Assignment No 7: Analysis of Circuits using Laplace Transforms

EE18B120: Sarvani Cheruvu

March 24, 2020

1 Introduction

Two powerful capabilities of python, i.e. symbolic algebra and circuit analysis by Laplace transform, are focussed on in this assignment via two different circuits.

2 The Assignment

2.1 Theory

The equations are written in matrix form. The solution is obtained using

$$V = A.inv() * b.$$

Output voltage is the fourth element of the vector V.
To obtain the magnitude of impulse response,

```
def magresponse(H, title):
        ww=np.logspace(0,8,801) #frequency
        ss=1j*ww #jw
        hf=lambdify(s,H,'numpy')
        v=hf(ss) #computes the sympy function at every jw
        plt.title(title)
        plt.loglog(ww,abs(v),lw=2)
```

```
    plt.grid(True)
    plt.show()
```

To obtain the response to a given input function, the sympy function has to be converted to an LTI function first.

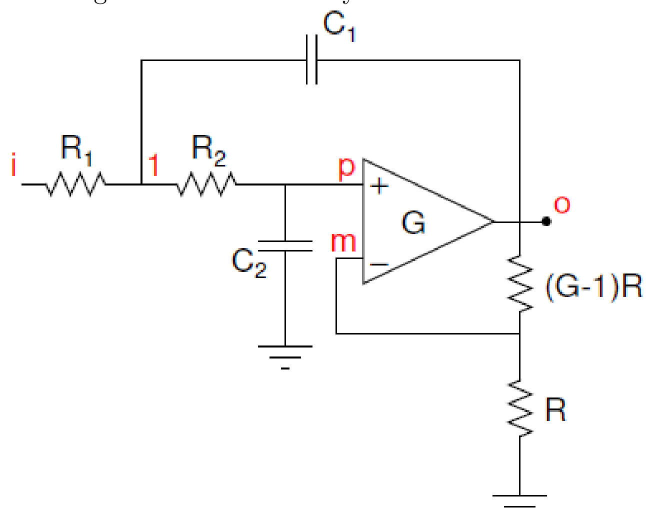
```
def sympytolti(Y):
    Y = expand(simplify(Y))
    num_p, den_p = fraction(Y)
    num_p, den_p = Poly(num_p, s), Poly(den_p, s)
    num, den = num_p.all_coeffs(), den_p.all_coeffs()
    num, den = [float(f) for f in num], [float(f) for f in den]
    return sp.lti(num, den)
```

The output response to a given input function is obtained as follows.

```
def output(lti, input_v, t, title):
        t, vo, svec=sp.lsim(lti, input_v, t)
        plt.title(title)
        plt.plot(t, vo)
        plt.grid(True)
        plt.show()
```

2.2 Low pass filter

The following circuit is to be analysed.



The equations can be written in a matrix form as follows.

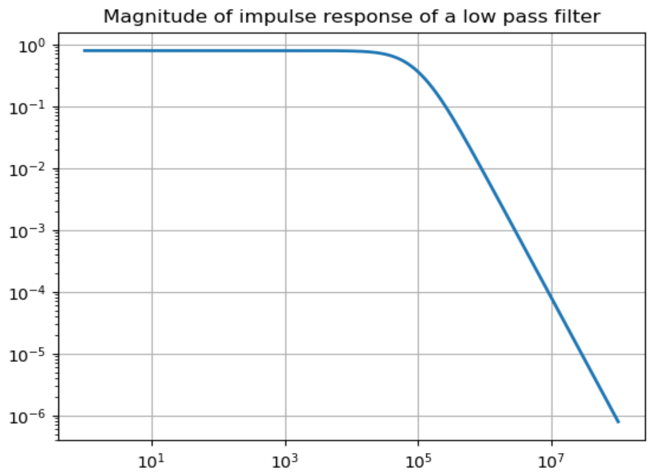
$$\begin{pmatrix} 0 & 0 & 1 & \frac{-1}{G} \\ \frac{-1}{1+sR_2C_2} & 1 & 0 & 0 \\ 0 & -G & G & 1 \\ -\frac{1}{R_1}-\frac{1}{R_2}-sC_1 & \frac{1}{R_2} & 0 & sC_1 \end{pmatrix} \begin{pmatrix} V_1 \\ V_p \\ V_m \\ V_o \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{V_i(s)}{R_1} \end{pmatrix}$$

The filter can be coded as follows:

```
def lowpass(Vi=1,R1=1e4,R2=1e4,C1=1e-9,C2=1e-9,G=1.586):
        A=Matrix([[0,0,1,-1/G],[-1/(1+s*R2*C2),1,0,0],[0,-G,G,-1],[-1/R1-1/R2-s*
        b=Matrix([0,0,0,-Vi/R1])
        V=A.inv()*b
        return V
```

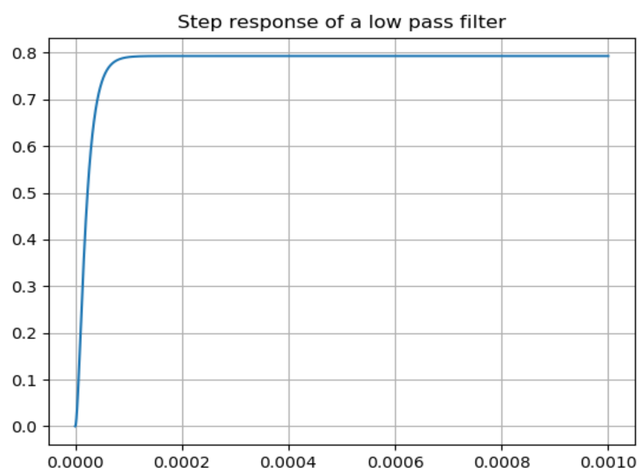
2.2.1 Magnitude of impulse response

```
magresponse(Vl,'Magnitude of impulse response of a low pass filter')
```



2.2.2 Step response

```
output(Hl,(t>0),t,'Step response of a low pass filter')
```

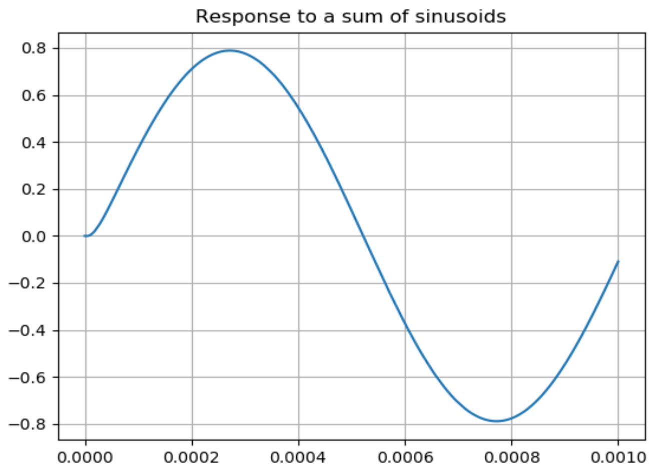


2.2.3 Output to a sum of sinusoids

We want to determine the output if the input function is:

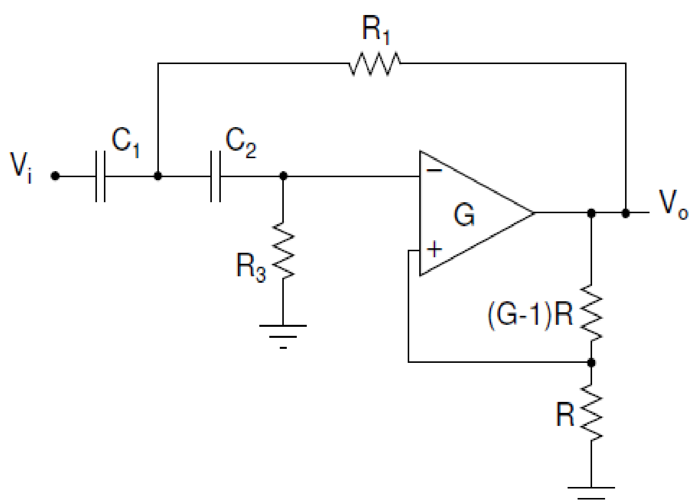
$$v_i(t) = \sin(2000 * \pi * t) + cos(2 * 10^6 * \pi * t)$$

```
output(Hl,((np.sin(2000*np.pi*t)+np.cos(2e6*np.pi*t)))*(t>0),t,'Response to a sur
```

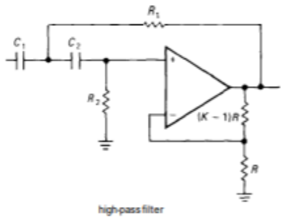


2.3 High pass filter

The following circuit is to be analysed.



However, the signs of the gain block have been switched. This is the correct circuit (from pg. 274 of Horowitz and Hill).



Accordingly, the equations can be written in a matrix form as follows.

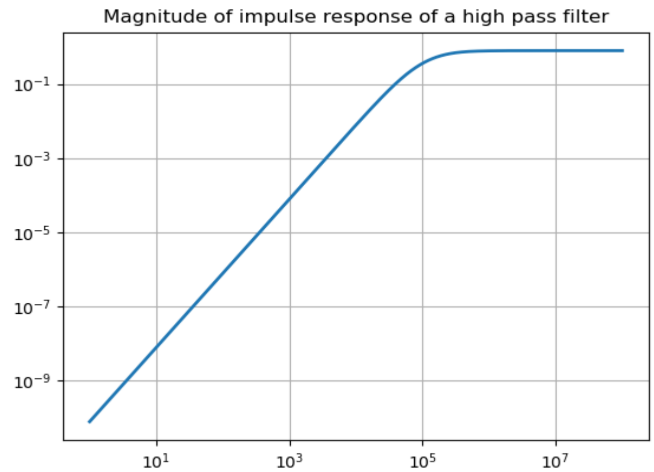
$$\begin{pmatrix} 0 & 0 & 1 & \frac{-1}{G} \\ \frac{-sC_2R_3}{1+sC_2R_3} & 1 & 0 & 0 \\ 0 & G & -G & 1 \\ -sC_1 - \frac{1}{R_1} - sC_2 & sC_2 & 0 & \frac{1}{R_2} \end{pmatrix} \begin{pmatrix} V_1 \\ V_p \\ V_m \\ V_o \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{V_i(s)}{sC_1} \end{pmatrix}$$

The filter can be coded as follows:

```
def highpass(Vi=1,C1=1e-9,C2=1e-9,R1=1e4,R3=1e4,G=1.586):
        A=Matrix([[0,0,1,-1/G],[-(s*C2*R3)/(1+s*C2*R3),1,0,0],[0,G,-G,-1],[-(s*C
        b=Matrix([0,0,0,-Vi*s*C1])
        V=A.inv()*b
        return V
```

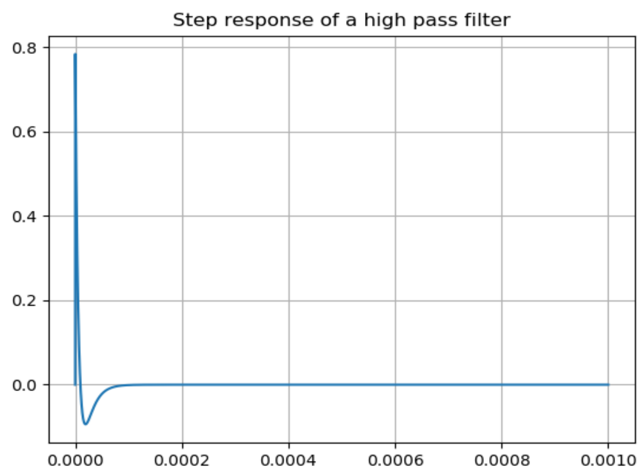
2.3.1 Magnitude of impulse response

```
magresponse(Vh,'Magnitude of impulse response of a high pass filter')
```



2.3.2 Step response

```
output(Hh,(t>0),t,'Step response of a high pass filter')
```



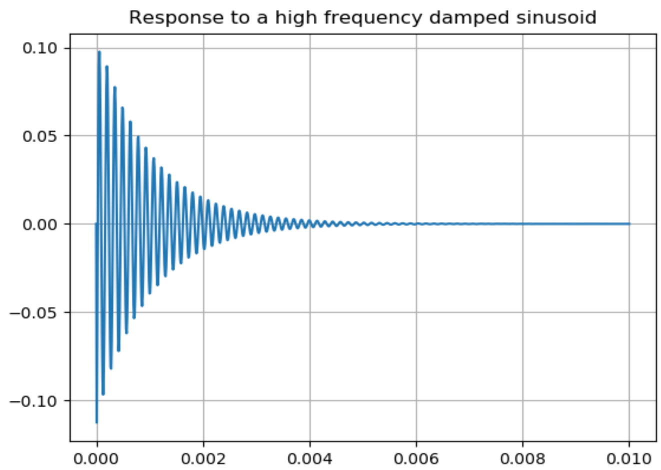
2.3.3 Output to a high frequency damped sinusoid

We want to determine the output if the input function is:

$$v_i(t) = \sin(10^7 * t)\exp\{-10^3 * t\} * u(t)$$

```
t=np.linspace(0,1e-2,1000)
```

```
output(Hh,(np.sin(1e7*t)*np.exp(-1e3*t))*(t>0),t,'Response to a high frequency d
```

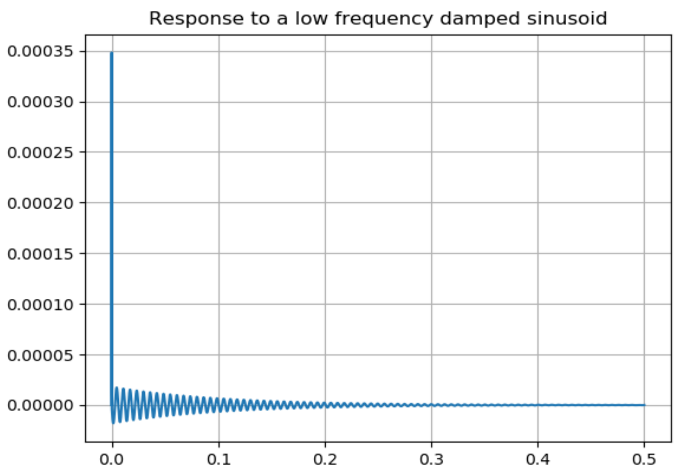


2.3.4 Output to a low frequency damped sinusoid

We want to determine the output if the input function is:

$$v_i(t) = \sin(10^3 * t)\exp\{-10 * t\} * u(t)$$

```
t=np.linspace(0,0.5,10000)
output(Hh,(np.sin(1e3*t)*np.exp(-1e1*t))*(t>0),t,'Response to a low frequency da
```



3 Conclusion

Upon passing a sum of sinusoids to a low-pass filter, only the high-frequency sinusoid persists. Upon passing damped sinusoids to a high-pass filter, the low-frequency signals die out very quickly, while the high-frequency signal attenuates for some time before dying out.