Assignment No. 6

Srivenkat A (EE18B038)

March 11, 2020

Introduction

The assignment focusses to use **scipy.signal** library to perform Linear Circuit Analysis.

Task 1

Given the time variant signal f(t) :

$$f(t) = cos(1.5t)e^{-0.5t}u_0(t)$$

Its Laplace transform is known to be:

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

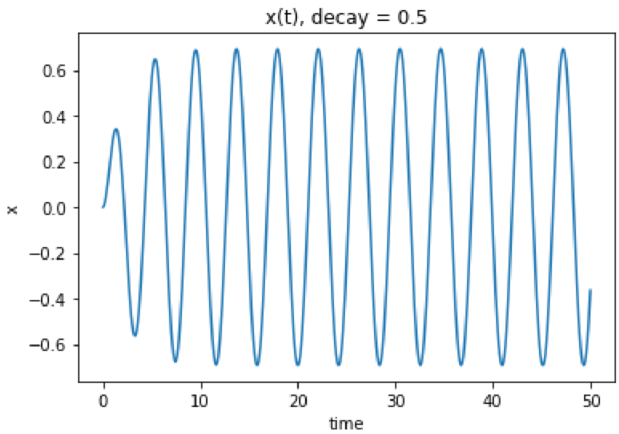
To obtain the time response of a spring system satisfying

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

we find X(s) from the time response of the spring and use *system.impulse* to find the inverse x(t)

```
num = np.poly1d([1,0.5])
den = np.poly1d([1,1,2.5])
den = np.polymul(den,np.poly1d([1,0,2.25]))
X = sig.lti(num,den)
time,x = sig.impulse(X,None,np.linspace(0,50,500))
plt.plot(time,x)
plt.show()
```

Under the initial condition x(0) = 0, $\dot{x}(x) = 0$, The plot of x(t) vs time can be seen as:



Task 2

Given the time variant signal f(t) with a smaller decay coefficient :

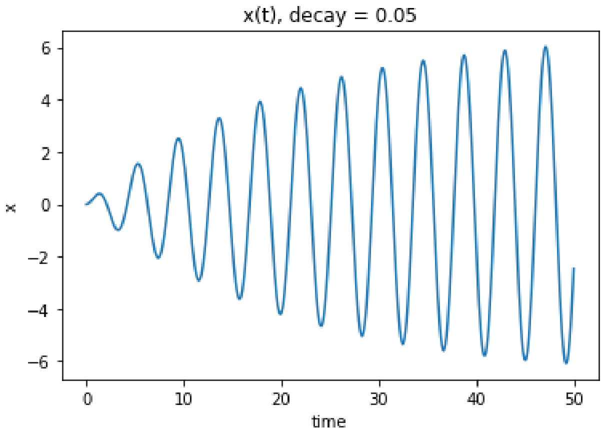
$$f(t) = cos(1.5t)e^{-0.05t}u_0(t)$$

We obtain the Laplace transform to be:

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

```
num = np.poly1d([1,0.05])
den = np.poly1d([1,0.1,2.2525])
den = np.polymul(den,np.poly1d([1,0,2.25]))
X = sig.lti(num,den)
time,x = sig.impulse(X,None,np.linspace(0,50,500))
plt.plot(time,x)
plt.show()
```

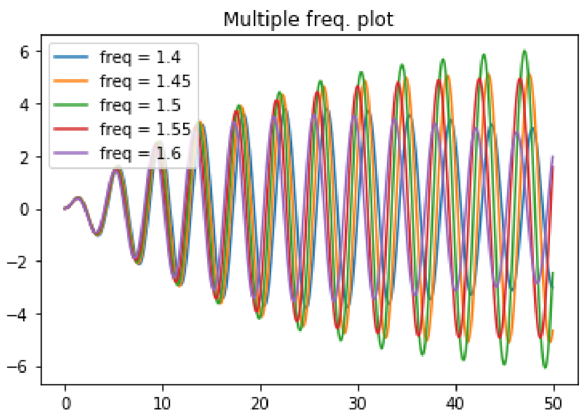
Finding the corresponding the time response of the spring and plotting it,



Task 3

For a fixed decay coefficient of 0.05 and a set of frequencies between 1.4 and 1.6 in steps of 0.05, we try to plot the corresponding time responses pf the spring

```
H = np.poly1d([1,0,2.25])
H = sig.lti(1,H)
time = np.linspace(0,50,500)
for i in range(5):
    var = 1.4 + i*0.05
    f = np.cos(var*time)*np.exp(-0.05*time)
    time,x,svec = sig.lsim(H,f,time)
    plt.plot(time,x)
plt.show()
```



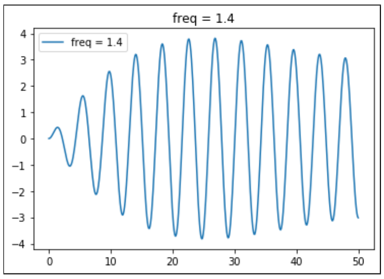


Figure 1: freq = 1.4

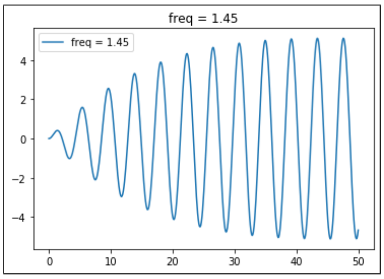


Figure 2: freq = 1.45

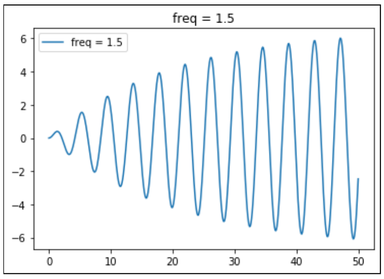


Figure 3: freq = 1.5

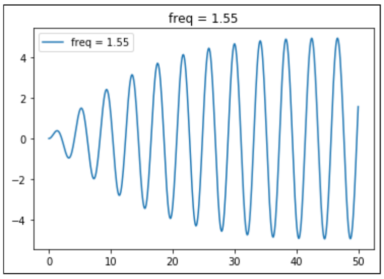


Figure 4: freq = 1.55

For the equation, the natural response has frequency - 1.5rad/s. So, max. amplitude is applied for freqc = 1.5rad/s case.

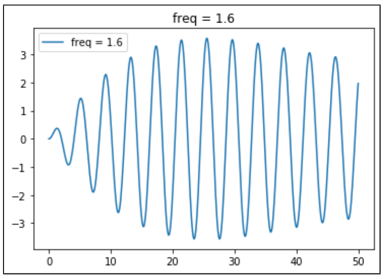


Figure 5: freq = 1.6

Task 4

For a set of coupled springs related by the response equations as:

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

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

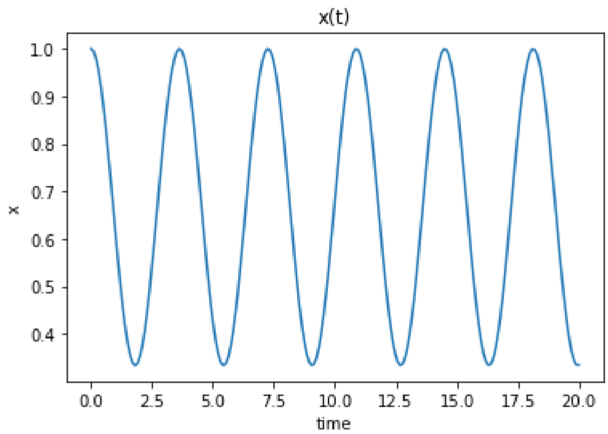
we try to find the individual x(t) and y(t) responses.

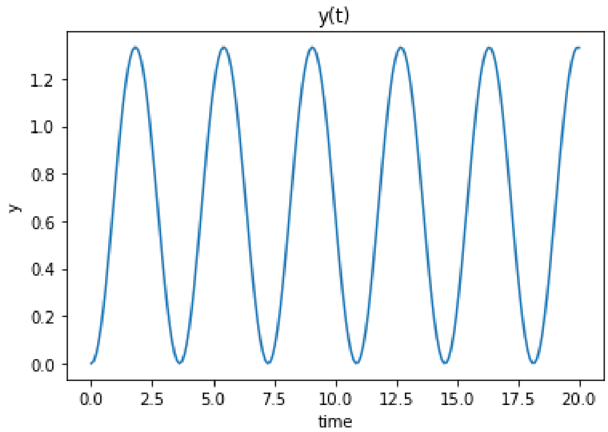
```
Xnum = np.poly1d([1,0,2,0])
Xden = np.poly1d([1,0,3,0,0])
X = sig.lti(Xnum,Xden)
time,x = sig.impulse(X,None,np.linspace(0,20,200))
plt.plot(time,x)
plt.show()
Ynum = np.polymul(Xnum,[1,0,1]) - np.polymul([1,0],Xden)
Y = sig.lti(Ynum,Xden)
time,y = sig.impulse(Y,None,np.linspace(0,20,200))
plt.plot(time,y)
plt.show()
```

Under the initial conditions, x(0) = 1, $\dot{x}(0) = y(0) = \dot{y}(0) = 0$, We get the transfer functions X(s) and Y(s) as

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

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





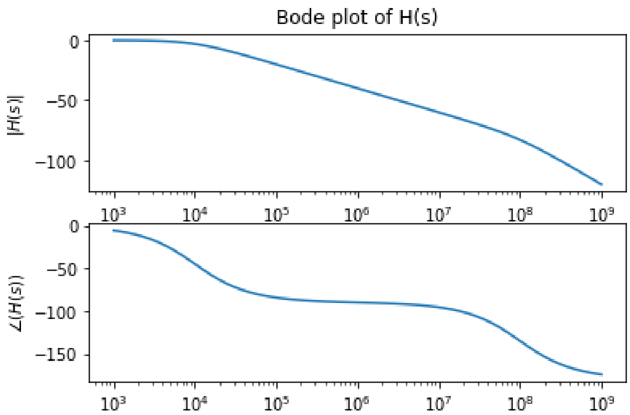
Task 5

For the given RLC circuit, the steady state transfer function is obtained as

$$H(s) = \frac{10^6}{s^2 + 100s10^6}$$

```
H = sig.lti([1000000],[0.000001,100,1000000])
w,mag,phase = sig.bode(H)
plt.semilogx(w,mag)
plt.show()
plt.semilogx(phase)
plt.show()
```

Plotting the bode plot of the time-domain impulse response, we get



Task 6

Given an input signal

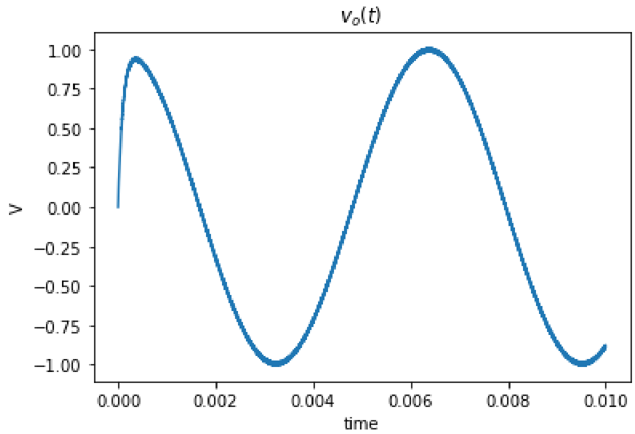
$$v_i(t) = cos(10^3 t)u(t) - cos(10^6 t)u(t)$$

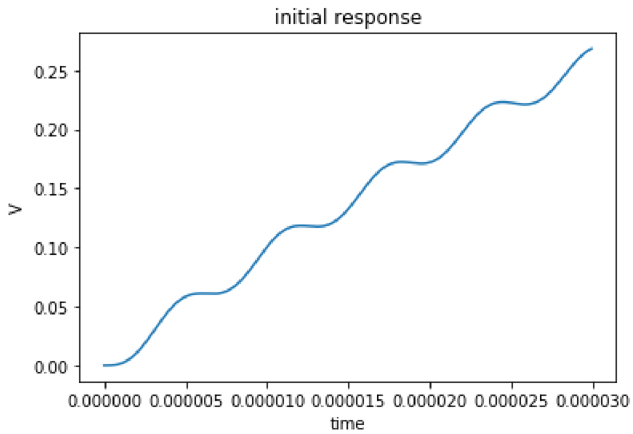
```
time = np.linspace(0,0.01,100000)
x = np.cos(1000*time) - np.cos(1000000*time)
time,y,svec = sig.lsim(H,x,time)
plt.plot(time,y)
plt.show()
plt.plot(time[:150],y[:150])
plt.show()
plt.plot(time[32000:35000],y[32000:35000])
plt.show()
```

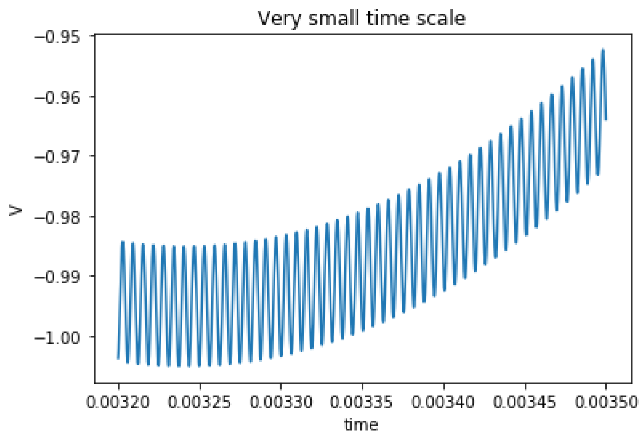
When this input signal is passed to the above RLC circuit, we get the time domain output signal as

Zooming into the 1st 30 microseconds:

The high frequency oscillations are observed as:
The system behaves as a low-pass filter. The lower frequencies get slightly damped whereas the high frequency component gets highly damped to only be seen as a ripple.


$v_o(t)$
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
v
0.000
0.002
0.004
0.006
0.008
0.010
time


initial response
0.25
0.20
0.15
0.10
0.05
0.00
v
0.000000
0.000005
0.000010
0.000015
0.000020
0.000025
0.000030
time


Very small time scale
−0.95
−0.96
−0.97
−0.98
−0.99
−1.00
v
0.00320
0.00325
0.00330
0.00335
0.00340
0.00345
0.00350
time

Conclusion

The scipy.signal library provides a useful toolkit of functions for circuit analysis. The toolkit can be used for the analysis of LTI systems in various domains.

- The forced response of a simple spring body system was obtained over various frequencies of the applied force, and highest amplitude was observed at resonant frequency.
- A coupled spring problem was solved using the sp.impulse function to obtain two sinusoids of the same frequency.
- A two-port network, functioning as a low-pass filter was analysed and the output was obtained for a mixed frequency input.