# Solution 1

[t,y] = ode45(@deriv,[0,0.01],[0,0,0]);

figure(1)

plot(t,y(:,1),'b');

title('Position of mass');

xlabel('Time - [s]');

ylabel('Displacement- [m]');

figure(2)

plot(t,y(:,3),'r');

title('Current in system');

xlabel('Time - [s]');

ylabel('Current - [A]');

function XDOT = deriv(t,X)

    % System Parameters

    m = 0.002;

    k = 8e5;

    c = 2;

    Kf = 16;

    Kb = 16;

    La = 2e-3;

    R = 12;

    % Rename states

    x = X(1); xd = X(2); i = X(3);

    % Initiate forcing function

    v = 10;

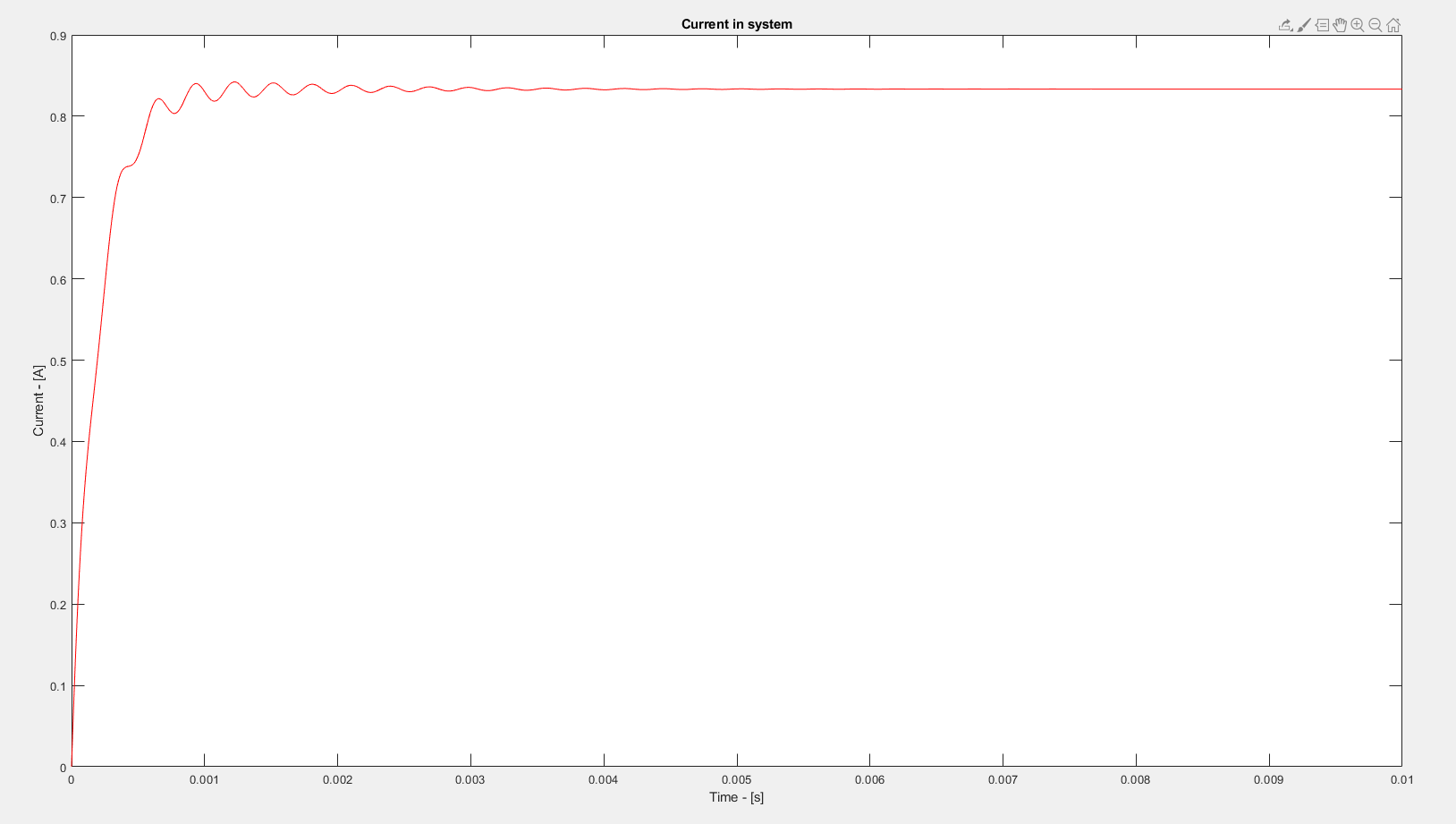
    % write the non-trivial equations using nice names

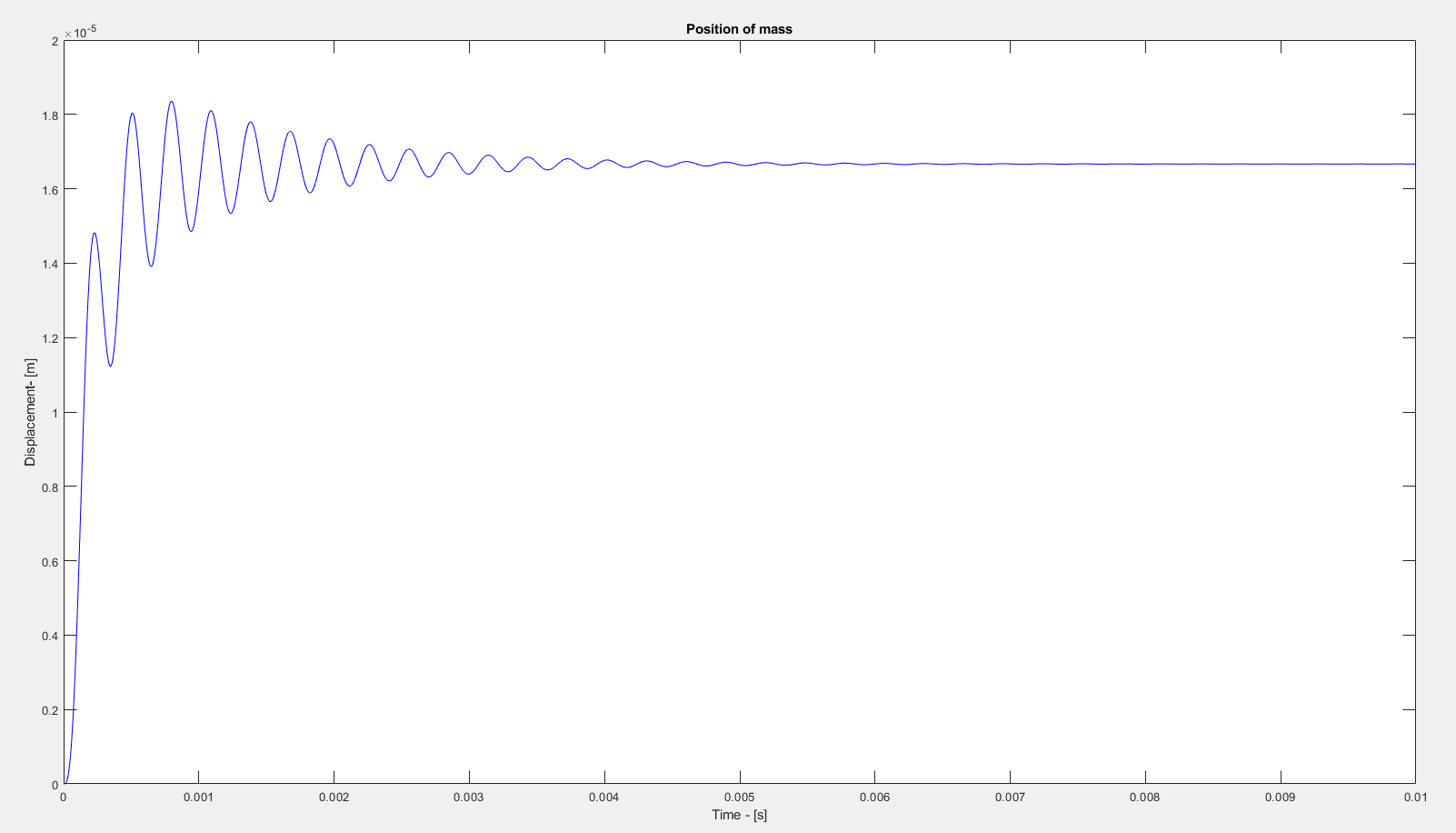
    xdd = (-c\*xd - k\*x + Kf\*i)/m;

    i\_dot = (v - R\*i -Kb\*xd)/La;

    XDOT = [ xd; xdd; i\_dot] ;  %return the derivative values

end

****

****

# Solution 2

clc

global c;

c = 2;

[t1,states1] = ode45(@deriv,[0,0.01],[0,0,0]);

c = 10;

[t2,states2] = ode45(@deriv,[0,0.01],[0,0,0]);

c = 20;

[t3,states3] = ode45(@deriv,[0,0.01],[0,0,0]);

c = 50;

[t4,states4] = ode45(@deriv,[0,0.01],[0,0,0]);

figure(1)

plot(t1,states1(:,1))

hold on

plot(t2,states2(:,1))

hold on

plot(t3,states3(:,1))

hold on

plot(t4,states4(:,1))

hold off

title('C Parameter Study')

xlabel('Time - [s]')

ylabel('Position - [m]')

legend('c = 2', 'c = 10', 'c = 20','c = 50')

function XDOT = deriv(t,X)

    % System Parameters

    m = 0.002;

    k = 8e5;

    global c;

    Kf = 16;

    Kb = 16;

    La = 2e-3;

    R = 12;

    % Rename states

    x = X(1); xd = X(2); i = X(3);

    % Initiate forcing function

    v = 10;

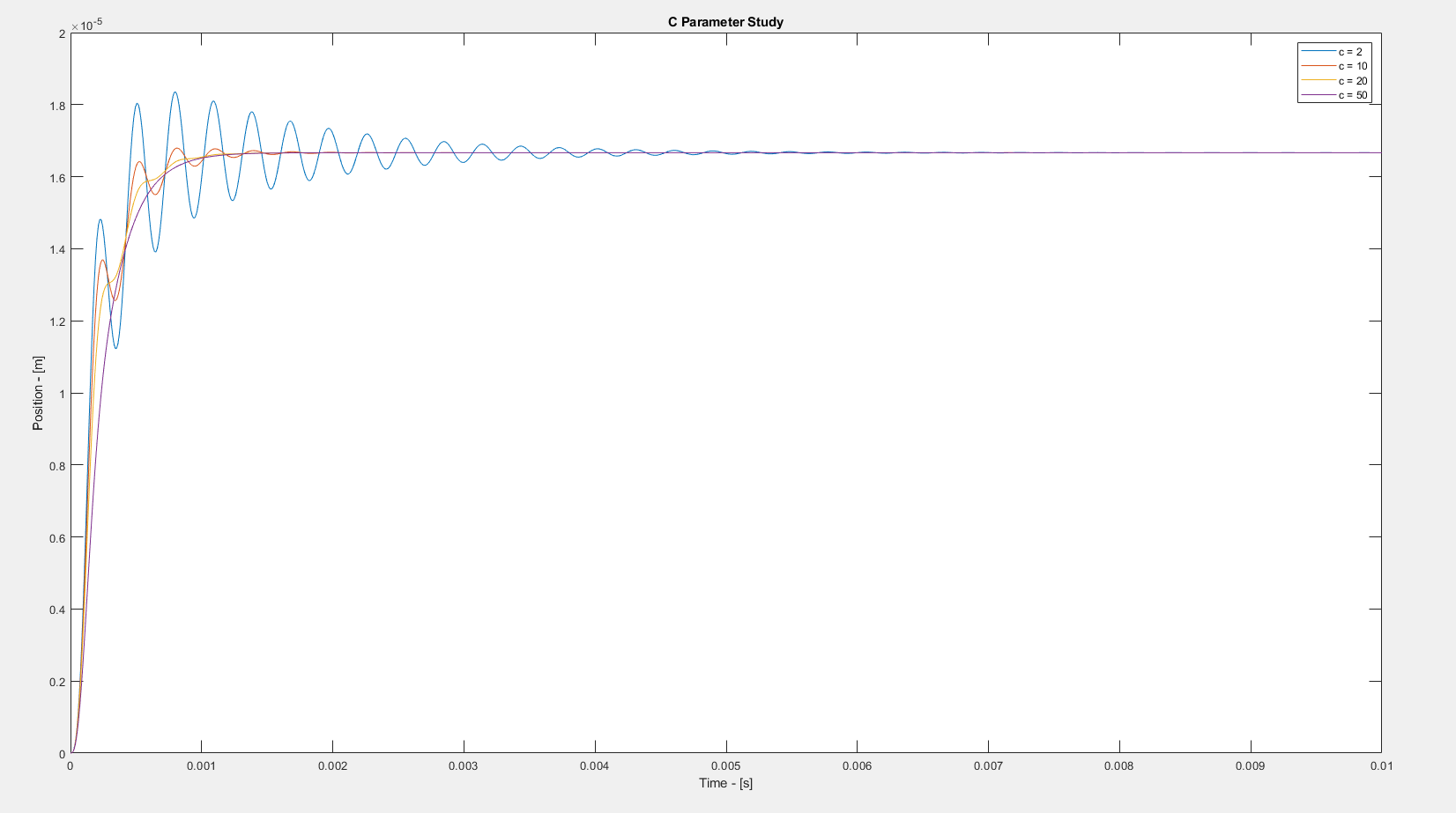
    % write the non-trivial equations using nice names

    xdd = (-c\*xd - k\*x + Kf\*i)/m;

    i\_dot = (v - R\*i -Kb\*xd)/La;

    XDOT = [ xd; xdd; i\_dot] ;  %return the derivative values

end

****

# Solution 3

[t,y] = ode45(@deriv,[0,0.01],[0,0,0]);

x\_analytic = (1.67 - 0.3\*exp(-500.\*t).\*sin(20000.\*t + 1.55))/1e5;

figure(1)

plot(t,y(:,1),'b');

hold on

plot(t,x\_analytic)

hold off

title('Solution Comparison');

xlabel('Time - [s]');

ylabel('Displacement- [m]');

legend('Simulation','Analytic')

function XDOT = deriv(t,X)

    % System Parameters

    m = 0.002;

    k = 8e5;

    c = 2;

    Kf = 16;

    Kb = 16;

    La = 2e-3;

    R = 12;

    % Rename states

    x = X(1); xd = X(2); i = X(3);

    % Initiate forcing function

    v = 10;

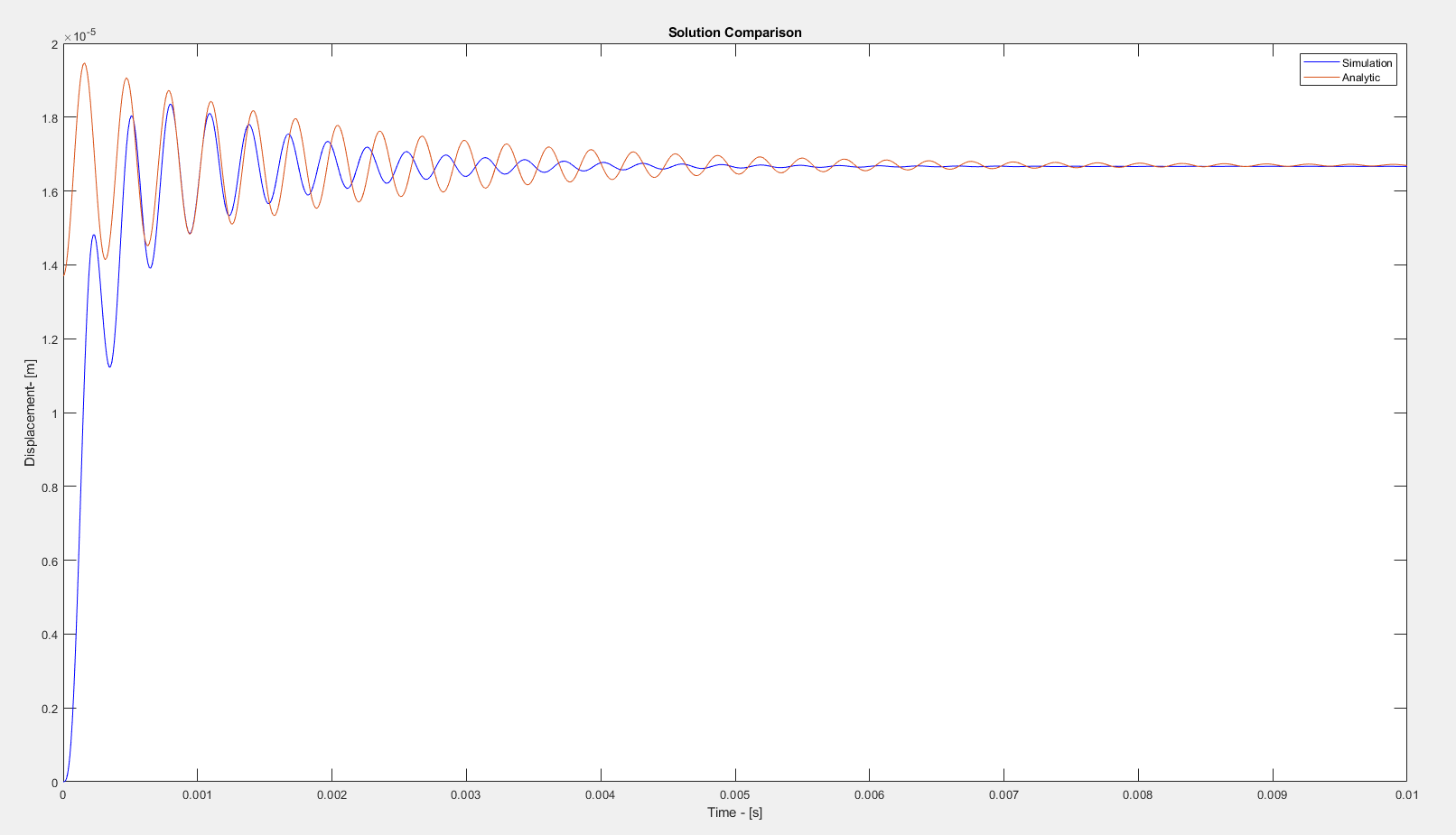
    % write the non-trivial equations using nice names

    xdd = (-c\*xd - k\*x + Kf\*i)/m;

    i\_dot = (v - R\*i -Kb\*xd)/La;

    XDOT = [ xd; xdd; i\_dot] ;  %return the derivative values

end

****

The analytic solution and simulation both have the same steady state value, but the transient response have different frequencies and amplitudes. They also start at different initial conditions.

# Solution 4

m = 0.002;

k = 8e5;

c = 2;

Kf = 16;

Kb = 16;

La = 2e-3;

R = 12;

Va = 10;

num = [Kf];

den = [(m\*La),(c\*La+m\*R),(k\*La + c\*R + Kf\*Kb),(k\*R)];

T = tf(num,den)

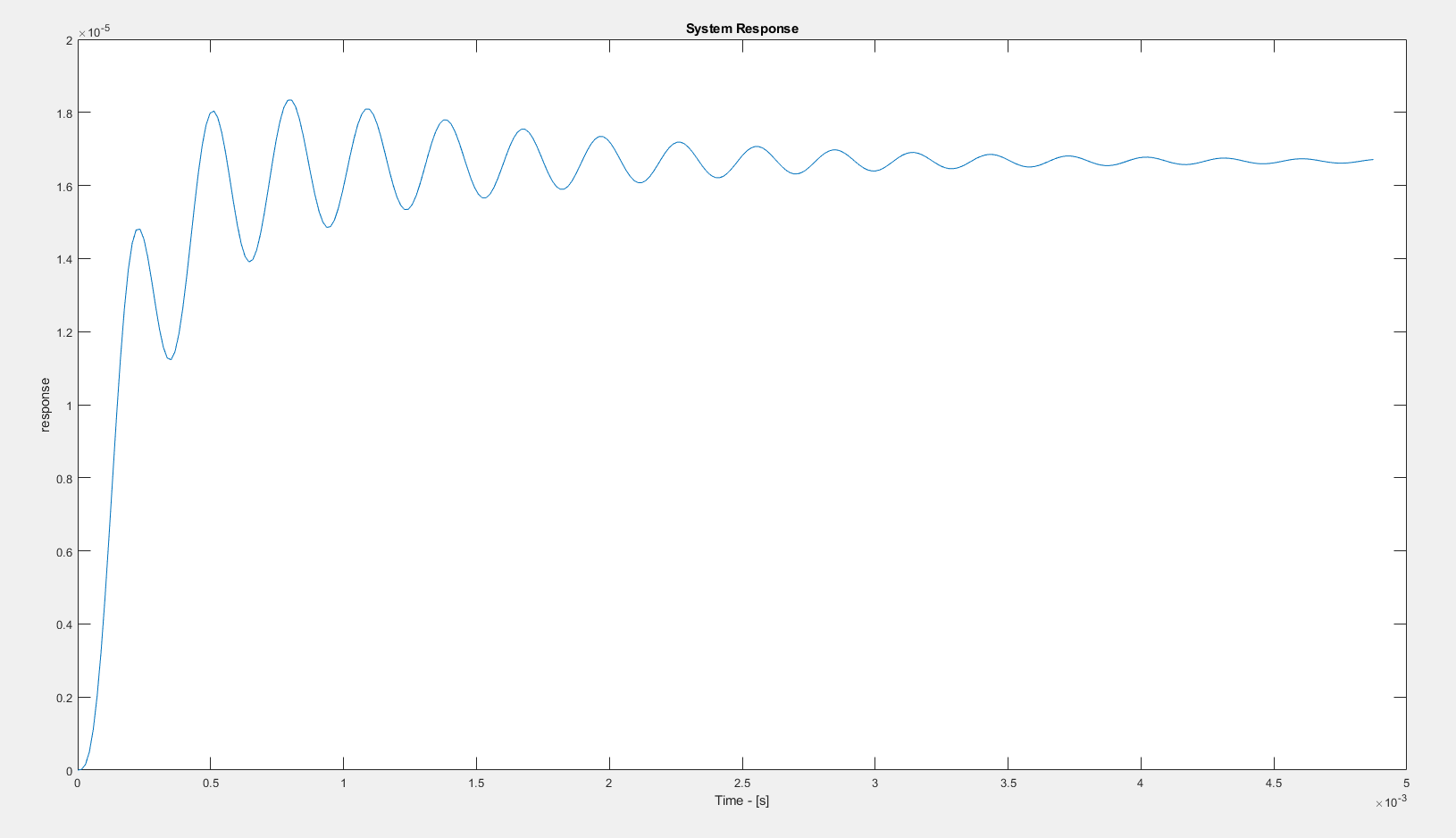
[y,t] = step(T);

plot(t,y\*Va)

title('System Response')

xlabel('Time - [s]')

ylabel('response')

****