EE2703 : Applied Programming Lab
Assignment - 6

Jawhar S
EE20B049

March 27, 2022

Sinusoidal Force with Different Decays Applied to a Spring System

Firstly, an exponentially decaying sinusoidal force is applied as an input to a spring system and the physics of the system is governed by a 2^{nd} order differential equation. On taking the Laplace transform of the equation, we get

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

$x(t)$ is obtained through the *sp.impulse()* function, which takes a transfer function and computes the impulse response. Using impulse response works since convolving $x(t)$ with $\delta(t)$ gives $x(t)$ itself and convolution in time domain is multiplication in frequency domain, where the Laplace transform of $\delta(t)$ is 1. So X(s) can be transformed to x(t) using *sp.impulse()*.

Two different inputs were given, with same frequency but with different decay factors. The obtained time response are plotted below.

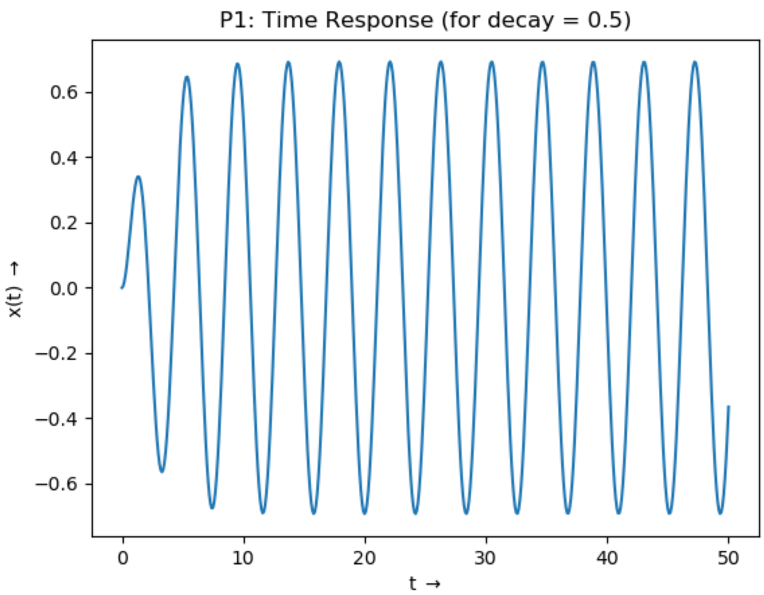

Figure 1: Plot for Problem 1

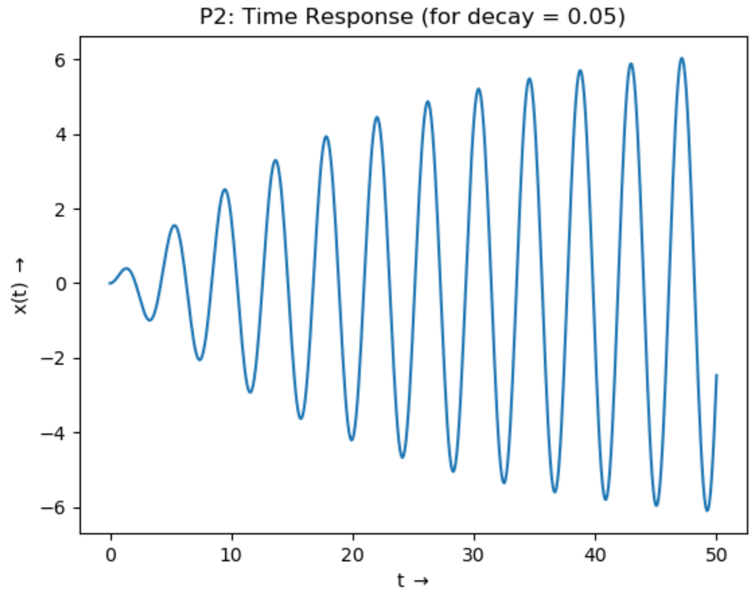

Figure 2: Plot for Problem 2

We notice that the force with smaller decay factor takes longer time to settle to sinusoidal steady state. This is because a smaller decay factor means less decay rate, so the force acts on the system for a longer time. Since it is at resonant frequency (the natural frequency of the system), the force always adds energy to the system. So while the force is active, the amplitude of the sinusoid only increases. This also explains why the final amplitude is higher for low decay factor, since the force is active for more time.

Below is the code for obtaining the time response using *sp.impulse()*

```
num = np.poly1d([1,decay])
denom = np.polymul([1,2*decay,decay**2 + freq**2],[1,0,2.25])
H = sp.lti(num,denom)
t,x = sp.impulse(H,None,np.linspace(0,50,501))
```

Sinusoidal Force with Different Frequencies Applied to a Spring System

In this section, we obtain the transfer function of the system as

$$H(s) = \frac{1}{s^2 + 2.25}$$

Now, we use *sp.lsim()* and obtain $x(t)$ by simulating the convolution of the impulse response of the system and the force applied. The time response obtained for same decay rate but with different frequencies are plotted below.

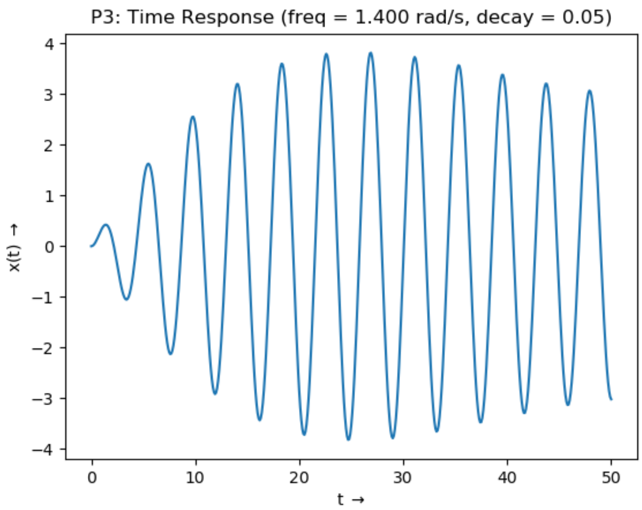

Figure 3: Plot for Problem 3(a)

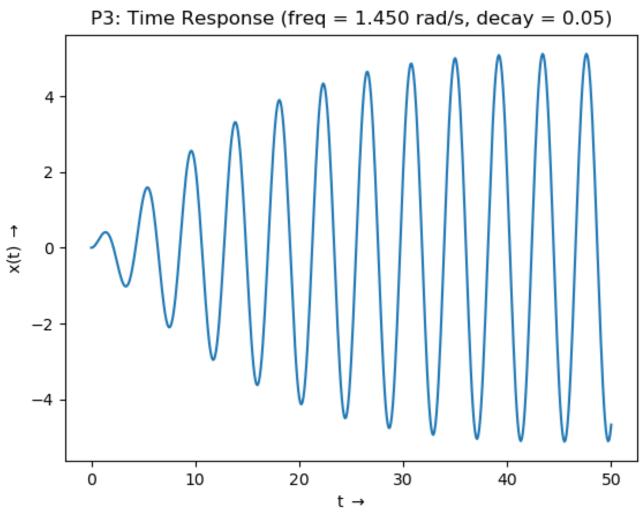

Figure 4: Plot for Problem 3(b)

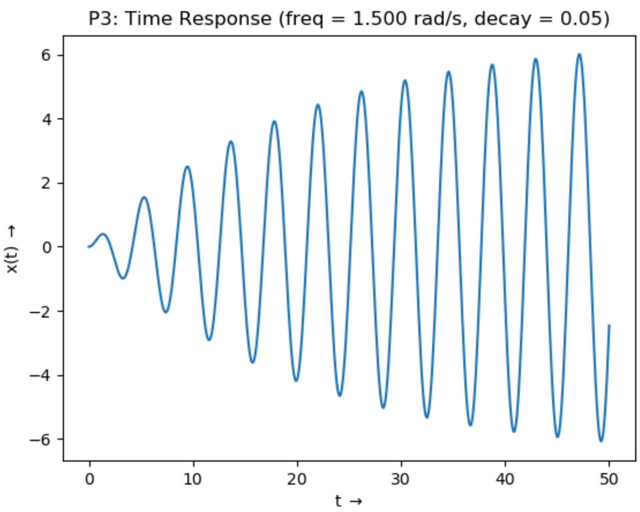

Figure 5: Plot for Problem 3(c)

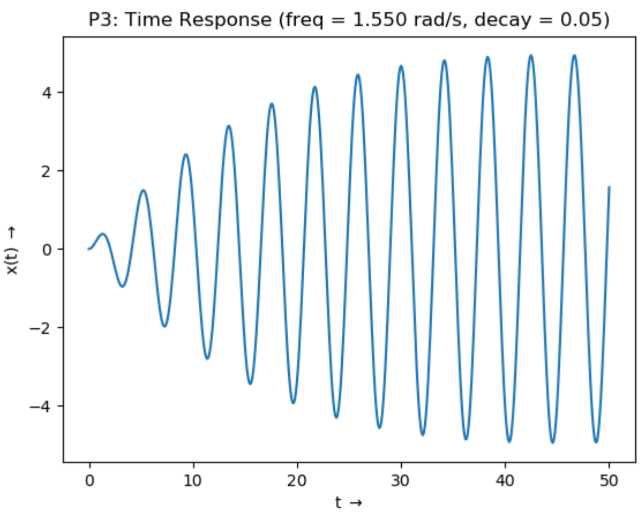

Figure 6: Plot for Problem 3(d)

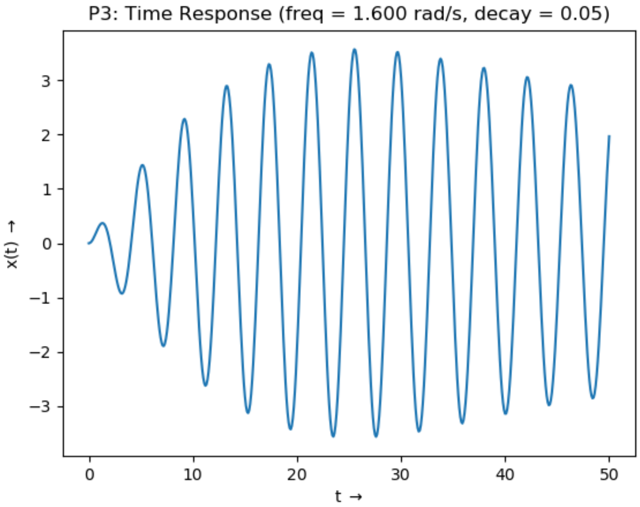

Figure 7: Plot for Problem 3(e)

As we can observe from the plots, at both lower and higher frequencies, the amplitude of the sinusoid reaches a peak, then falls. This is because the input frequency does not match the natural frequency. So instead of the resonant case where the force always increased energy, in this case the force will in fact become out of phase with the natural response of the system, hence reducing the energy, which is seen by the dip in amplitude. As the frequencies come closer to the natural frequency, the dipping reduces.

Below is the code for obtaining the time response using *sp.lsim()*

```
H = sp.lti([1],[1,0,2.25])
t = np.linspace(0,50,501)
for freq in np.linspace(1.4,1.6,5):
    f = np.cos(freq*t)*np.exp(-decay*t)
    t,x,svec = sp.lsim(H,f,t)
```

Solution to the Coupled Spring System

On taking Laplace transform of the given equations and solving with the given initial conditions, we get,

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

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

Using sp.impulse(), we get x(t) and y(t). The code for the same is given below.

```
num = [1,0,2]
denom = [1,0,3,0]
H = sp.lti(num,denom)
t = np.linspace(0,20,201)
t,x = sp.impulse(H, None, t)

num = [2]
denom = [1,0,3,0]
H = sp.lti(num,denom)
t = np.linspace(0,20,201)
t,y = sp.impulse(H, None, t)
```

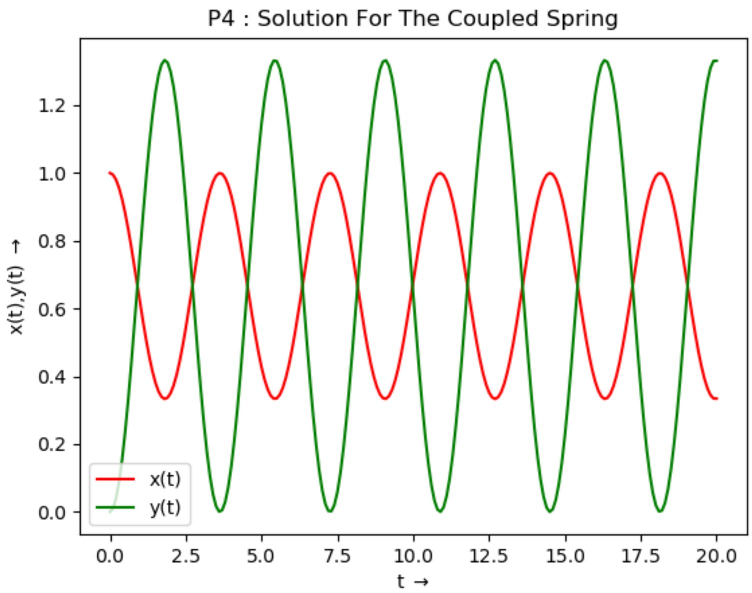

Figure 8: Plot for Problem 4

Obtaining Bode Plots for the given RLC Filter

Firstly, the transfer function was found as,

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

The obtained magnitude and phase Bode plots are given below.

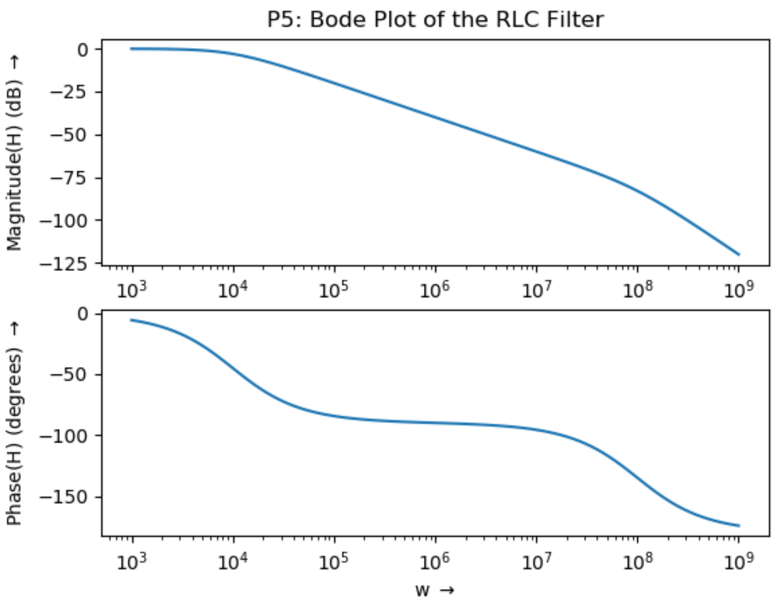

Figure 9: Plot for Problem 5

On factorising the denominator of $H(s)$, we get 2 poles, approximately at $\omega = 10^8, 10^4$. This is reflected in the Bode plots as the slope of magnitude becomes more negative around these points. Since the frequency terms in the denominator control the phase, higher $j\omega$ in denominator causes phase to decrease while passing a pole.

The code for plotting the Bode plot is given below.

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

Obtaining Output Voltage for the given Input Voltage in the RLC Filter

The input voltage applied on the system is given to be,

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

The output response is observed on the millisecond and microsecond timescales to notice the short and long term behaviour. The code used for the same is given below.

```
n = int(1e5)
t = np.linspace(0,1e-2, n+1)
f = np.cos(1e3 * t) - np.cos(1e6 * t)
t,v_o,svec = sp.lsim(H,f,t)
```

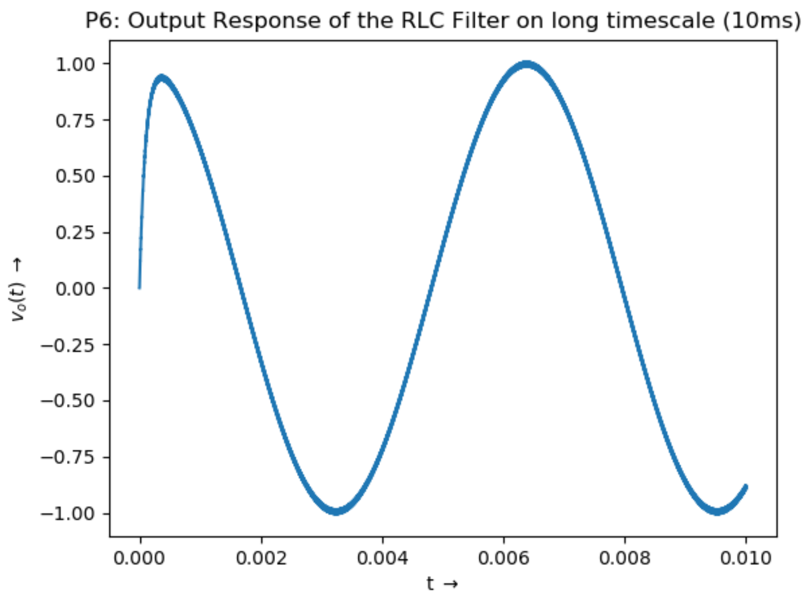

Figure 10: Plot for Problem 6(a)

The lower frequency sinusoid has mostly been preserved while the higher frequency sinusoid has very small effect. This is possibly due to the magnitude response at 10^3 being 0 dB, at 10^6 being around -50 dB (between 100 to 1000 times attenuated).

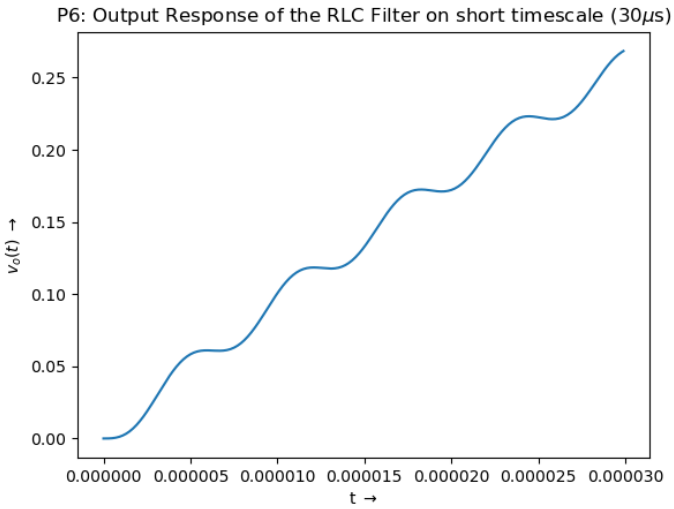

Figure 11: Plot for Problem 6(b)

Conclusion

This assignment explored the analysis of LTI systems using Scipy in Python. Briefly, the following inferences were made from each of the given problems.

- P1,P2: A lower decay factor causes slower settling time and the amplitude increases continuously if the input sinusoid frequency matches the natural frequency of the system.
- P3: The force goes out of phase with the natural response at some point of time if the two frequencies don't match. Once this happens, the force reduces energy and the amplitude dips. Moving the forcing frequency closer to natural frequency reduces the dipping effect.
- P4: The equations governing a coupled system can be decoupled and transformed into Laplace domain by carefully applying the initial conditions.
- P5,P6: From the magnitude response, we see that the high frequency is attenuated while the low frequency passes. This is seen in the rippling of the small timescale plot where the high frequency ripple amplitude is small. The large timescale plot looks sinusoidal, but the first peak occurs a bit early. This is because of transients in the initial few time periods.