EE2703 : Applied Programming Lab
Assignment 7
Laplace Transformation

Arun Krishna A M S
EE19B001

April 21, 2021

Objective

Most general systems around us can be modelled as **Linear Time-invariant Systems** and is extensively used in Electrical Engineering. Example: Linear Circuit Analysis. In this assignment, we try to attain the following objectives

- Analysis of continuous time LTI systems in laplace domain through python libraries
- Solve **LCCDE - Linear Constant Coefficient Differential Equations** in laplace domain using the `signal` toolbox of `scipy` library.
- Explore various functions of the above mentioned library like `impulse, bode, lti, lsim`

Section 1

The time response of a lossless spring system is given by

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

where $f(t)$ is the forced input on the spring system. Suppose if the forced input $f(t)$ is a decaying sinusoidal force as given by

$$f(t) = cos(1.5t)e^{-0.5t}u(t)$$

In Laplace domain

$$F(s) = \frac{s+0.5}{(s+0.5)^2+2.25}$$

with $x(0) = 0$ and $\dot{x} = 0$ input conditions. This corresponds to

$$s^2X(s) + 2.25X(s) = F(s)$$

$$H(s) = s^2 + 2.25 = \frac{F(s)}{X(s)}$$

$$X(s) = \frac{s+0.5}{(s^2+2.25)((s+0.5)^2+2.25)}$$

When the damping coefficient is 0.05

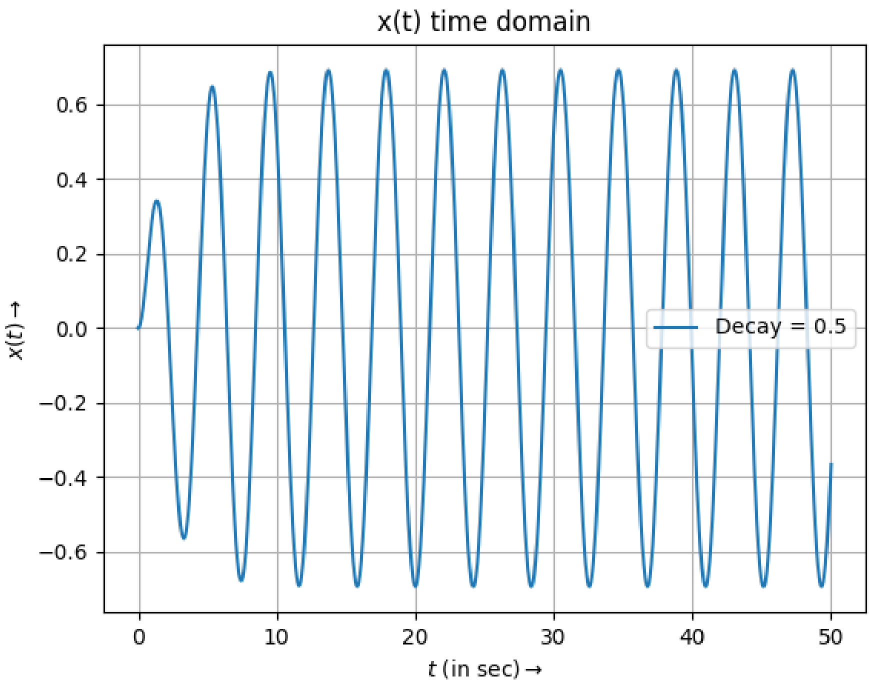
$$f(t) = cos(1.5t)e^{-0.05t}u(t)$$

$$F(s) = \frac{s+0.05}{(s+0.05)^2+2.25}$$

$$X(s) = \frac{s+0.5}{(s^2+2.25)((s+0.5)^2+2.25)}$$

We try to find the time domain form of $X(s)$ using the `impulse` function (`a` is the decay coefficient)

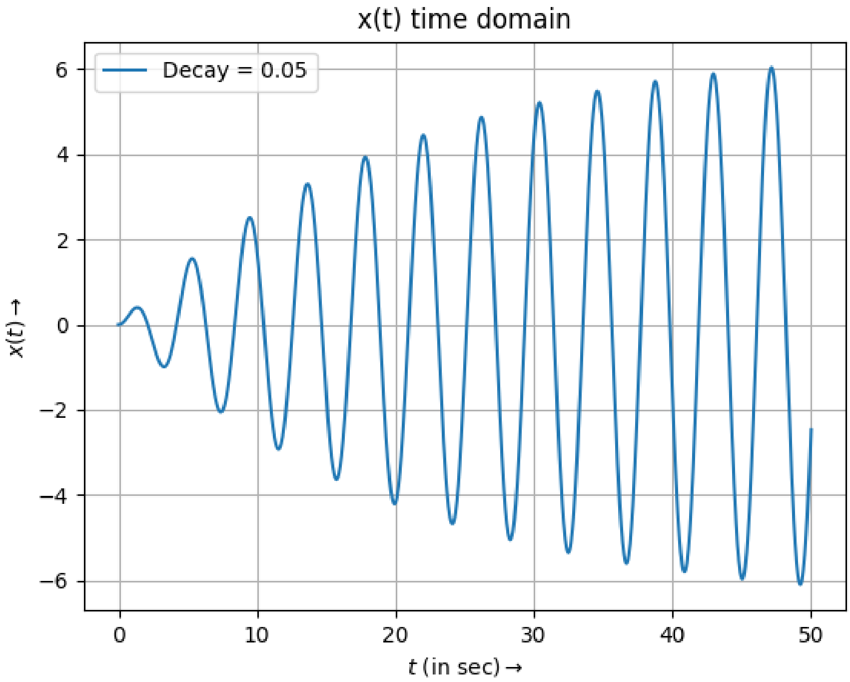
```
#Time vector going from 0 to 50 seconds
t = np.linspace(0,50,1000)

#Laplace domain expression for X(s) - derivation in report
X = sp.lti([1, a],np.polymul([1,0,2.25], np.polyadd(np.polymul([1, a],[1, a]),[2.25])))

#Time domain function values
t, x = sp.impulse(X, None, t)
```



X(t) in time domain for decay coefficient 0.5

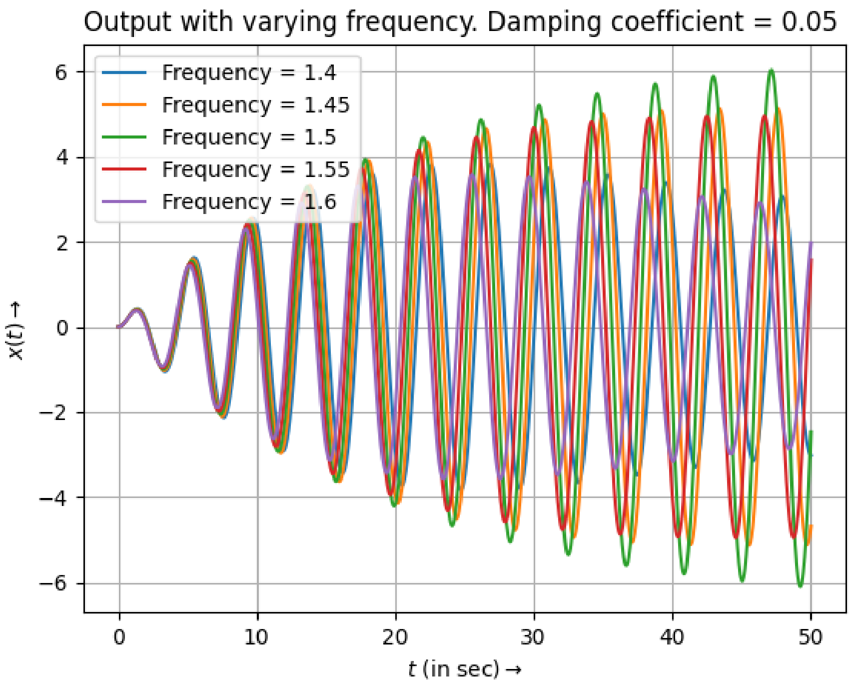
We can observe that the output of the system when the decay coefficient = 0.05 (in the next graph) has higher amplitude - since the energy supplied by the forced input in case of decay coefficient = 0.05 will be higher compared to decay coefficient = 0.5 as the former one decays slower compared to the second one.

Suppose if the decay coefficient = 0, then the forced input with oscillation frequency $\omega = 1.5$ resonates with the natural frequency of the system thus blowing up the output. This because the frequency response $H(j\omega)$ has a double pole at $\omega = 1.5$



X(t) in time domain for decay coefficient 0.05

We can also notice this while plotting the output for various frequencies. The output with frequency 1.5 rad/s corresponds to the maximum amplitude



Section 2

We try to solve the following coupled spring problem with:

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

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

with initial conditions $x(0) = 1$, $\dot{x}(0) = y(0) = \dot{y}(0) = 0$. Solving further

$$s^2X(s) - sx(0^-) - \dot{x}(0^-) = Y(s)$$

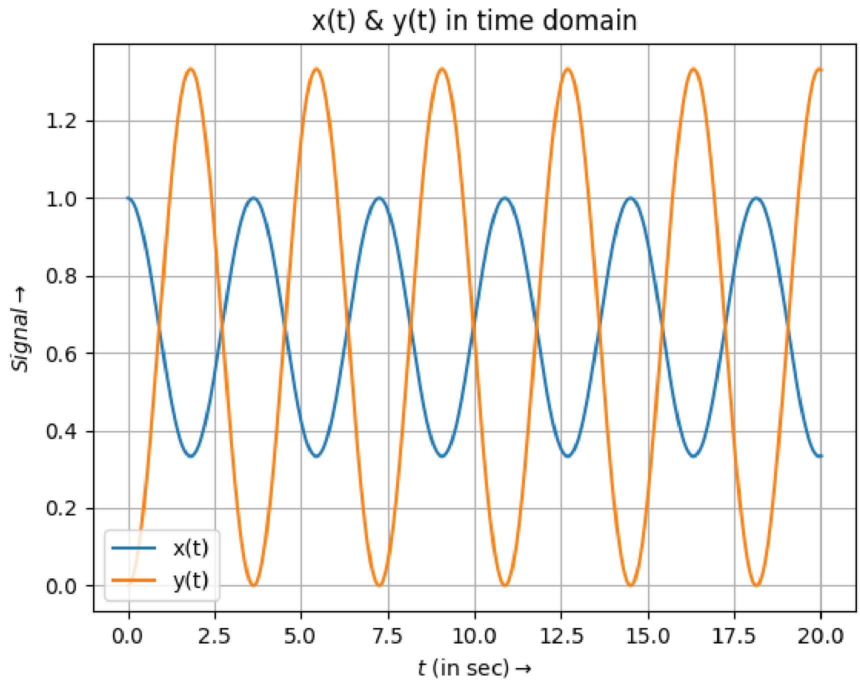
$$s^2Y(s) - sy(0^-) - \dot{y}(0^-) + 2Y(s) = X(s)$$

Substituting and solving further, we arrive at

$$X(s) = \frac{(0.5s^2 + 1)s}{(s^2 + 1)(0.5s^2 + 1) - 1}$$

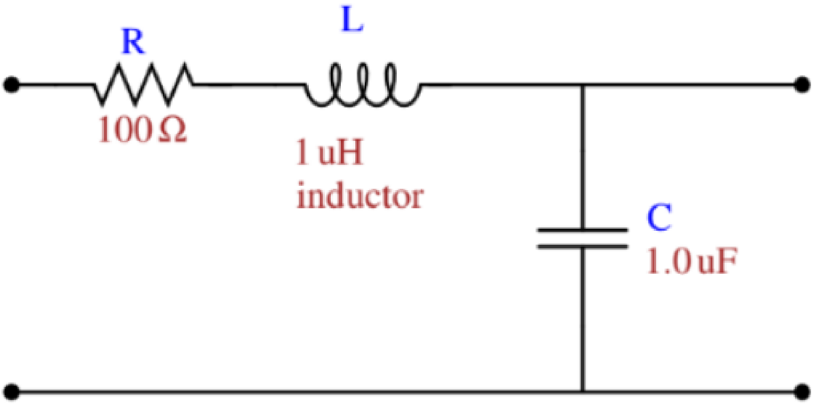
$$Y(s) = \frac{s}{(s^2 + 1)(0.5s^2 + 1) - 1}$$

```
t = np.linspace(0, 20, 1000)

X = sp.lti(np.polymul([1, 0], [0.5, 0, 1]), np.polyadd(np.polymul([1, 0, 1], [0.5, 0, 1]),
Y = sp.lti([1, 0], np.polyadd(np.polymul([1, 0, 1], [0.5, 0, 1]), [-1]))

t, x = sp.impulse(X, None, t)
t, y = sp.impulse(Y, None, t)
```



Section 3

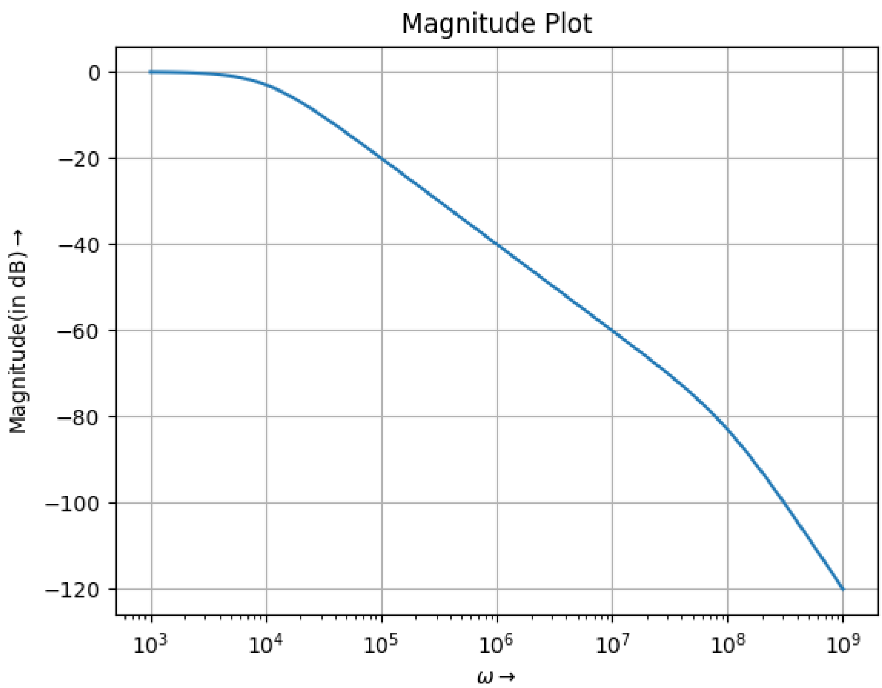
The next part of the problem is to obtain the magnitude and phase response of the steady state transfer function of the RLC Circuit.

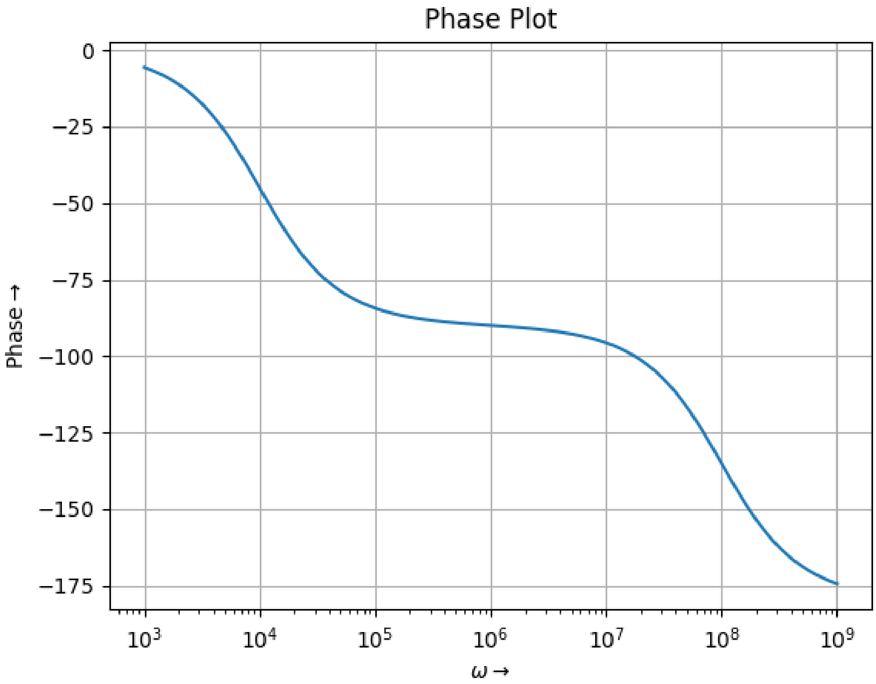


We can observe that the circuit acts as a voltage divider. Thus it reduces to the following transfer equation

$$H(s) = \frac{1}{1 + RCs + LCs^2} = \frac{1}{1 + 10^{-4}s + 10^{-12}s^2}$$

```
H = sp.lti([1], [1e-12, 1e-4, 1])
w, S, phi = H.bode()
```





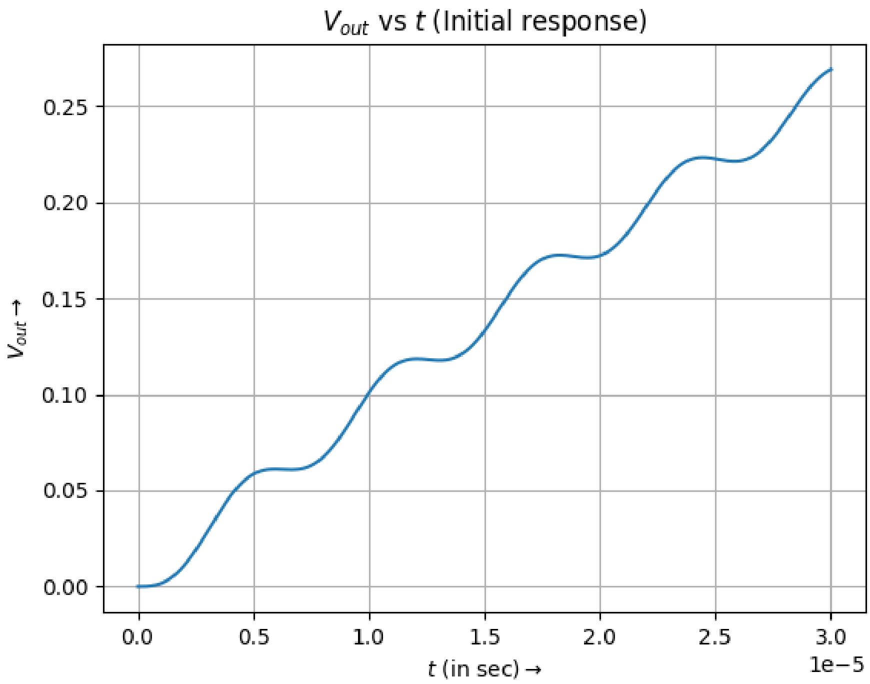
We can observe that there are two major deflections in both graphs indicating the two poles of the system. The magnitude plot remains constant until it sees a pole after which it decreases rapidly at -20 dB/decade. It further sloped down to -40 dB/decade after it encounters the second pole. In the phase plot, we can observe that there is a phase drop of approximately 90 degrees at each pole.

In the given circuit, if the applied input voltage is of form,

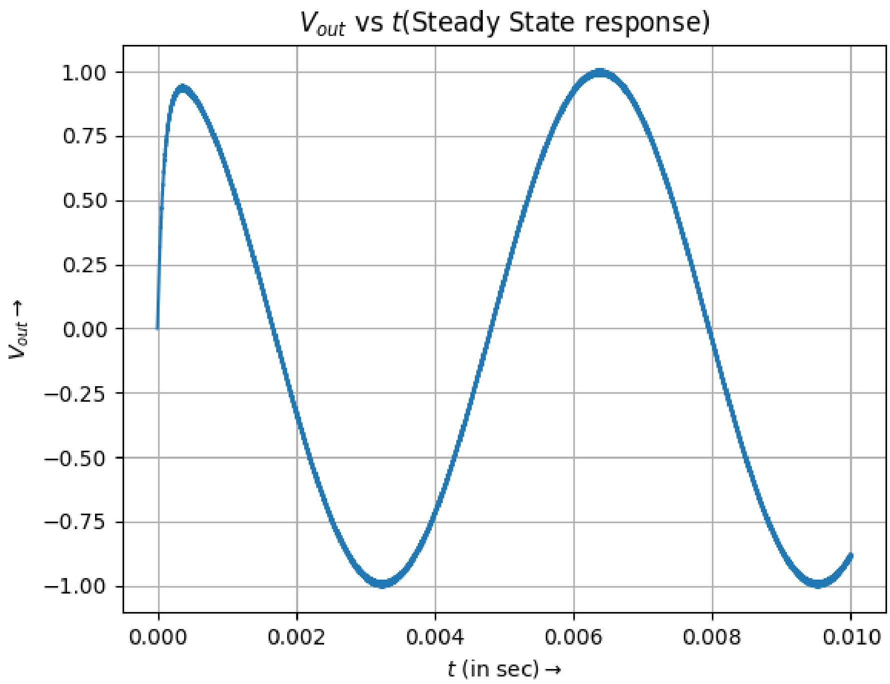
$$V_{in}(t) = cos(10^3 t)u(t) - cos(10^6 t)u(t)$$

We obtain the final output V_{out} of the RLC circuit by

```
#Intitial short term response
t = np.linspace(0, 30e-6, 10000)
Vin = np.cos(1e3*t)-np.cos(1e6*t)
t, Vout, svec = sp.lsim(H, Vin, t)

#Long term response
t = np.linspace(0, 1e-2, 10000)
Vin = np.cos(1e3*t)-np.cos(1e6*t)
t, Vout, svec = sp.lsim(H, Vin, t)
```



The above graph corresponds to the initial transient response of the system. We can observe that the sinusoidal component `cos(1e6t)` is showed up as ups and downs in the graph.



But in steady state, we can observe that the output is primarily composed of 1000 rad/s frequency while the frequency 10^6 rad/s is almost attenuated. This is because the later frequency experiences attenuation of about 100 times as shown in the magnitude plot of the transfer function. Essentially the given circuit acts as a low pass filter supporting frequencies upto 1000 rad/s.

Conclusion

In this assignment, we explored the idea of solving laplace equations of various systems like spring system, coupled spring problem and a RLC system using the `signal` toolbox of the `scipy` python library. We observed that when the forced input operates at a frequency close to the natural frequency it blows up the output. We also observed how a RLC circuit can be used as a low pass filter