Assignment6 :Laplace Transform

Gudivada Harshad Kumar
EE20B038

March 18, 2022

1 Abstract

The main aim of the assignment Laplace Transforms is

- Analyze ”Linear Time Invariant” systems using *scipy.signal* in python
- To solve coupled system of differential equations
- To observe transfer function and filter properties of RLC filter
- Analyzing the rational laplace transforms

2 Question 1

The equation representing forced oscillations is given by:

$$\ddot{x} + 2.25x = f(t) \tag{1}$$

The input signal f(t) is given by:
$f(t) = \cos(\omega t)\exp(-at)u(t)$
where a is the decay factor and ω is the frequency of the cosine. Here $\omega = 1.5$ and a = 0.5 We use the following python code snippet to calculate and plot the solution

```
p1 = mp.poly1d([1,0.5])
p2 = mp.polymul([1,0,2.25],[1,1,2.5])
X_1 = sp.lti(p1,p2)
t_1,x_1 = sp.impulse(X_1,None,mp.linspace(0,50,1000))
mp.figure(num=0,figsize = (7,7))
mp.plot(t_1,x_1)
mp.title("The plot x(t) in Q1")
mp.xlabel(r'$t\rightarrow$')
mp.ylabel(r'$x(t)\rightarrow$')
mp.grid(True)
mp.show()
```

The plot of the solution looks:

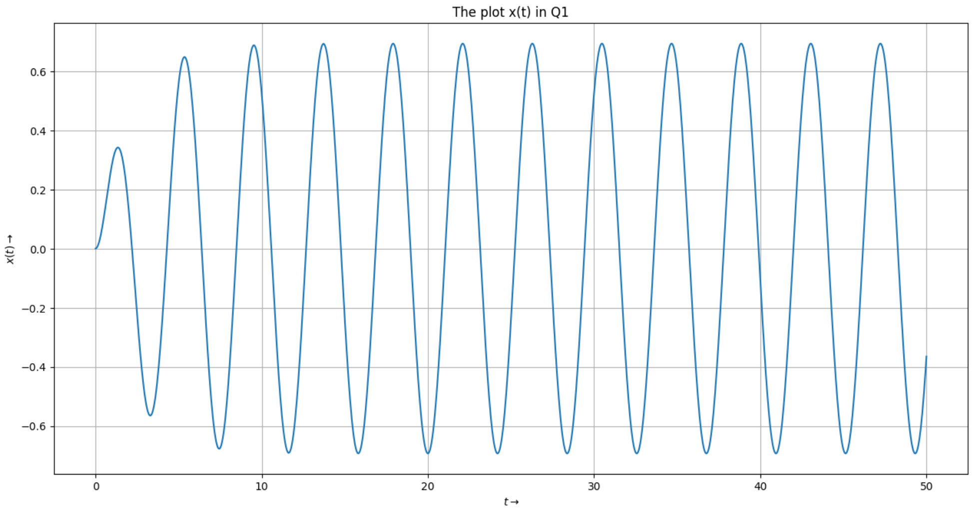

Figure 1: Plot of $x(t)$ for decay of 0.5

3 Question 2

Now the decay is changed to 0.05 the plot for this smaller decay is

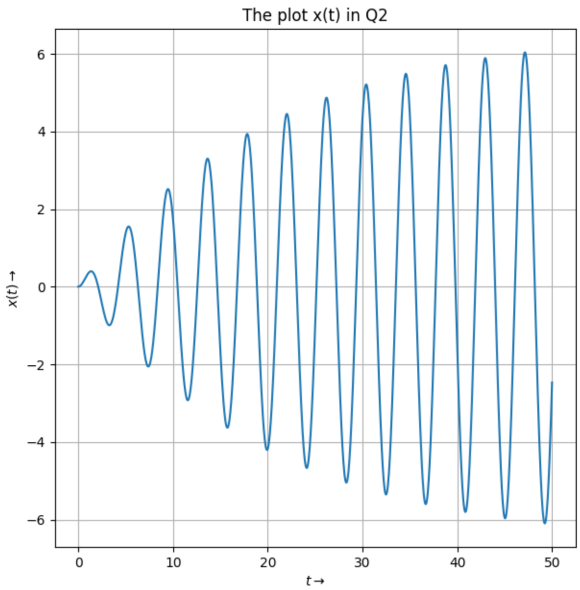

Figure 2: Plot of $x(t)$ for decay of 0.05

4 Question 3

Now the frequency is varied in steps of 0.05 from 1.4 to 1.6, keeping the value of a (decay constant) constant(0.05) The following python code snippet compute the value of $x(t)$ for each frequency and plots them.

```
freq = mp.arange(1.4,1.6,0.05)
H = sp.lti([1],[1,0,2.25])
for i in freq:
    t =  mp.linspace(0,50,1000)
    f = mp.cos(i*t)*mp.exp(-0.05*t)
    t_3,x_3,svec = sp.lsim(H,f,t)
    mp.figure(num=2,figsize=(7,7))
    mp.plot(t_3,x_3,label='w = ' + str(i))
    mp.title("Plot of x(t) for different frequencies")
    mp.xlabel(r'$t\rightarrow$')
    mp.ylabel(r'$x(t)\rightarrow$')
    mp.grid(True)
    mp.legend()
mp.show()
```

The plot of $x(t)$ for different frequencies is:

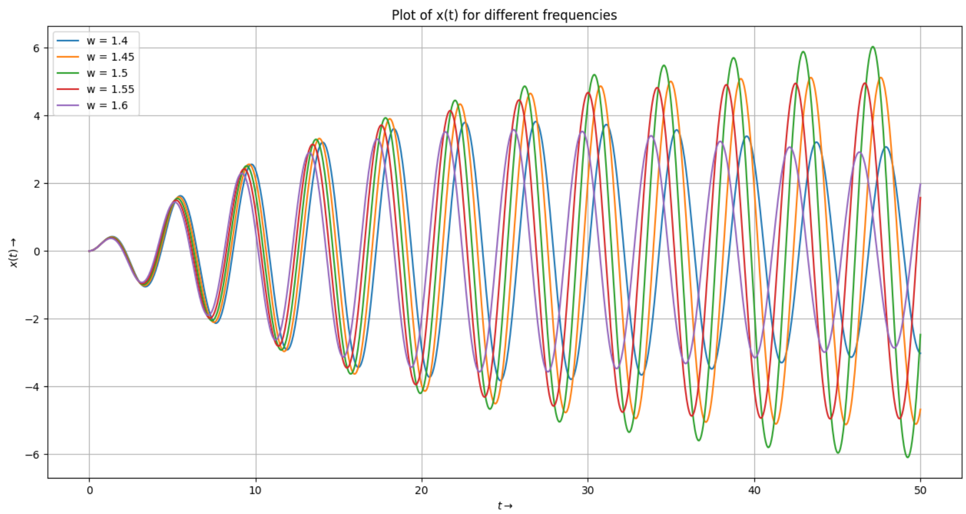

Figure 3: Plot of $x(t)$ for decay of 0.05

5 Question 4

The given coupled differential equations are:

$$\ddot{x} + (x - y) = 0 \tag{2}$$

and

$$\ddot{y} + 2(y - x) = 0 \tag{3}$$

Initial conditions: $\dot{x}(0) = 0, \dot{y}(0) = 0, x(0) = 1, y(0) = 0$.
The laplace transforms of $X(s)$ and $Y(s)$ are:

$$X(s) = \frac{s^2 + 2}{s^3 + 3s} \tag{4}$$

$$Y(s) = \frac{2}{s^3 + 3s} \tag{5}$$

The following code computes the values of $x(t)$ and $y(t)$ and plots them

```
t4 = mp.linspace(0,20,1000)
X_4 = sp.lti([1,0,2],[1,0,3,0])
Y_4 = sp.lti([2],[1,0,3,0])
t_4,x_4 = sp.impulse(X_4,None,t4)
t_4,y_4 = sp.impulse(Y_4,None,t4)
mp.figure(num=3,figsize=(7,7))
mp.plot(t_4,x_4,label='x(t)')
mp.plot(t_4,y_4,label='y(t)')
mp.title("x(t) and y(t)")
mp.xlabel(r'$t\rightarrow$')
mp.ylabel(r'$functions\rightarrow$')
mp.legend()
mp.grid(True)
mp.show()
```

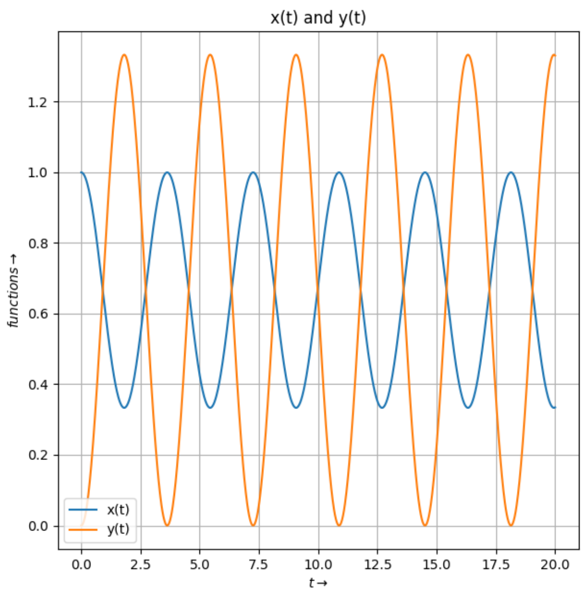

Figure 4: Coupled Oscillations

6 Question 5

Now we calculate the transfer function of given filter and plot bode plots. The following python code snippet helps us to plot the bode plots

```
den = mp.poly1d([1e-12,1e-4,1])
H_C = sp.lti([1],den)
w,S,phi=H_C.bode()
mp.figure(num=4,figsize = (7,7))
mp.subplot(2,1,1)
mp.title("Magnitude plot")
mp.xlabel(r'$\omega\rightarrow$',loc ='left')
mp.ylabel(r'$20\log|H(j\omega)|\rightarrow$')
mp.grid(True)
mp.semilogx(w,S)
mp.subplot(2,1,2)
mp.title("Phase plot")
mp.semilogx(w,phi)
mp.xlabel(r'$\omega\rightarrow$',loc = 'left')
mp.ylabel(r'$\phi\rightarrow$')
mp.grid(True)
mp.show()
```

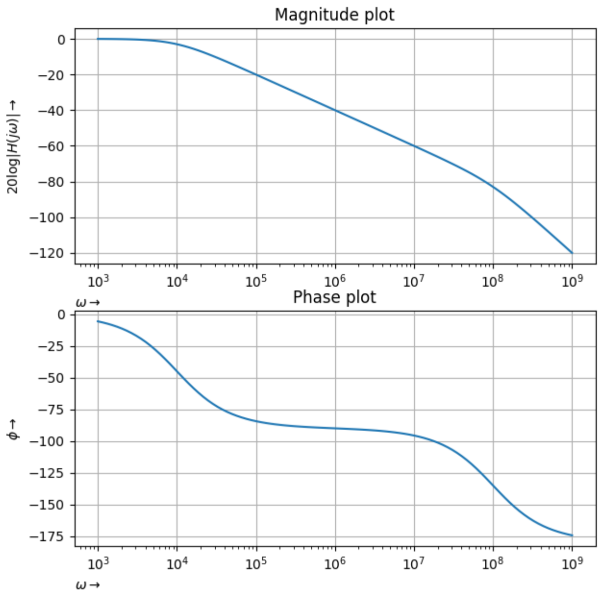

Figure 5: Bode Plots For RLC Low pass filter

7 Question 6

For the given input we plot the response
The input is: $V_i(t) = (cos(10^3t) - cos(10^6t))u(t)$
For solving we use the following python code snippet.

```
t6 = mp.arange(0,1e-3,1e-7)
vi_t = mp.cos(1e3*t6) - mp.cos(1e6*t6)
t_6,vo_t,vsec = sp.lsim(H_C,vi_t,t6)
```

for $0 < t < 30\mu s$, we use the following python code snippet

```
mp.figure(num=5,figsize = (7,7))
mp.plot(t_6[0:300],vo_t[0:300])
mp.title("The Output Voltage over small time interval")
mp.xlabel(r'$t\rightarrow$')
mp.ylabel(r'$V_o(t)\rightarrow$')
mp.grid(True)
```

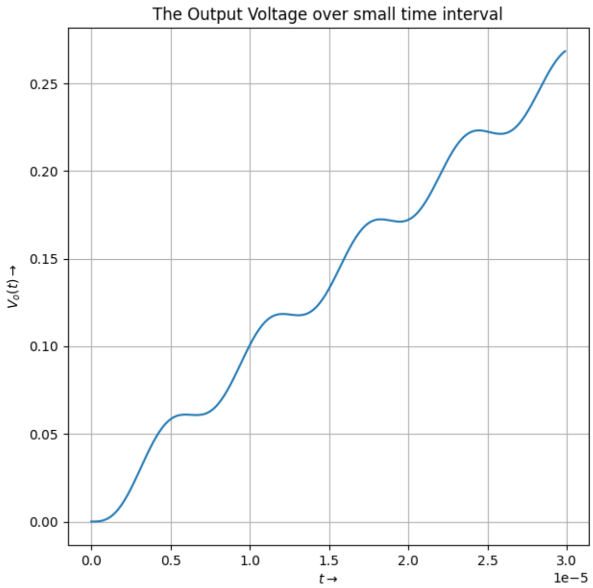

Figure 6: System response for $t < 30\mu s$

for $0 < t < 10ms$, we use the following python code snippet

```
mp.figure(num=6,figsize = (7,7))
mp.plot(t_6,vo_t)
mp.title("The Output Voltage over large time interval")
```

```
mp.xlabel(r'$t\rightarrow$')
mp.ylabel(r'$V_o(t)\rightarrow$')
mp.grid(True)
mp.show()
```

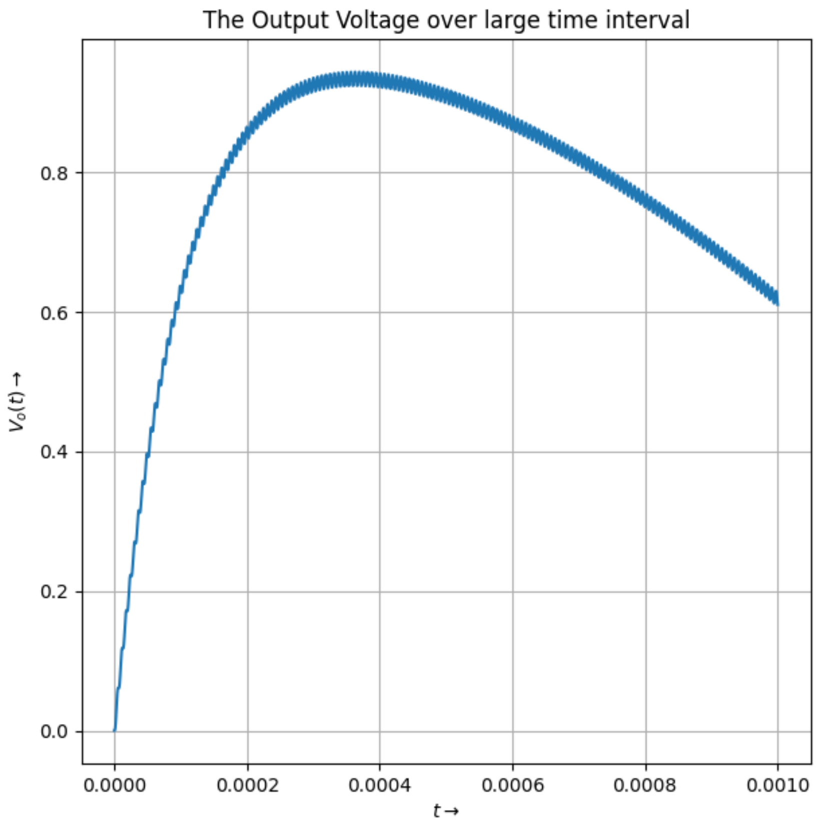

Figure 7: System response for $t < 10ms$

8 conclusion

By solving the above equations we realise the importance of scipy.signal module in Python in analysing and solving the problems in Laplace Domain. We solved forced oscillations of spring, coupled oscillations and Filter circuits.