ELL702: NON-LINEAR SYSTEMS

Assignment-1

Ashwathram S
2023MEZ8467

The vector field for the system for both cases of u value is given in Figure 1. We can see that the system is diverging and the node is unstable for both the cases. This can also be seen from the eigen values as the eigen values are positive for both values of u.
From Figure 2, we can see that the system moves towards the origin when in first and the third quadrant when $u = -1$ and $u = 1$ respectively.
Thus it is necessary for the initial conditions of state variables to lie in first or the third quadrant. The rectangles specified in figures 3 and 4 enclose the area of suitable inital conditions for the state variables.

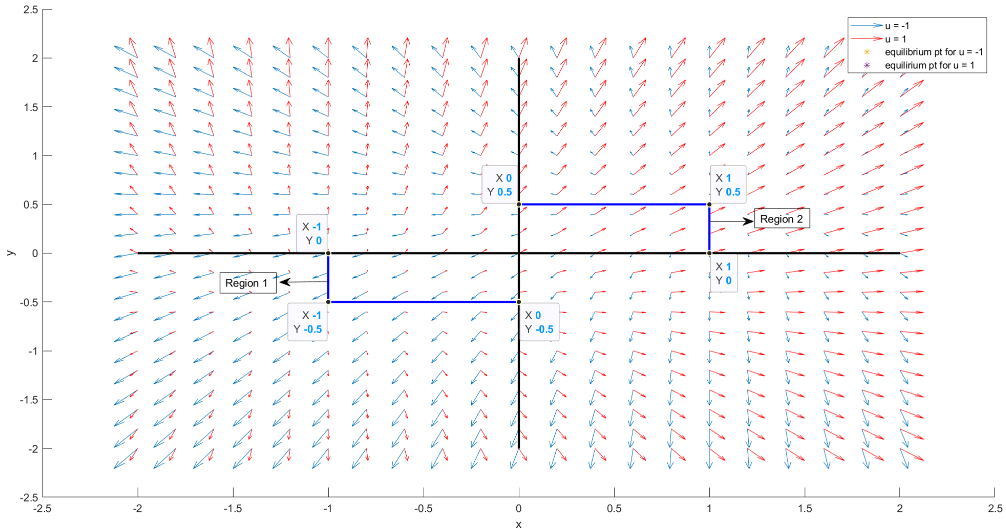

Figure 1: Vector field for the system

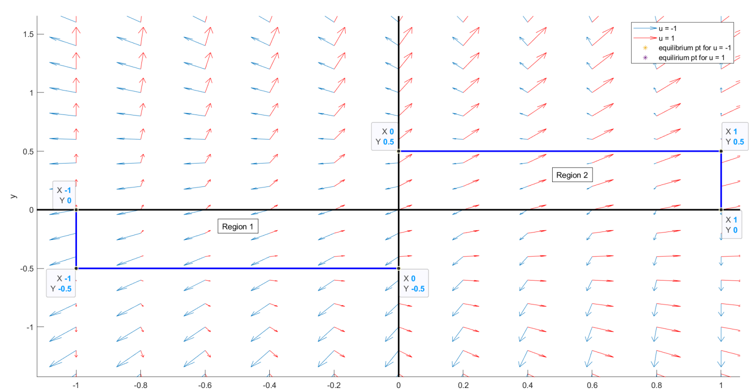

Figure 2: Direction of vector field near origin

1 Switching Strategy 1

For both the switching strategies, we will keep the initial state to be the same. Assuming the states start from the first quadrant close to 0, the initial u value is chosen to be -1 since the vector fields point towards the origin for the system when u=-1 in the first quadrant, if we consider the initial state to start from the 3rd quadrant, then we take u to be 1 since

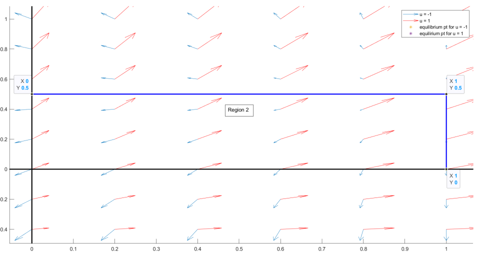

Figure 3: First Quadrant

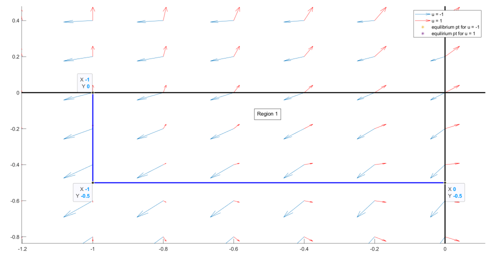

Figure 4: Third Quadrant

the system's vector field is moving towards the origin at 3rd quadrant when u=1. This can be seen clearly in the Figures 3 and 4

The strategy implemented here for when the state goes to either the 2nd or the 4th quadrant is, when the state goes to the second quadrant, u is taken as -1 and in the fourth quadrant it is taken as +1.

The reason behind the strategy is the difference in the slopes of x and y coordinates, since state y changes twice as quick as state x, in the second quadrant, close to the origin, the vector fields due to u=-1 as shown by blue arrows in figure 2 tends to push the state to third quadrant this recovering its position and stabilizing towards origin. Similarly, in the fourth quadrant, u is taken as 1 as the vector fields corresponding to that u push the state to the first quadrant, and stabilizes it at origin as seen from the red arrows in 2.

The time domain data for x and y are plotted and the initial condition are taken as $x = 0.01$ and $y = 0.01$. The simulation is done for 5 secs.

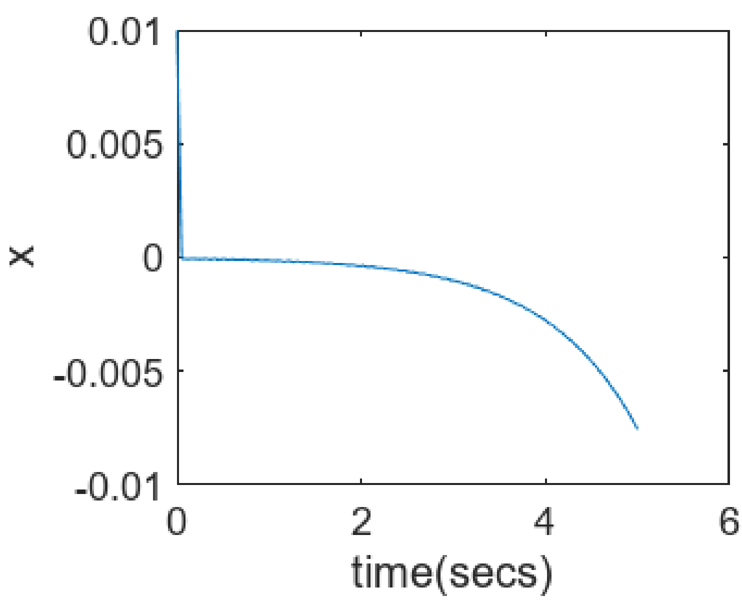

Figure 5: Change of x wrt time: Strategy-1

Through this strategy, we can see that for the given initial conditions, y stabilizes to the origin whereas x starts diverging after reaching the origin.

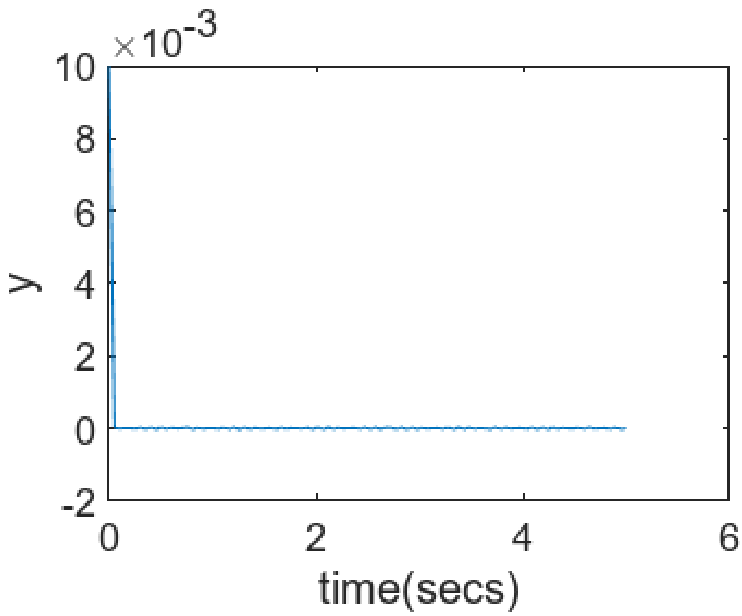

Figure 6: Change of y wrt time: Strategy-1

2 Switching Strategy 2

The value of control variable is kept similar to the previous strategy when the state is in the first or the third quadrant. Thus $u = -1$ in first quadrant and $u = 1$ in third quadrant. The difference here is, when the state enters the second or the fourth quadrant, instead of changing the u value permanently, it is switched between 1 and -1 at a particular frequency, In this case the frequency chosen is 20Hz. This rapid switching is done at the second and third quadrants to reroute the state back to either the first or the third quadrant. Simulation is done for 5 secs keeping the same initial conditions as before. $x = 0.01$ and $y = 0.01$.

Through this strategy, we can see that for the given initial conditions, the state variables change in contrast to the previous strategy, i.e. x stablizies to 0 whereas y diverges from origin.

3 Appendix: Matlab Code

```
global u;
u = -1;
[X,Y] = meshgrid(-2:0.2:2,-2:0.2:2);
U = X + u;
V = (2*Y) + u;

xL = [-2,2];
yL = [-2,2];


U2 = X + (-u);
```

```
V2 = (2*Y) + (-u);

figure
hold on
quiver(X,Y,U,V)
quiver(X,Y,U2,V2,'r')
plot(1,0.5,'*')
plot(-1,-0.5,'*')
line([0 0], yL,'Color','black','LineWidth',2);  %x-axis
line(xL, [0 0],'Color','black','LineWidth',2);  %y-axis
line([-1 -1],[0 -0.5],'Color','blue','LineWidth',2);
line([-1 0],[-0.5 -0.5],'Color','blue','LineWidth',2);
line([0 1],[0.5 0.5],'Color','blue','LineWidth',2);
line([1 1],[0 0.5],'Color','blue','LineWidth',2);
legend('u = -1','u = 1','equilibrium pt for u = -1','equilirium pt for u = 1','','',
hold off

tstart = 0;
tend = 5;
dt = 0.005;
n = 100;

tspan = transpose(linspace(tstart,tend,n));
% X_final = zeros(1,2);
% tspan2 = zeros(1,1);

xinit = [0.01 0.01];

[t,x] = ode45(@(t,x)integratingfunc(t,x), tspan, xinit);
plot(t,x(:,1))
plot(t,x(:,2))

function dxdt = integratingfunc(t,x)
%INTEGRATINGFUNC_SPRINGMASS_SYSTEM Summary of this function goes here
%   Detailed explanation goes here

global u
x1 = x(1);
y = x(2);
u;
if x1>0 && y>0
    u = -1;
elseif x1>0 && y<0
    u = -u;
    %u = 1;
```

```
elseif x1<0 && y<0
    u = 1;
else
    u = -u;
    %u = -1;
end


dxdt = zeros(2,1);
dxdt(1) = x1 + u;
dxdt(2) = (2*y)+u;
end
```