EE2703 : Applied Programming Lab
Experiment 6

Ch Venkat Sai
EE18B042

March 07, 2020

Overview

The experiment is about:
Learning how to use tools from the signal toolbox.
Using laplace domain in finding out the output from a differential equation and getting transfer functions from circuits and using inverse laplace to finally obtain the output.
Analysing the output behaviour and relating them to the input.

Theory

If the laplace transform of the input f is F(s) and the laplace transform of the impulse response h(t) is H(s) then the laplace transform of the output is H(s)F(s). The transform of any function can be found out using the formula:

$$H(s) = \int_{-\infty}^{\infty} h(t)e^{-st}dt \tag{1}$$

Though some of the functions have classic laplace transforms and corresponding inverses.

Q1

The laplace transform of the function

$$f(t) = \cos{(1.5t)}e^{-0.5t}u_0(t) \tag{2}$$

is given by

$$F(s) = \frac{s+0.5}{(s+0.5)^2+2.25} \tag{3}$$

Now solving for x knowing that a double differential in time domain with zero initial conditions gives $s^2X(s)$ we get the laplace transform of x as

$$X(s) = \frac{s+0.5}{(s^2+s+2.5)(s^2+2.25)} \tag{4}$$

which we plot it with the function as shown in the code. The sp.impulse gives plots in time domain with given time range.

```
p = poly1d([1,1,2.5])
q = poly1d([1,0,2.25])
w = polymul(p,q)
```

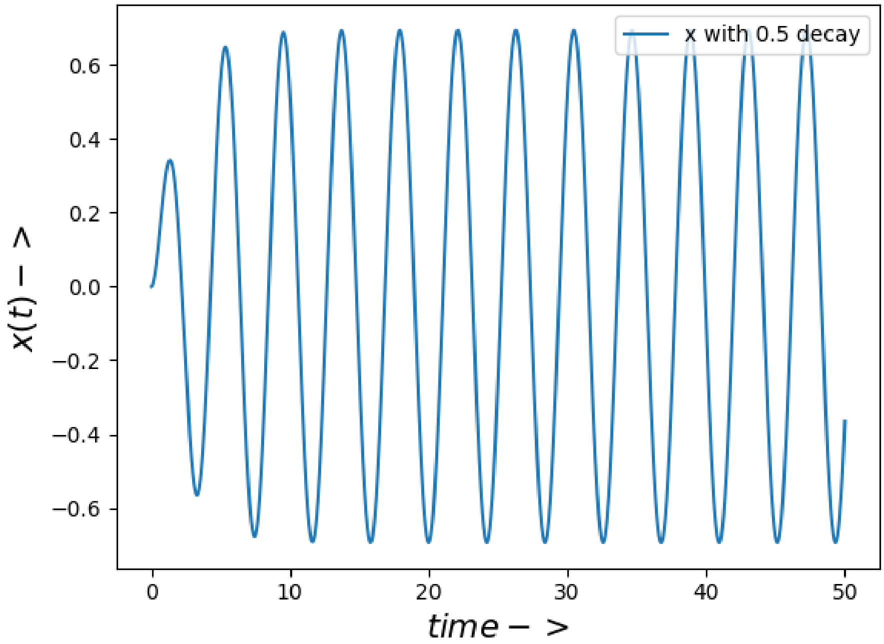

Figure 1: x(t) wrt t with decay=0.5

```
X = sp.lti([1,0.5],w)
t, x = sp.impulse(X, None, linspace(0, 50, 2000))
plot(t, x)
show()
```

The plot is as shown in figure.

Q2

Now the same problem with a smaller decay i.e 0.05.

```
p_1 = poly1d([1,0.1,2.2525])
w_1 = polymul(p_1,q)
X_1 = sp.lti([1,0.5],w_1)
t, x_1 = sp.impulse(X_1, None, linspace(0, 50, 2000))
plot(t, x_1)
show()
```

The corresponding plot is as shown in fig:2.

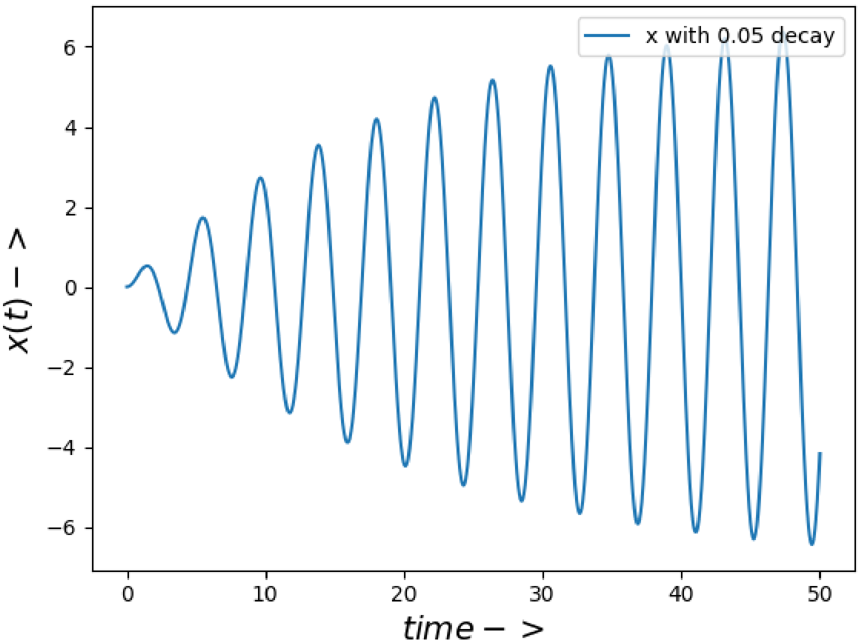

Figure 2: x(t) wrt t with decay=0.05

Q3

Now in the previous problems we had F(s) and we obtained X(s). If the system was LTI then $\frac{X(s)}{F(s)}$ gives us H(s) i.e the frequency response. Now we vary the frequency from 1.4 to 1.6 of f(t) and plot the output using sp.lsim as shown.

```
H = sp.lti([1], [1,0,2.25])
def f(t, w):
    f = cos(w*t)*exp(-0.05*t)
    return f

w = 1.4
for k in range(1, 5):
    t, y, svec = sp.lsim(H,f(t, w),linspace(0, 50, 2000))
    w += 0.05
    plot(t, y)
show()
```

This gives us a plot as shown in fig:3.

Q4

Now we have two coupled differential equations with given initial conditions. Solving them in the laplace domain presents us with these two rational functions.

$$X(s) = \frac{s^2+2}{s^3+3s} \tag{5}$$

$$Y(s) = \frac{2}{s^3+3s} \tag{6}$$

Now we plot their time-domain plots as earlier with sp.impulse.

```
Y = sp.lti([2],[1,0,3,0])
t, y = sp.impulse(Y, None, linspace(0, 20, 2000))
plot(t, y)
X = sp.lti([1,0,2],[1,0,3,0])
t, x = sp.impulse(X, None, linspace(0, 20, 2000))
plot(t, x)
show()
```

The graph turns out be as shown in fig:4.

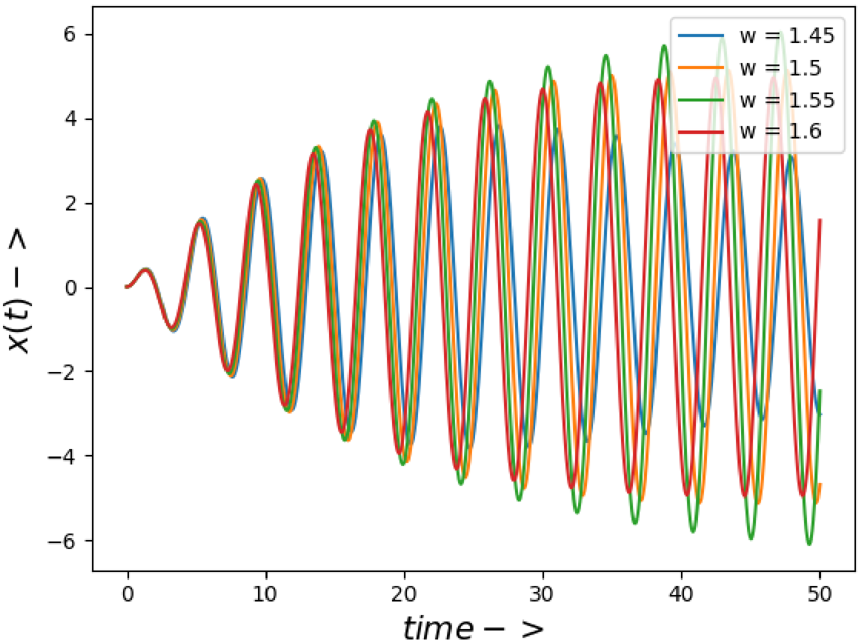

Figure 3: x(t) with frequency 1.4-1.6

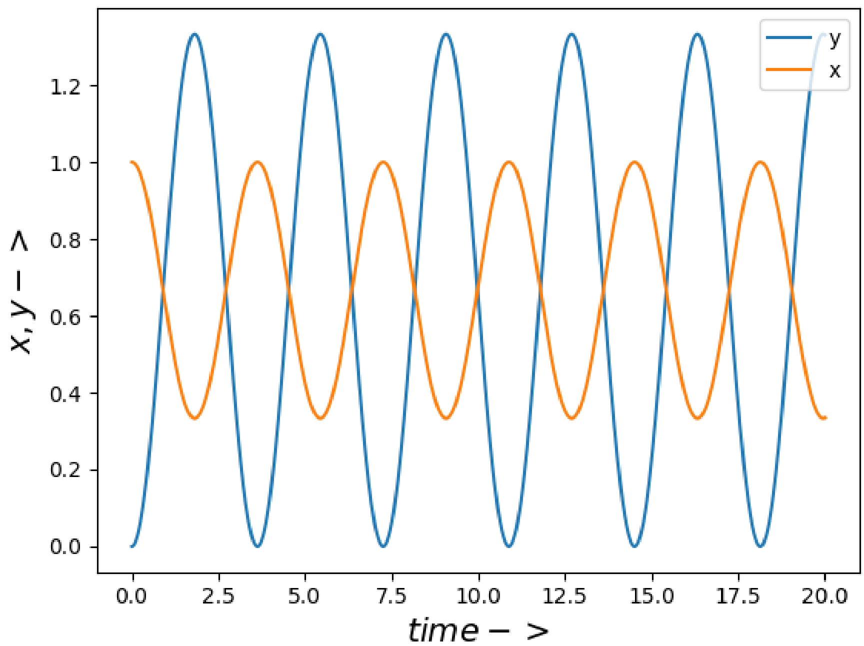

Figure 4: x(t) and y(t)

Q5

Now from the given diagram we obtain the s-domain frequency response and correspondingly plot the bode plots. The s-domain frequency response turns out to be

$$H(s) = \frac{1}{10^{-12}s^2 + 10^{-4}s + 1} \tag{7}$$

```
H = sp.lti([1],[10**(-12), 10**(-4), 1])
w, S, phi = H.bode()
subplot(2,1,1)
semilogx(w,S)
subplot(2,1,2)
semilogx(w,phi)
show()
```

The function semilogx turns the w-scale to log axis. And the y-scale already comes in log scale from H.bode. The function subplot helps us plot two graphs with same x-axis in a single graph with their relative position given in as argument. The plots are as shown in fig:5.

Q6

Now we have a input function defined as:

$$v_(t) = \cos{(10^3 t)}u(t) - \cos{(10^6 t)}u(t) \tag{8}$$

And if this is feeded to the circuit above in laplace domain with initial conditions given to be 0 we get the output signal in the laplace domain. We then use signal.lsim to plot get the time domain output.

```
def v(t):
    v = cos((10**3)*t) - cos((10**6)*t)
    return v

t = linspace(0, 10**(-2), 10**(5))
t1, y, svec = sp.lsim(H, v(t), t)
plot(t, y)
show()
```

We need to capture the fast variation, which means our time step should be smaller than 10^{-6}. Yet we need to see the slow time, which means we must

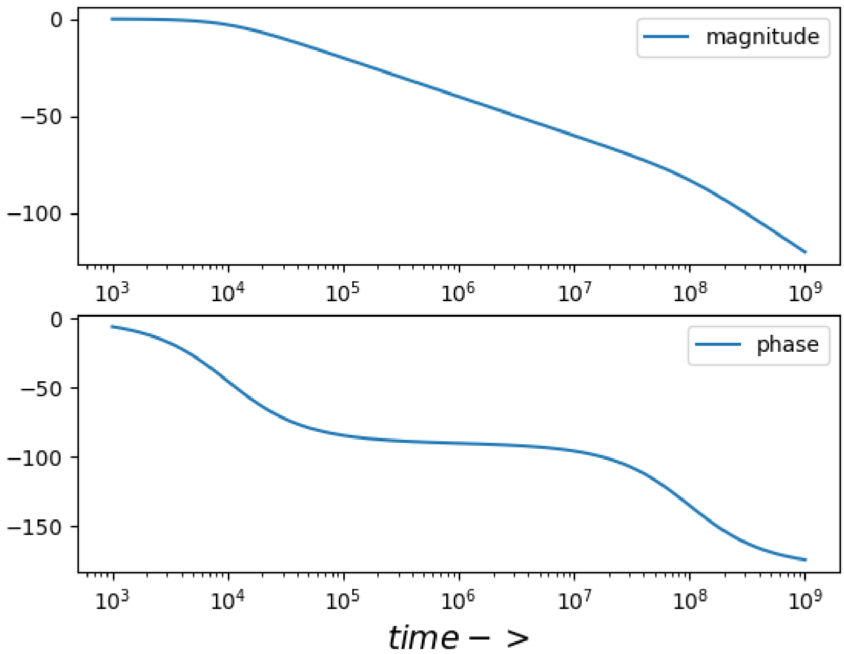

Figure 5: Bode plot mag. and phase

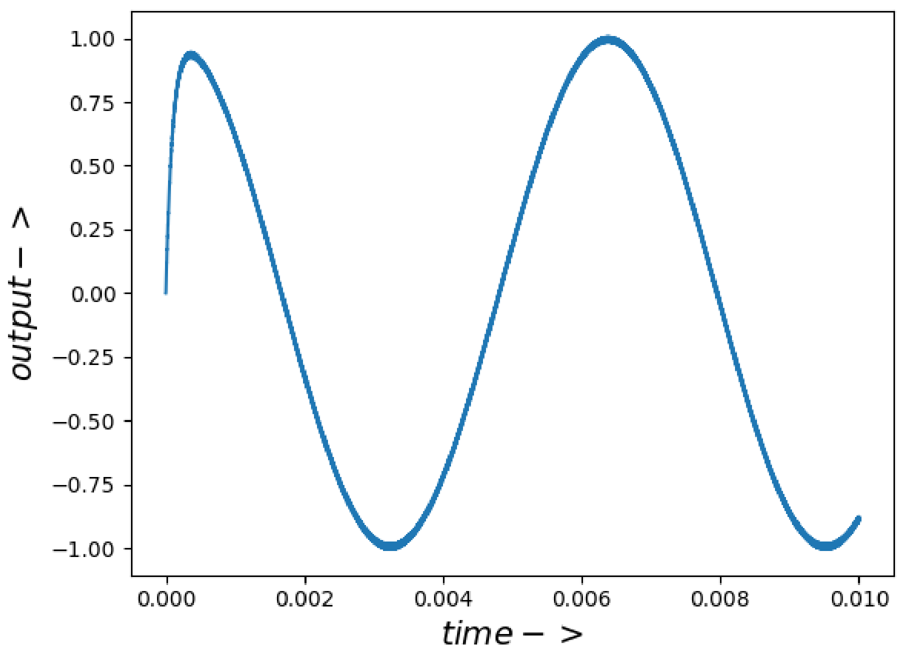

Figure 6: Output y(t)

simulate till about 10 msec. Hence the time vector. We now obtain the plot as shown in fig:6. For a time $t < 30\mu$ the long term response is due to the natural response which dies down slowly i.e amplitude decreases. So as time passes by we have majorly the forced response left and hence the plot turns to be sinusoidal.

Conclusion

At the end we were able to use the tools from the toolbox. And we were able to solve circuits in as well as double differential equation in laplace domain. We were also able to plot bode plots.