EE2703 : Applied Programming Lab
Assignment 6
Laplace Transform

Keerthana Rachuri
EE20B102

March 27, 2022

1 Finding the Laplace Transforms

we are using laplace tranform to solve the spring system,

The laplace transform of the spring equation is of the form $\mathrm{d}^2x/dt^2 + 2.25x = f(t)$

The input signal is of the form f(t) = cos (t) exp(at)u(t), where a is the decay factor and is the frequency of the cosine. The Laplace Transform of the input signal is

F(s) = s + a/ (s+a)$^2 + w^2$

the following code gives us the solution x(t) for the decay of 0.5

```
  p11 = poly1d([1,0.5])
  p21 = polymul([1,1,2.5],[1,0,2.25])
  X1 = sp.lti(p11,p21)
  t1,x1 = sp.impulse(X1,None,linspace(0,50,500))
```

the following code gives us the solution x(t) for the decay of 0.05

```
  p12 = poly1d([1,0.05])
p22 = polymul([1,0.1,2.2525],[1,0,2.25])
X2 = sp.lti(p12,p22)
t2,x2 = sp.impulse(X2,None,linspace(0,50,500))
```

the plot of x(t) vs t for decay 0.5

```
 # x(t) vs t for decay 0.5
figure(0)
plot(t1,x1)
title("The solution x(t) for Q.1")
xlabel(r'$t\rightarrow$')
ylabel(r'$x(t)\rightarrow$')
grid(True)
```

the plot of x(t) vs t for decay 0.05

```
# x(t) vs t for decay 0.05
figure(1)
plot(t2,x2)
title("The solution x(t) for Q.2")
xlabel(r'$t\rightarrow$')
ylabel(r'$x(t)\rightarrow$')
grid(True)
```

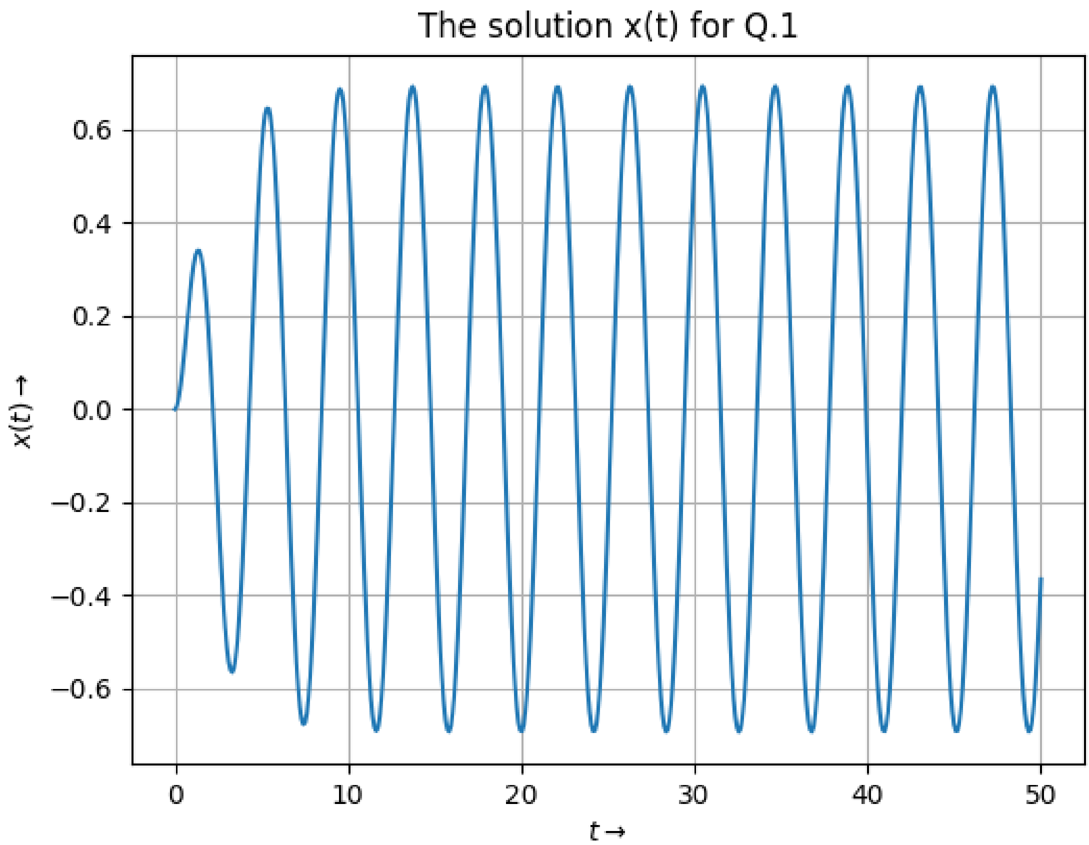

Figure 1.1: x(t) vs t for decay =0.5

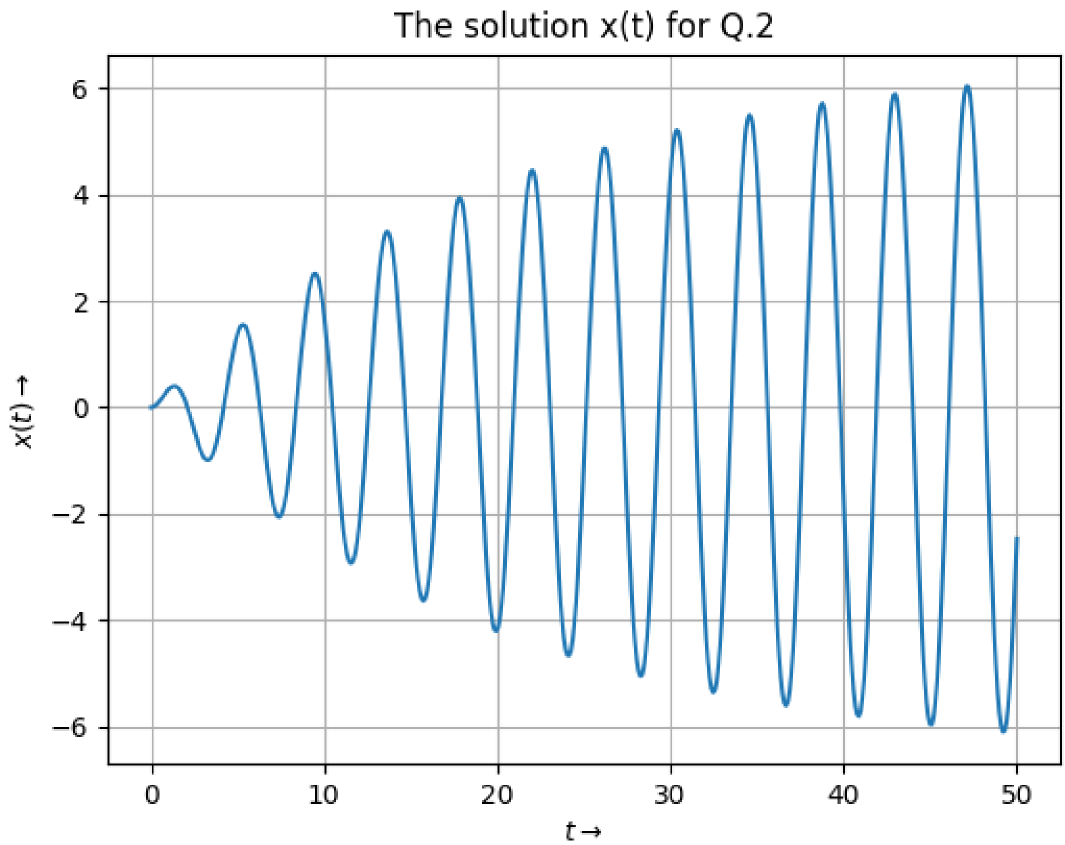

Figure 1.2: x(t) vs t for decay =0.05

Oscillation settles to fixed value after some time but it is faster if the decay is more.

2 Varying the frequency of the input:

we see the effect on output when we vary cos(t) The following code gives us x(t) at different frequencies:

```
H = sp.lti([1],[1,0,2.25])
print(H)
for w in arange(1.4,1.6,0.05):
  t = linspace(0,50,500)
  f = cos(w*t)*exp(-0.05*t)
  t,x,svec = sp.lsim(H,f,t)
```

```
\begin{lstlisting}[language=Python]
H = sp.lti([1],[1,0,2.25])
print(H)
for w in arange(1.4,1.6,0.05):
  t = linspace(0,50,500)
  f = cos(w*t)*exp(-0.05*t)
  t,x,svec = sp.lsim(H,f,t)
```

The plot showing x(t) for different frequencies are as shown below:

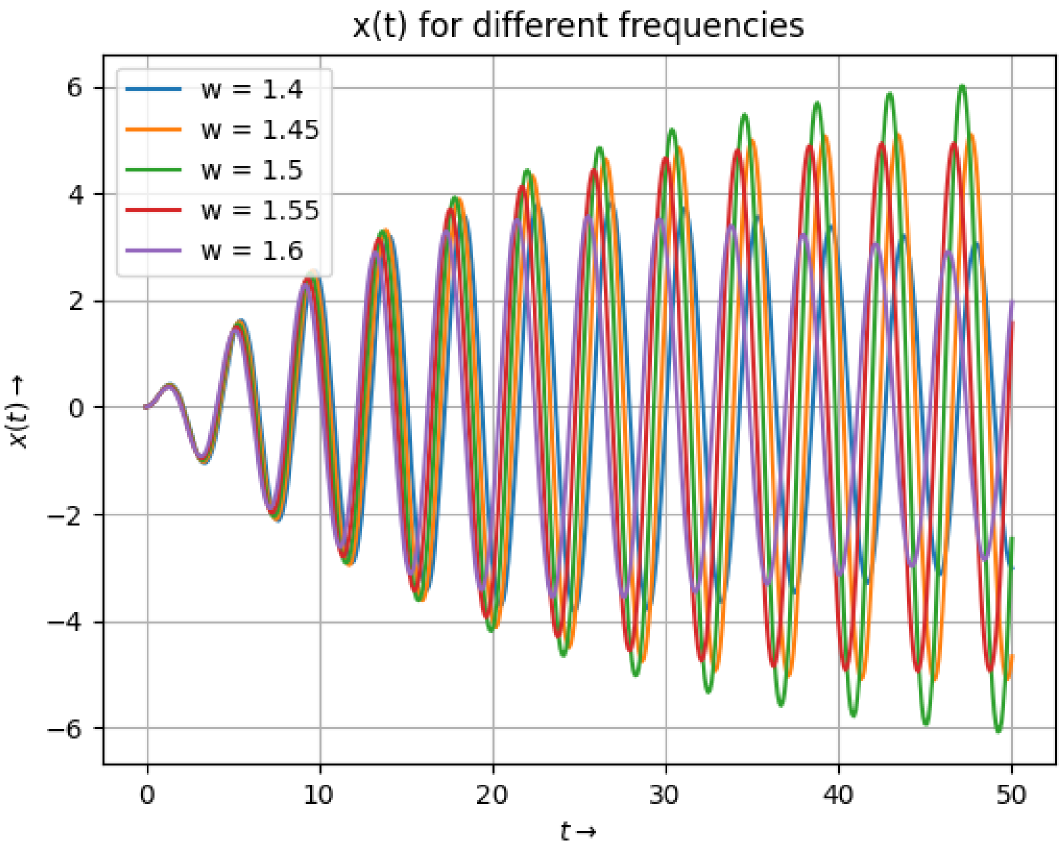

Figure 1.3: x(t) for different frequencies

When the input frequency is at the natural frequency, the output amplitude is maximum. In the other cases the output amplitude decreases. This phenomenon is known as resonance.

3 Coupled Spring Problem

In this problem we have two differential equations and two variables to solve for. The equations are

$\mathrm{d}^x/dt^+(x-y)=0d^y/dt^+2(y-x)=0$

With initial condition as x(0) = 1 We substitute for y in the second equation from the first, and we get a fourth order differential equation in terms of x. Simplifying this and substituting to find the y equation, we get. X(s) = $s^2+2/s^3+3sY(s)=2/s^3+3sWewilltaketheILTofthesetwoexpressionstofi$

```
# Finding x(t) and y(t)
t4 = linspace(0,20,500)
X4 = sp.lti([1,0,2],[1,0,3,0])
```

```
Y4 = sp.lti([2],[1,0,3,0])
t4,x4 = sp.impulse(X4,None,t4)
t4,y4 = sp.impulse(Y4,None,t4)
```

The following code gives us the plot of x(t) and y(t) vs t

```
#  x(t) and y(t) vs t
figure(3)
plot(t4,x4,label='x(t)')
plot(t4,y4,label='y(t)')
title("x(t) and y(t)")
xlabel(r'$t\rightarrow$')
ylabel(r'$functions\rightarrow$')
legend(loc = 'upper right')
grid(True)
```

plots of x(t) and y(t) is shown below:

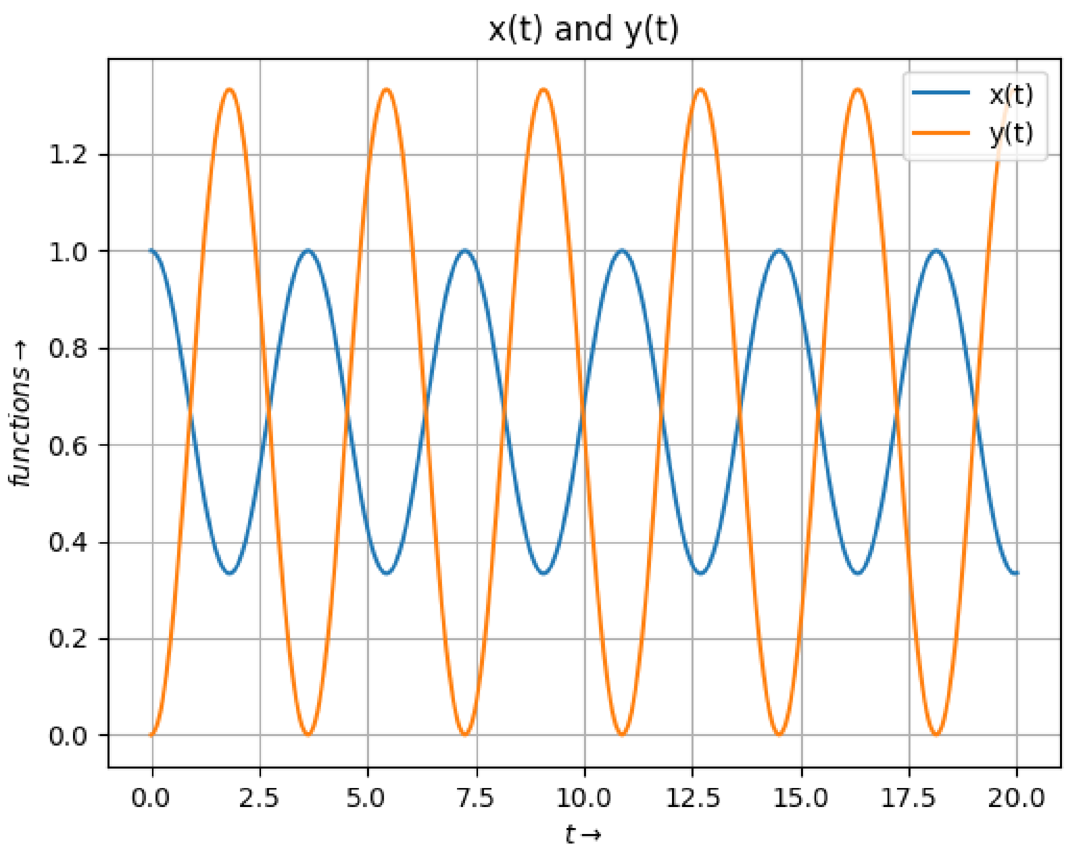

Figure 1.4: plots of x(t) and y(t) VS t

The amplitude of y is greater than x and the phase of the two are opposite

4 RLC Filter:

H(s) = 1/ $(10^-12s^2 + 10^-4s + 1) The input is of the form : x(s) = cos(10^3t) - cos(10^6t)$

we will plot the bode plot of transfer function and then we use sp.lsim to find the output of the function. We plot the output from 0 to 30µs (Small time interval) as well as from 0 to 25ms (Large time interval).

The following code gives us the bode plot of transfer function

```
# Bode plot for H5
temp = poly1d([1e-12,1e-4,1])
H5 = sp.lti([1],temp)
w,S,phi = H5.bode()
```

The following code finds the output

```
#finds the output for the filter function
t6 = arange(0,25e-3,1e-7)
vi = cos(1e3*t6) - cos(1e6*t6)
t6,vo,svec = sp.lsim(H5,vi,t6)
```

The following code gives the bode plot

```
# The magnitude bode plot
figure(4)
semilogx(w,S)
title("Magnitude Bode plot")
xlabel(r'$\omega\rightarrow$')
ylabel(r'$20\log|H(j\omega)|\rightarrow$')
grid(True)

# The phase bode plot
figure(5)
semilogx(w,phi)
title("Phase Bode plot")
xlabel(r'$\omega\rightarrow$')
ylabel(r'$\angle H(j\omega)\rightarrow$')
grid(True)
```

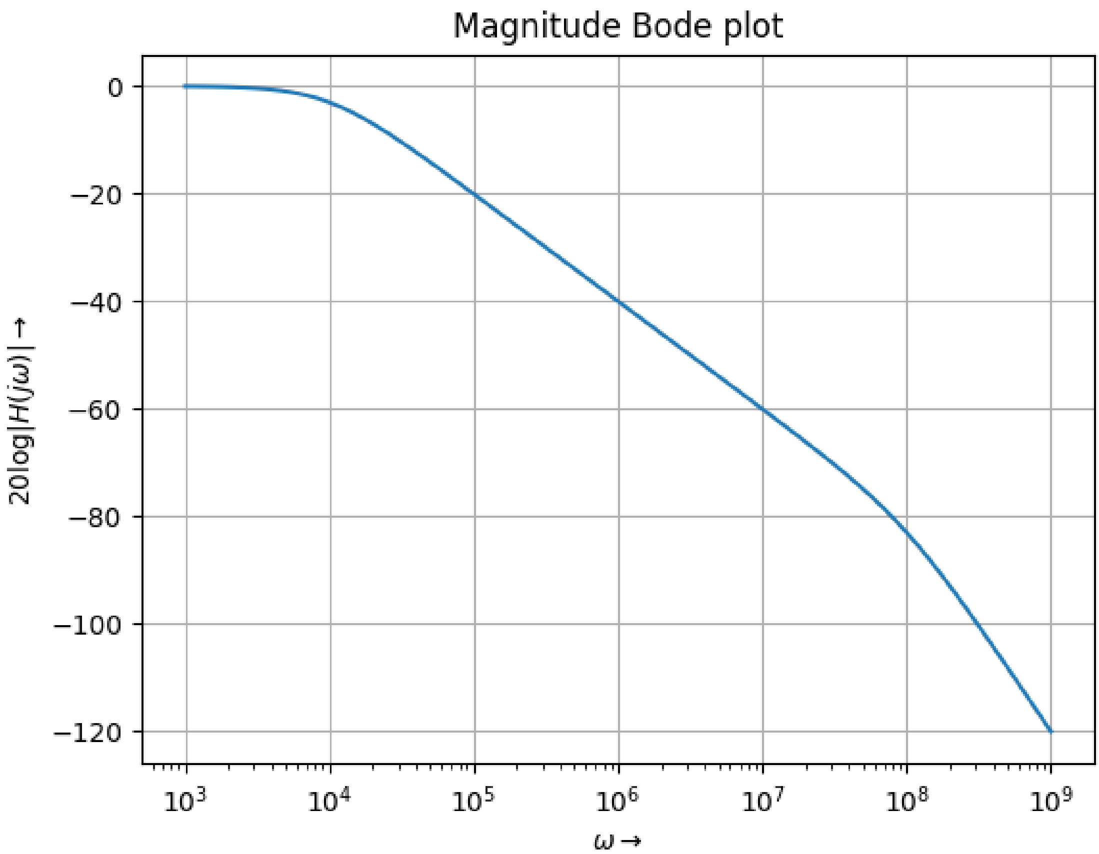

Figure 1.5: Magnitude bode plot

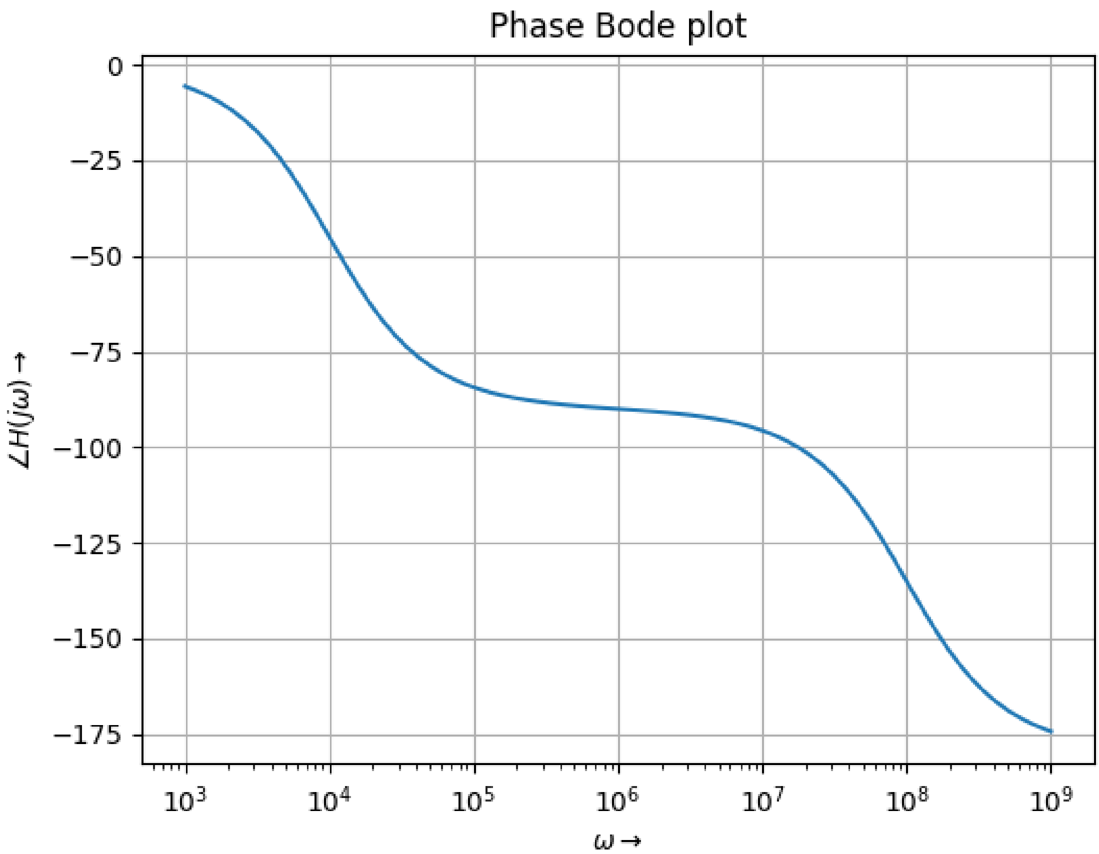

Figure 1.6: Phase bode plot

The following code gives us the plot of Vo(t) vs t for large time interval.

```
# The plot of Vo(t) vs t for large time interval.
figure(6)
plot(t6,vo)
title("The Output Voltage for large time interval")
xlabel(r'$t\rightarrow$')
ylabel(r'$V_o(t)\rightarrow$')
grid(True)
```

The following code gives us the plot of Vo(t) vs t for small time interval.

```
# The plot of Vo(t) vs t for small time interval.
figure(7)
plot(t6[0:300],vo[0:300])
title("The Output Voltage for small time interval")
xlabel(r'$t\rightarrow$')
ylabel(r'$V_o(t)\rightarrow$')
grid(True)
```

The Bode plots for H(s) is as shown:

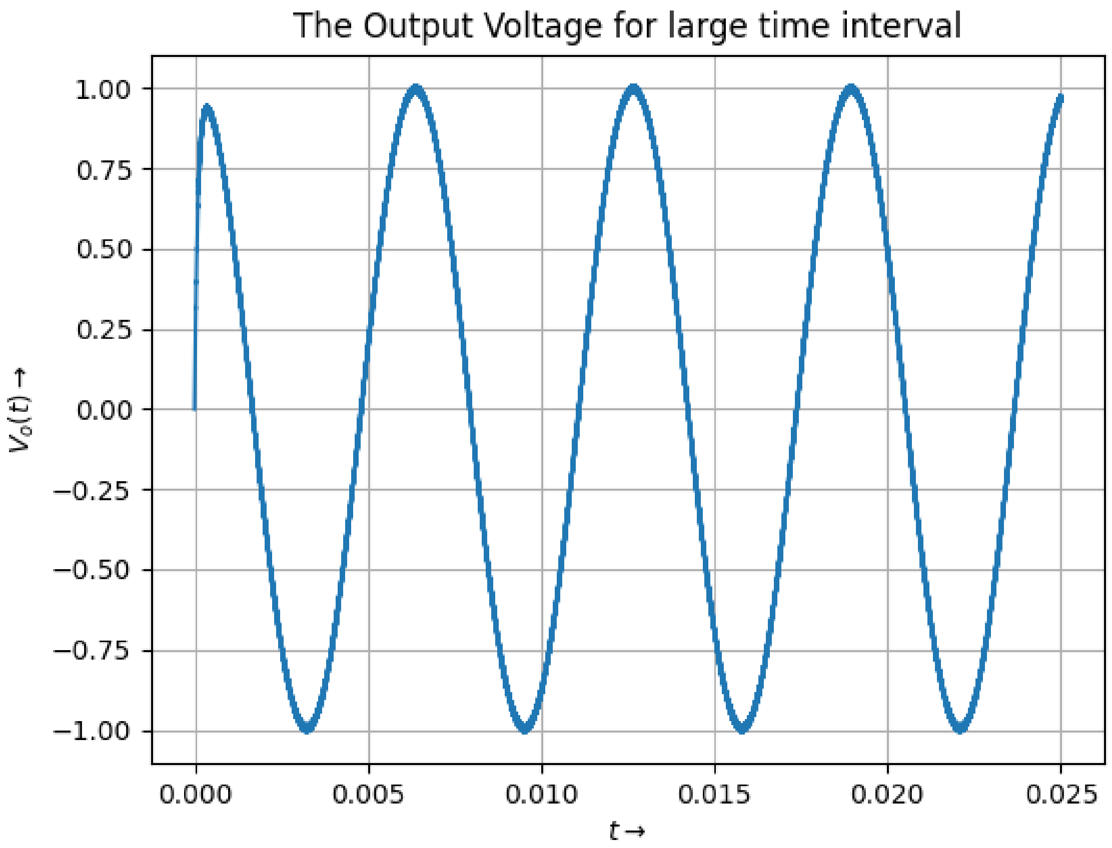

Figure 1.7: Vo(t) vs t for large time interval.

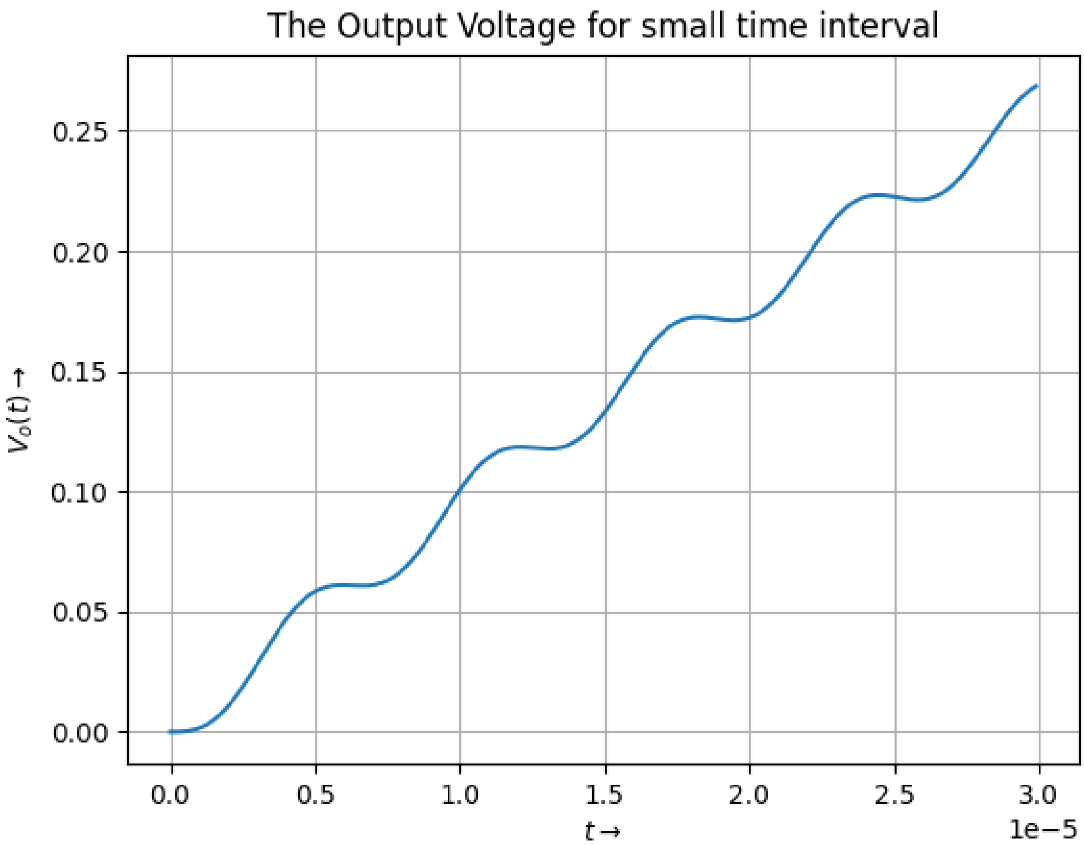

Figure 1.8: Vo(t) vs t for small time interval.

5 Conclusion:

- We analyzed LTI Systems using Laplace Transform.

- We saw a low pass filter constructed from an RLC circuit.

- We used the scipy signals toolkit to calculate the time domain response and the Bode Plot.

- We plotted graphs to understand the above