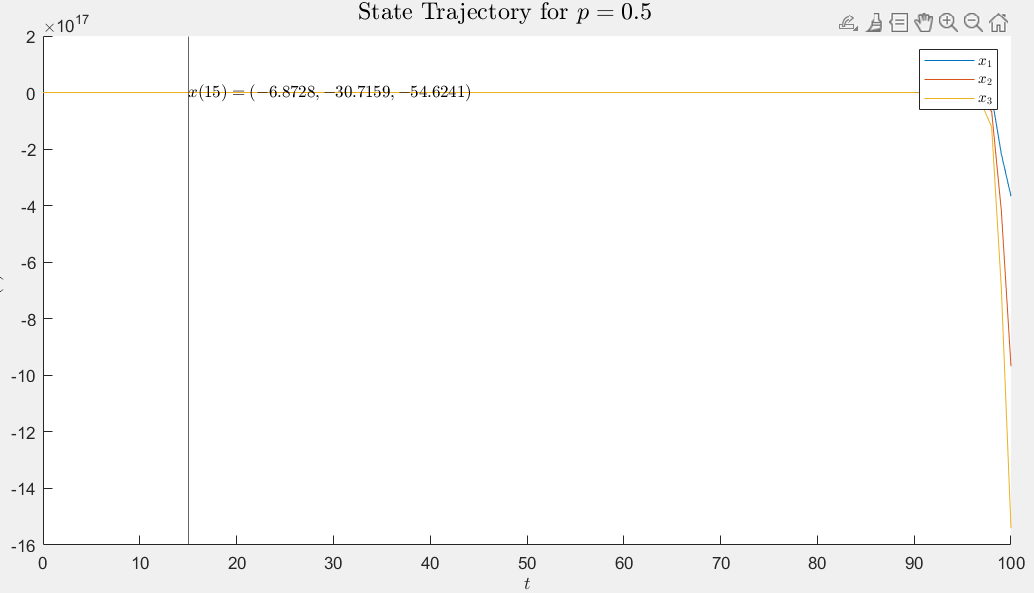
|  |
| --- |
| 1. clc; clear; close all; 2. A = [ 3. -2 -13 9; 4. -5 -10 9 5. -10 -11 12; 6. ]; 7. B = [1;4;7;]; 8. K = [2.2225 -10.44 5.5944]; 9. for p = 1:9 10. xt = gen\_xt(100,A,B,K,x0,p/10); t = 0:100; 11. fig = figure; fig.Position(3) = 1000; fig.Position(4) = 500; movegui('center'); 12. sgtitle(['State Trajectory for $p=',num2str(p/10),'$'], Interpreter='latex'); 13. hold on; 14. plot(t,xt(1,:)); 15. plot(t,xt(2,:)); 16. plot(t,xt(3,:)); 17. xline(15); text(15,1,['$x(15)=(',num2str(xt(1,16)),',',num2str(xt(2,16)),',',num2str(xt(3,16)),')$'], Interpreter='latex') 18. hold off; 19. xlabel('$t$', Interpreter='latex'); 20. ylabel('$x(t)$', Interpreter='latex'); 21. legend('$x\_1$', '$x\_2$', '$x\_3$',Interpreter='latex'); 22. end |
| 1. function xt = gen\_xt(n, A, B, K, x0, p) 2. xt = zeros(length(x0),n+1); ut = zeros(height(K),n+1); 3. k = (rand(1,n+1)>p); 4. xt(:,1) = x0; ut(:,1) = K\*x0\*k(1); 5. for i = 2:n+1 6. xt(:,i) = A\*xt(:,i-1) + B\*ut(:,i-1); 7. ut(:,i) = K\*xt(:,i)\*k(i); 8. end 9. end |

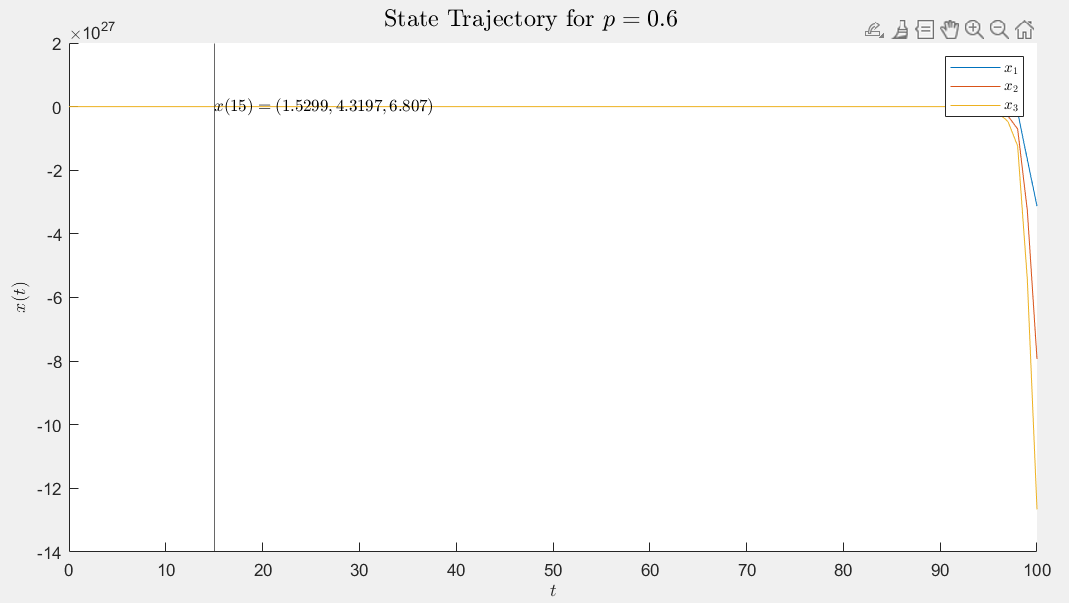


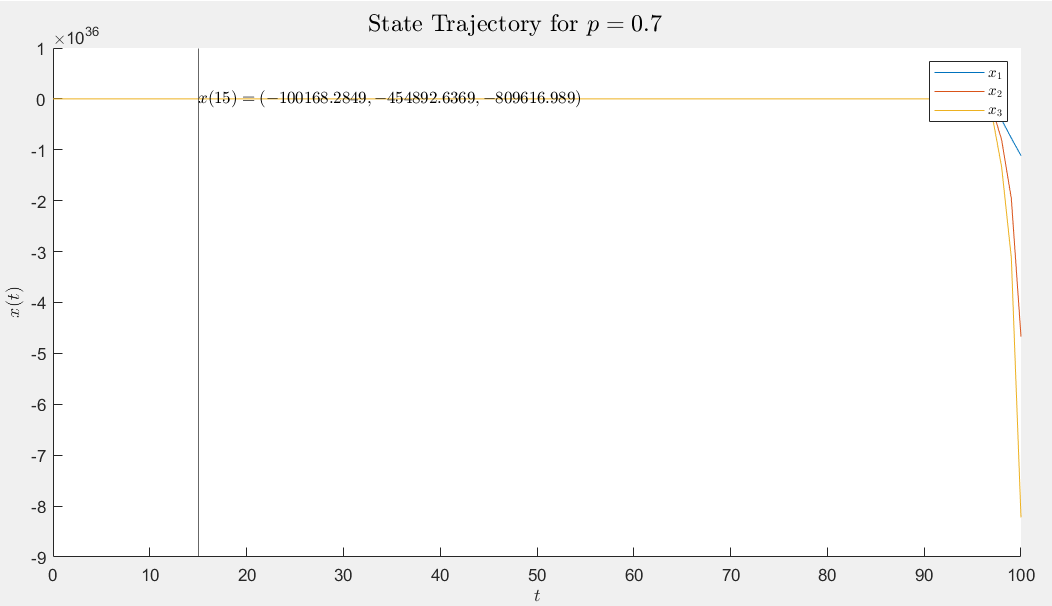


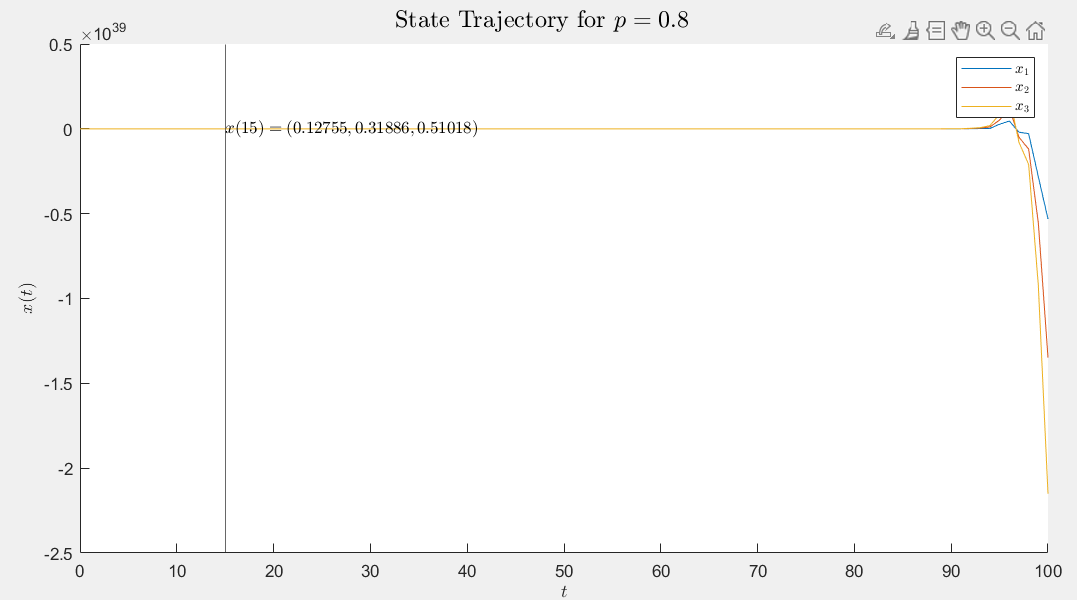


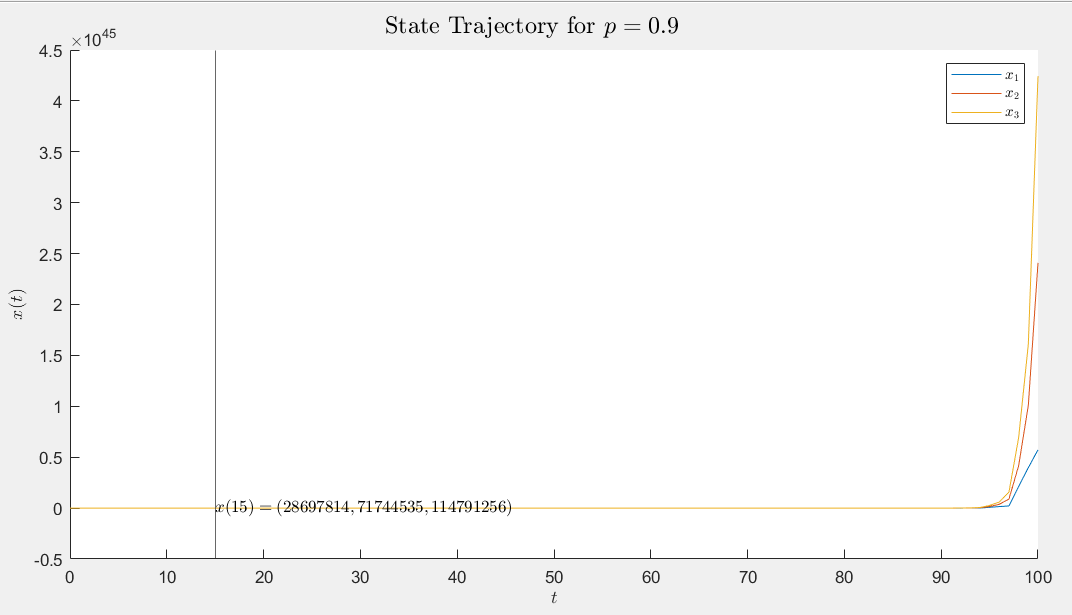












**Fig.** Lossy Media with different probabilities for losses

As can be observed for .

**Section-II: Switched Systems**

# Trajectory of the Switched Systems

A switched system is characterized by

Where,

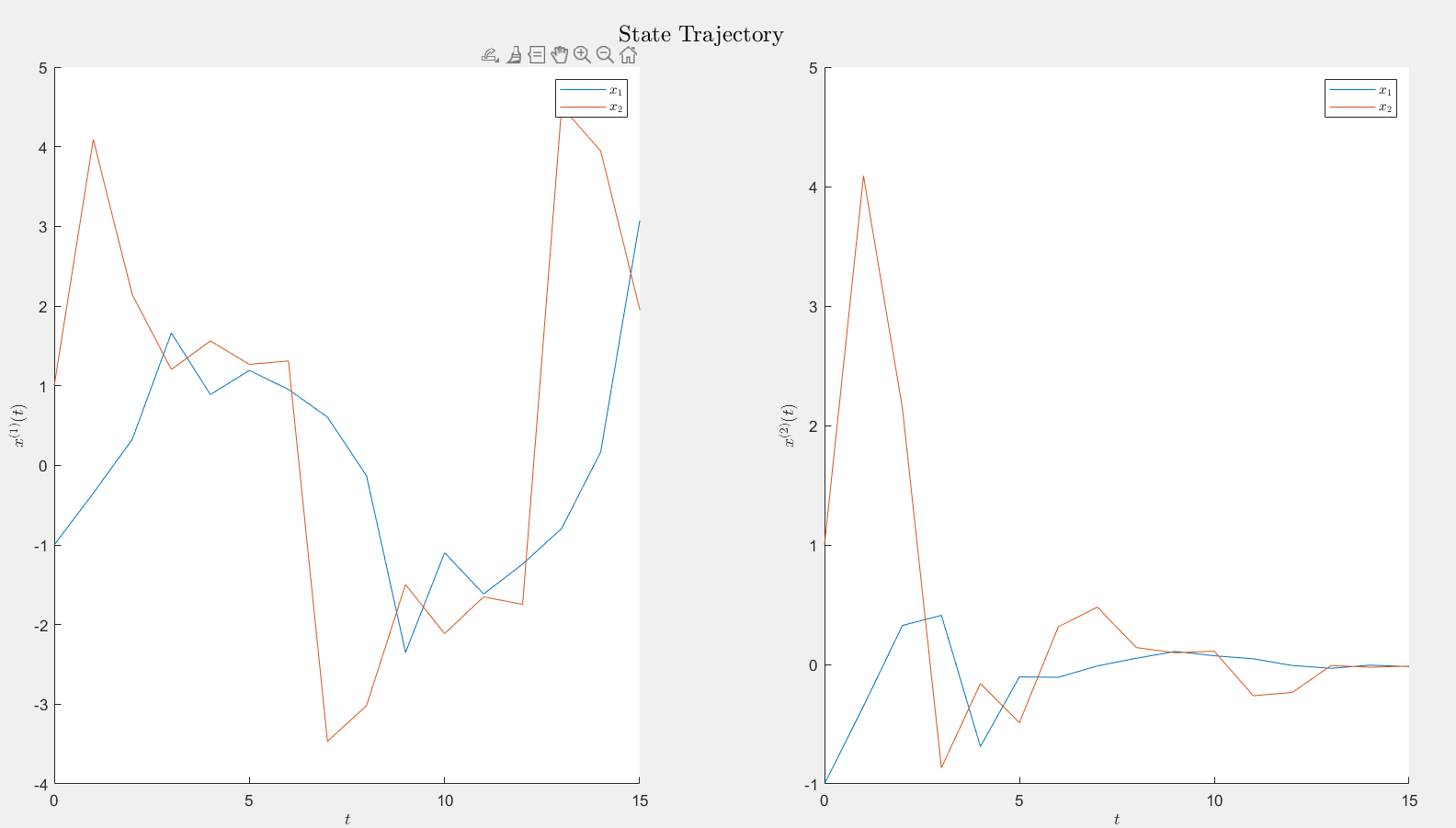
|  |  |
| --- | --- |
|  |  |

We have:

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

**MATLAB Code**

|  |
| --- |
| 1. clear; close all; clc; 2. A1 = [ 3. 0.47, 0.12; 4. -3.90, 0.19 5. ]; 6. A2 = [ 7. -0.03, 0.78; 8. 0.60, 0.47; 9. ]; 10. A = zeros(2,2,2); 11. A(:,:,1) = A1; 12. A(:,:,2) = A2; 13. x0 = [-1;1]; 14. s1 = [1,1,2,2,2,2,1,1,2,2,2,2,1,1,2]; 15. s2 = [1,1,1,2,2,1,1,1,2,2,1,1,1,2,2]; 16. n=15; 17. xt = zeros(length(x0), n+1); 18. xt(:,1) = x0; 19. for i = 1:n 20. xt(:,i+1) = A(:,:,s1(i))\*xt(:,i); 21. end 22. t = 0:15; 23. xt1 = gen\_xt1(15,A,s1,x0); 24. xt2 = gen\_xt1(15,A,s2,x0); 25. fig = figure; fig.Position(3) = 2000; fig.Position(4) = 1000; movegui('center'); 26. sgtitle('State Trajectory', Interpreter='latex'); 27. subplot(1,2,1); 28. hold on; 29. plot(t,xt1(1,:)); 30. plot(t,xt1(2,:)); 31. hold off; 32. xlabel('$t$', Interpreter='latex'); 33. ylabel('$x^{(1)}(t)$', Interpreter='latex'); 34. legend('$x\_1$', '$x\_2$',Interpreter='latex'); 35. subplot(1,2,2); 36. hold on; 37. plot(t,xt2(1,:)); 38. plot(t,xt2(2,:)); 39. hold off; 40. xlabel('$t$', Interpreter='latex'); 41. ylabel('$x^{(2)}(t)$', Interpreter='latex'); 42. legend('$x\_1$', '$x\_2$',Interpreter='latex'); |
| 1. function xt = gen\_xt1(n,A,s,x0) 2. xt = zeros(length(x0), n+1); 3. xt(:,1) = x0; 4. for i = 1:n    1. xt(:,i+1) = A(:,:,s(i))\*xt(:,i); 5. end 6. end |



**Fig.** State trajectories corresponding to different values of

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