# Assignment No 6

Sumanth R Hegde, EE17B032

March 12, 2019

## Aim

The aim of this assignment is to explore the use of Python libraries in analysing Linear Time-Invariant Systems.

## Time response of a spring system

The goal is to obtain the time response of a spring system given by the equation

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

where

$$f(t) = cos(1.5t)e^{-0.5t}u(t) \tag{1}$$

On solving in the Laplace domain, we get

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

Using the given initial conditions, for $0 < t < 50s$ we get the following graph :

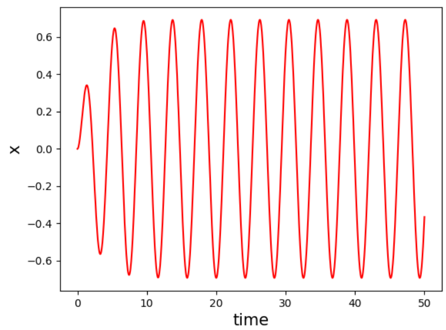


Figure 1: Time response of a spring

For a smaller decay of 0.05 , we get the following plot :

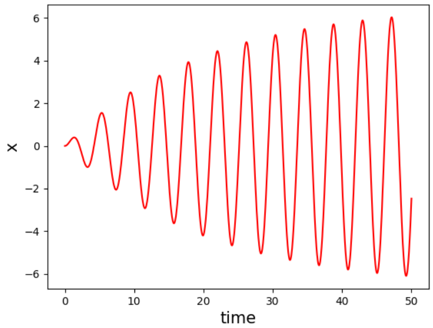


Figure 2: Time response of a spring with smaller decay

## Response over different frequencies

Modelling the system as an LTI system,the following graphs are obtained by varying the frequency of the force $f(t)$

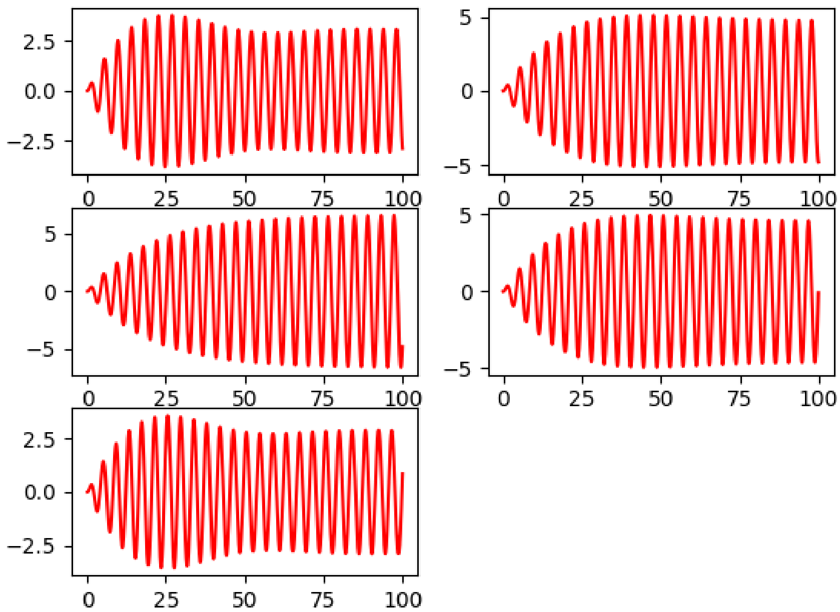


Figure 3: Time response with varying frequency of the applied force

From the given equation, we can see that the natural response of the system has the frequency $w = 1.5$ rad/s . Thus, as expected the maximum amplitude of oscillation is obtained when the frequency of $f(t)$ is 1.5 rad/s , as a case of resonance.

## The coupled spring problem

The coupled equations :

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

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

On Solving the coupled equations, we get the fourth order equation

$$\ddddot{x} + 3\ddot{x} = 0$$

The above equation can be solved for in the Laplace domain. Using the sp.impulse function, we can obtain $x(t)$ and $y(t)$ . We obtain

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

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

The following plots are obtained for $x(t)$ and $y(t)$ over the interval $0 < t < 20s$ :

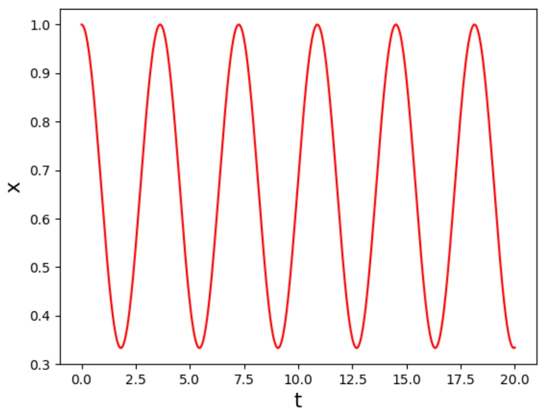


Figure 4: Solution for x(t)

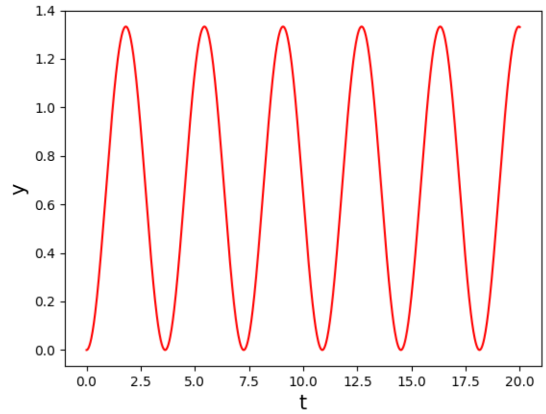


Figure 5: Solution for y(t)

We notice that $x(t)$ and $y(t)$ are sinusoids of the same frequency, but with different amplitude and phase.

## The Two-Port Network

The Steady-State transfer function of the given circuit is given by

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

The magnitude and phase response are obtained using the `signal.bode()` attribute :

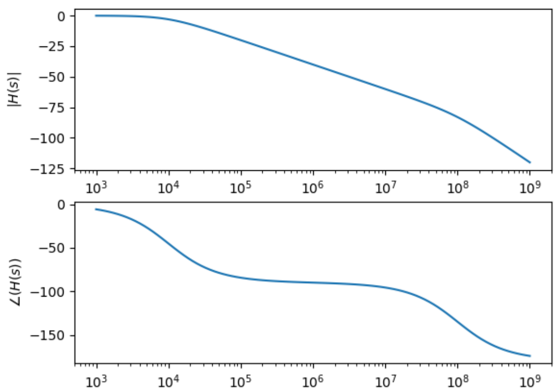


Figure 6: Magnitude and phase plot of $H(s)$

Now, when the input to this system is

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

The output is given by

$$V_o(s) = V_i(s)H(s)$$

On solving using the sp.lsim function, we get the following plot, for $0 < t < 10\mu s$ :

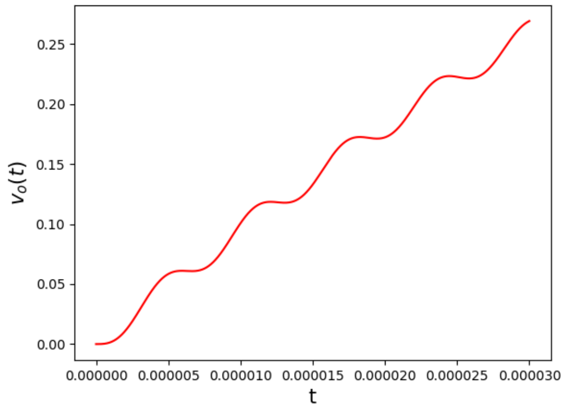


Figure 7: $v_o(t)$ for $0 < t < 10\mu s$

These variations are determined by the high frequency component of $v(t)$

.

On plotting from $0 < t < 10ms$ :

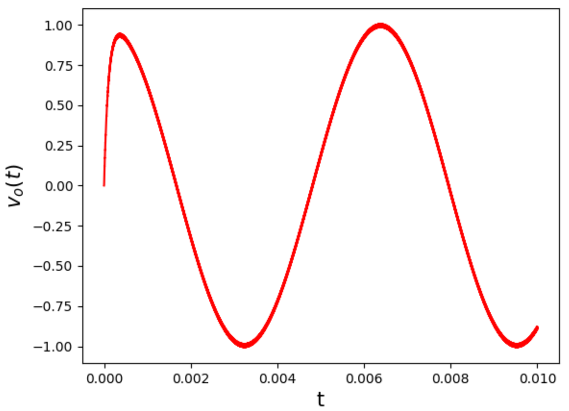


Figure 8: $v_o(t)$ for $0 < t < 10ms$

From the Bode plot of $H(s)$, we notice that the system provides unity gain for a low frequency of $10^3$ rad/s . Thus, the low frequency component is more or less preserved in the output. However, the system dampens a high frequency of $10^6$ rad/s, with $|H(s)|_{dB} \approx -40$ . This is because of the inherent nature of the given circuit to act as a low pass filter. Thus, magnitude of oscillations of these frequency is reduced, which is noticeable on zooming into the plot :

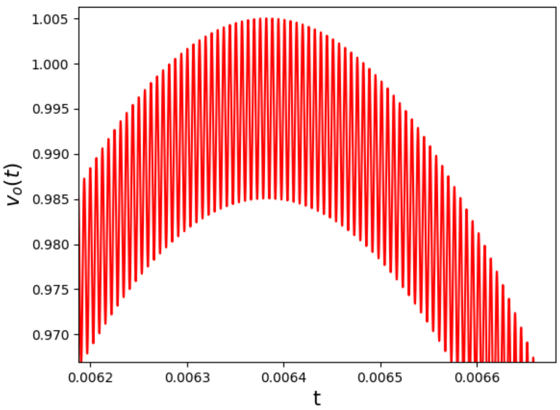


Figure 9: High frequency variations of $v_o(t)$

The peak-to-peak value of these oscillations is $\approx 0.02$ , as expected, as the system dampens such frequencies by about a factor of 100.

## Conclusion

The `scipy.signal` library provides a useful toolkit of functions for circuit analysis. The toolkit was 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.