ECE 351-51

FEBRUARY 22, 2022

Lab 5

Submitted By :
Kennedy Beach

Contents

1   Introduction

The purpose of this lab was to compare the hand-calculated and Python generated time-domain response of an RLC bandpass filter using Laplace transforms. The Sine Method was used to generate the time-domain response in the hand-calculation.

2   Equations

The following transfer function was found from an RLC bandpass filter:

$$H(s) = (s/RC)/(s^2 + s/RC + 1/LC) \tag{1}$$

Equations for the Sine Method:

$$s = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \tag{2}$$

$$p = \alpha + j\omega \tag{3}$$

$$g = \frac{1}{RC} * s \tag{4}$$

where s = p

$$|g| = \sqrt{\alpha^2 + \beta^2} \tag{5}$$

$$< g = tan^-1(\frac{\beta}{\alpha}) \tag{6}$$

Final Value Theorem:

$$\lim_{t \to \infty} f(t) \to \lim_{s \to 0} sF(S) \tag{7}$$

$$\lim_{s \to 0} sF(s) = \lim_{s \to 0} \frac{\frac{s^2}{RC}}{s^2 + \frac{1}{RC}s + \frac{1}{LC}} = 0 \tag{8}$$

3   Methodology

The first part involved changing the transfer function to a time-domain response using the Sine Method. These input into Python is seen below in Listing 1.

```python
import numpy as np
import matplotlib.pyplot as plt
import scipy.signal as sig


def u(t):
    if t < 0:
        return 0
    if t >= 0:
        return 1

def sine_method(R,L,C,t):
    y = np.zeros((len(t),1))
    alpha = -1/(2*R*C)
    omega = 0.5*np.sqrt(1/(R*C))**2 - 4*(1/(np.sqrt(L*C)))**2 + (0
    + 1j)
    p = alpha + omega
    g = 1/(R*C)*p
    mag_g = np.abs(g)
    g_rad = np.angle(g)
    angle_g = g_rad*(180/np.pi)

    for i in range(len(t)):
        y[i] = ((mag_g/np.abs(omega))*np.exp(alpha*t[i])*np.sin(np.
    abs(omega)*t[i] + g_rad))*u(t[i])
    return y
```

Listing 1: Hand-calculated response of the transfer function using the Sine Method

The impulse response using Python was generated next as seen below. A numerator and denominator array had to be made in order to be used in the sig.impulse function.

```python
steps = 1e-5
t = np.arange(0, 1.2e-3 + steps, steps)

R = 1e3
L = 27e-3
C = 100e-9
X = 1/(R*C)
Y = 1/(L*C)

num = [X, 0]
den = [1, X, Y]
```

```
12
13
14 h_hand = sine_method(R, L, C, t)
15 tout, yout = sig.impulse((num, den), T = t)
16
17 plt.figure(figsize=(20, 6))
18
19 plt.subplot(1, 2, 1)
20 plt.plot(t, h_hand)
21 plt.grid(True)
22 plt.ylabel('y(t)')
23 plt.xlabel('time')
24 plt.title('h1(t)')
25
26 plt.subplot(1, 2, 2)
27 plt.plot(tout, yout)
28 plt.grid(True)
29 plt.ylabel('y(t)')
30 plt.xlabel('time')
31 plt.title('sig.impulse')
```

Listing 2: sig.impulse function

The sig.step function was used to get the step response of the the transfer function. It used the same parameters as the sig.impulse function.

```
1 tout, yout = sig.step((num,den), T = t)
```

Listing 3: sig.step function

4   Results

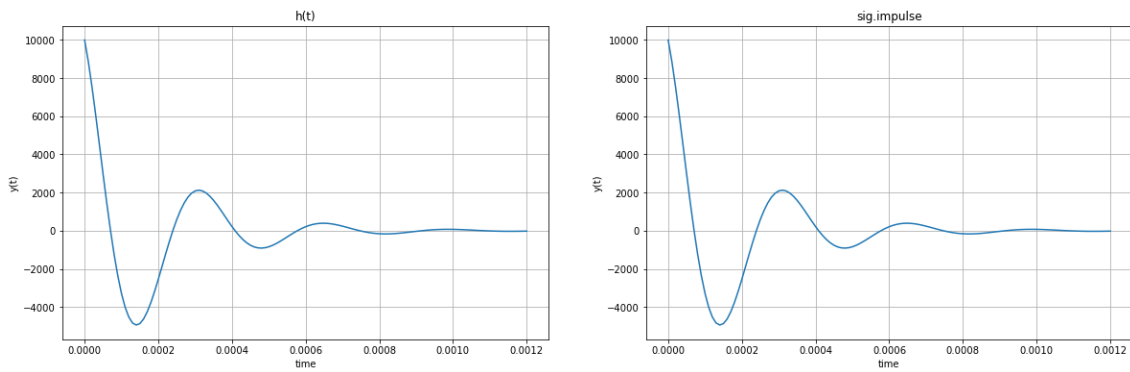The plots for the impulse and step responses are seen below.
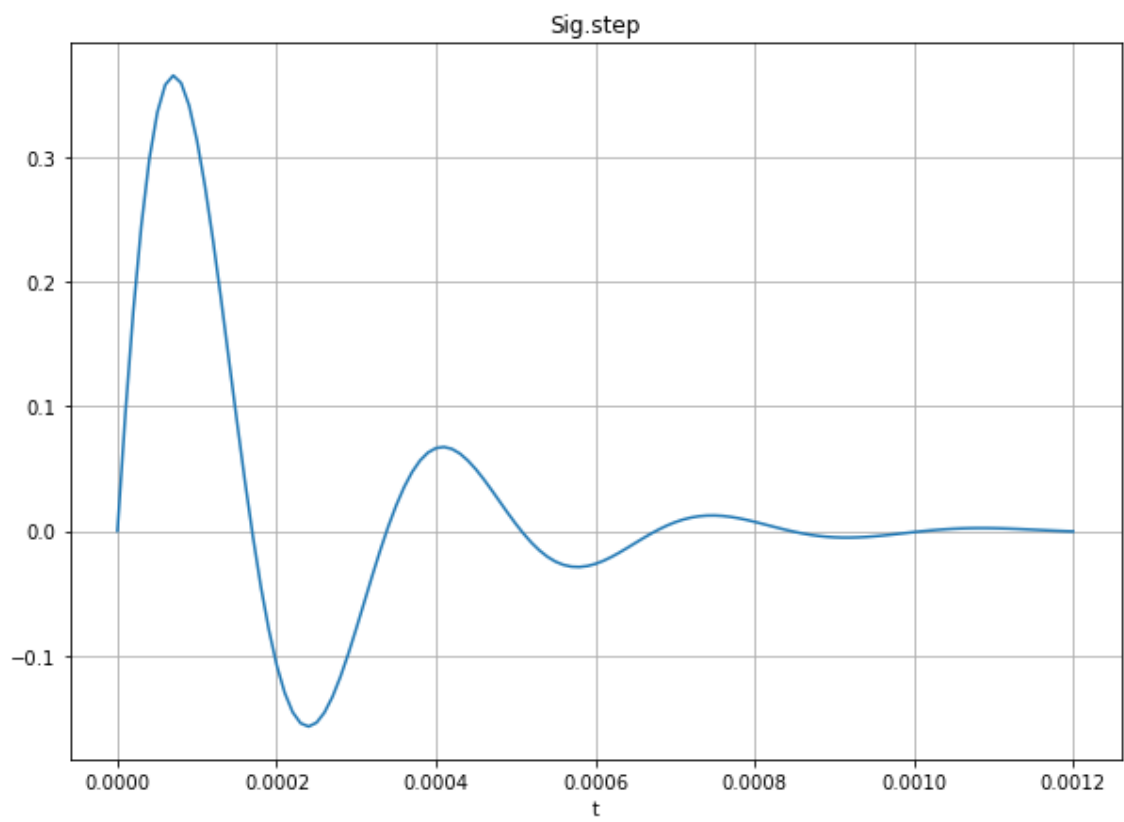
Figure 1: Plots of the impulse responses

Figure 2: Step response

5   Error Analysis

I ran into some problems getting the hand calculated impulse response to look like the Python generated one. When I looked over my Sine Method function, I found out the math performed in the omega variable was wrong. Once I fixed it, the impulse plots looked identical.

6   Questions

1. Compare your step response result to the plot in Part 1 Task 2 and discuss whether your result makes sense.

The step response starts from zero while the impulse response starts from 10,000. This makes sense since the step function goes from zero to one at time zero, then stays at one. When the step function was convolved with the impulse response, the step response acted like a step function initially.

2. Explain the result of the Final Value Theorem from Part 2 Task 2 in terms of the physical circuit components.

The Final Value Theorem for this circuit equals zero as is seen when the amplitude of the wave tapers off on the x-axis. The capacitor and inductor charge, then discharge. The final state of the components is zero, and the circuit will be off as the energy will be lost through the resistor.

3. Leave any feedback on the clarity of the expectations, instructions, and deliverables.

Everything was clear on what needed to be done and turned in.

7   Conclusion

The Sine Method of transforming a transfer function into a time-domain impulse response was explored. The work was the checked with Python and step response of the same transfer function was then created. The Python and LaTeX code are seen in `https://github.com/Eniac618/ECE351_Code` and `https://github.com/Eniac618/ECE351_Reports` respectively.