Fourier Series through Circuits

G V V Sharma*

Contents

***Abstract*—This manual provides a quick introduction to Fourier series and Low Pass Filters (LPF), besides facilitating the use of Python for Signals & Systems.**

1 Fourier Series

Problem 1.1. Type the following program in python to obtain $g(t)$. $g(t)$ is a periodic signal called a square wave with amplitude $A = 5V$ and time period $T = 20ms$.

Solution:

```
import numpy as np
from scipy import signal
import matplotlib.pyplot as plt

#Wave Amplitude
A=5
#Wave Period
T = 0.02;
#Time Samples
t = np.linspace(-2.5*T,2.5*T,1e4)
#Plot wave
plt.plot(t, A/2*(1+signal.square
    (2*np.pi*t/T)))
plt.ylim(-1, 6)
plt.grid()
plt.xlabel('$t$')
plt.ylabel('$g(t)$')
#Save figure
plt.savefig('../figs/1.1.eps')
plt.show()
```

*The author is with the Department of Electrical Engineering, Indian Institute of Technology, Hyderabad 502285 India e-mail: gadepall@iith.ac.in. This work was funded by the PMMMNMTT, MHRD, Govt. of India. All content in this manuscript is released under GNU GPL. Free to use for all.

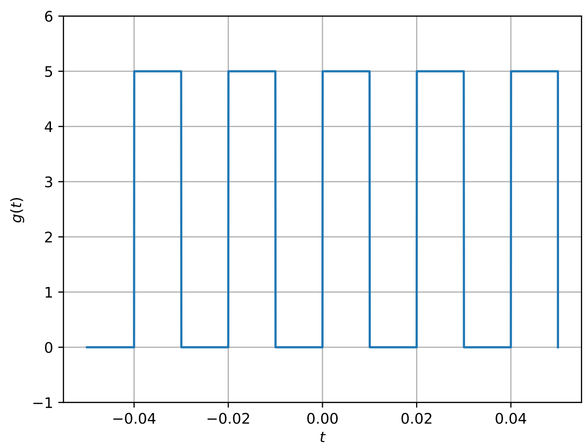

Fig. 1.1: Generating square wave.

Problem 1.2. Find the frequency of $g(t)$.

Solution: The frequency of $g(t)$ is given by $f = \frac{1}{T} = 50$ Hz.

Problem 1.3. The following expression

$$g(t) = \sum_{n=0}^{\infty} a_n \cos 2\pi n f t + b_n \sin 2\pi n f t \tag{1.1}$$

is known as the Fourier series expansion of $g(t)$, where $f = \frac{1}{T}$. Find

$$a_n = \frac{2}{T} \int_0^T g(t) \cos 2\pi n f t \, dt \tag{1.2}$$

$$b_n = \frac{2}{T} \int_0^T g(t) \sin 2\pi n f t \, dt \tag{1.3}$$

Solution:

$$a_0 = \frac{A}{T}\int_0^{T_0} dt \tag{1.4}$$
$$= \frac{AT_0}{T} \tag{1.5}$$

and

$$a_n = \frac{2A}{T}\int_0^{T_0} \cos 2\pi n f t\, dt \tag{1.6}$$
$$= \frac{A}{\pi n f T}\sin 2\pi n f T_0 \tag{1.7}$$

Similarly,

$$b_n = \frac{2A}{T}\int_0^{T_0} \sin 2\pi n f t\, dt \tag{1.8}$$
$$= \frac{A}{\pi n f T}\left[1 - \cos 2\pi n f T_0\right] \tag{1.9}$$

Problem 1.4. Using Python, compute the series

$$\sum_{n=0}^{15} a_n \cos 2\pi n f t + b_n \sin 2\pi n f t \tag{1.10}$$

for $A = 5, T = 20ms$ and a_n, b_n obtained in the previous problem. Comment.

Solution: Type the following program

```
import numpy as np
import matplotlib.pyplot as plt


T_0 = 0.01
T = 0.02
f = 1/T
A = 5
simlen = 1e3

 #generating points in t-axis
t = np.linspace(-2.5*T,2.5*T,1e4)


for n in range(21):
        if n == 0:
                g = A*T_0/T;
        else:
                cost = np.cos(2*np.pi*n*f*t);#Computing cosine term
                sint = np.sin(2*np.pi*n*f*t);#Computing sine term

                an = 2*A*np.sin(2*np.pi*n*f*T_0)/(2*np.pi*n*f*T); #Computing coefficients
                bn = 2*A*(1 - np.cos(2*np.pi*n*f*T_0))/(2*np.pi*n*f*T); #Computing coefficients
                g = g + an*cost + bn*sint; #evaluating the summation

plt.plot(t,g)
plt.grid ()
plt.xlabel('$t$')
plt.ylabel('$g(t)$')
#Save figure
plt.savefig('../figs/1.4.eps')
plt.show()
```

to obtain the following figure. Through this problem, we find that the square wave can be approximated through an infinite sum of sinusoids. The ripples in Fig. 1.4 occur due to convergence issues and is known as the Gibbs phenomenon.

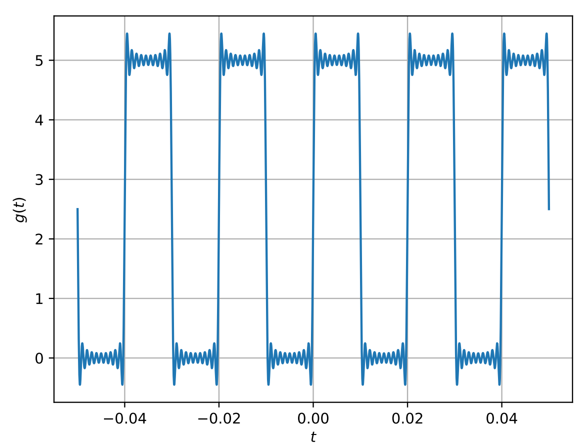

Fig. 1.4: Gibbs phenomenon.

Problem 1.5. Generate $g(t)$ using an arduino for $A = 5\,V$ and $T = 20\,ms$ using the blink.ino program.

2 Filter

2.1 RC Circuit

Problem 2.1. Refer to the circuit in Fig. 2.1. Suppose you are told that C has a resistance given by $\frac{1}{sC}$. Find the ratio $H(s)$ of the output voltage and input voltage using node analysis. The above circuit is known as a low pass filter and $H(s)$ is known as the transfer function.

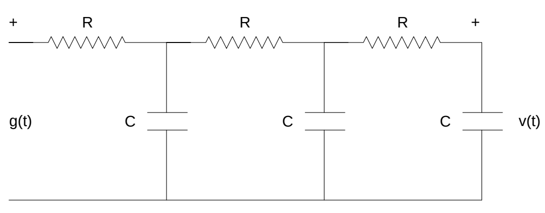

Fig. 2.1: Three stage $R - C$ low pass filter circuit

Solution: The equations at the nodes are given by

$$\frac{V_1 - V_i}{R} + sCV_1 + \frac{V_1 - V_2}{R} = 0 \tag{2.1}$$

$$\frac{V_2 - V_1}{R} + sCV_2 + \frac{V_2 - V_o}{R} = 0 \tag{2.2}$$

$$\frac{V_o - V_2}{R} + sCV_o = 0 \tag{2.3}$$

which can be expressed as

$$\begin{pmatrix} sC + \frac{2}{R} & -\frac{1}{R} & 0 \\ -\frac{1}{R} & sC + \frac{2}{R} & -\frac{1}{R} \\ 0 & -\frac{1}{R} & sC + \frac{1}{R} \end{pmatrix} \begin{pmatrix} \frac{V_1}{V_i} \\ \frac{V_2}{V_i} \\ \frac{V_o}{V_i} \end{pmatrix} = \begin{pmatrix} \frac{1}{R} \\ 0 \\ 0 \end{pmatrix} \tag{2.4}$$

Thus,

$$H(s) = \frac{V_o}{V_i} = \frac{\begin{vmatrix} sC + \frac{2}{R} & -\frac{1}{R} & \frac{1}{R} \\ -\frac{1}{R} & sC + \frac{2}{R} & 0 \\ 0 & -\frac{1}{R} & 0 \end{vmatrix}}{\begin{vmatrix} sC + \frac{2}{R} & -\frac{1}{R} & 0 \\ -\frac{1}{R} & sC + \frac{2}{R} & -\frac{1}{R} \\ 0 & -\frac{1}{R} & sC + \frac{1}{R} \end{vmatrix}} \tag{2.5}$$

$$= \frac{1/R^3}{\left(sC + \frac{1}{R}\right)\left\{\left(sC + \frac{2}{R}\right)^2 - \frac{1}{R^2}\right\} - \frac{1}{R^2}\left(sC + \frac{2}{R}\right)} \tag{2.6}$$

which can be expressed as

$$H(s) = \frac{1}{(sCR + 1)\left\{(sCR + 2)^2 - 1\right\} - (sCR + 2)} \tag{2.7}$$

$$= \frac{1}{(sCR + 2)^3 - (sCR + 2)^2 - 2\,(sCR + 2) + 1} \tag{2.8}$$

$$= \frac{1}{(sCR)^3 - 5\,(sCR)^2 + 6sCR + 1} \tag{2.9}$$

Problem 2.2. Substitute $s = \jmath 2\pi f, \jmath = \sqrt{-1}$ in (2.9) to obtain $H(f)$. $H(f)$ is known as the frequency response. Plot $|H(f)|$ in octave for $-20 < f < 20$, given that $R = 1\,k\Omega$ and $C = 10\,\mu F$.

Solution: Type the following code to get Fig. 2.2. You will find that $H(f)$ is a low pass filter.

```
import numpy as np
import matplotlib.pyplot as plt



#Filter Characteristics

R = 1e3; #10K ohm resistance
C = 10e-6;#10 uF capacitance

#Plotting the filter amplitude
    response
T = 0.02;
f_0 = 1/T;
f = np.linspace(-1.5*f_0,1.5*f_0,1
    e2)
s = 1j*2*np.pi*f

den = np.polyval([1,-5, 6, 1],s*C*
```

```
    R);
H = 1/den;

plt.plot(f,abs(H))
plt.grid()# minor
plt.xlabel('$f$ (Hz)')
plt.ylabel('$H(f)$')
#Save figure
plt.savefig('../figs/2.2.eps')
plt.show()
```

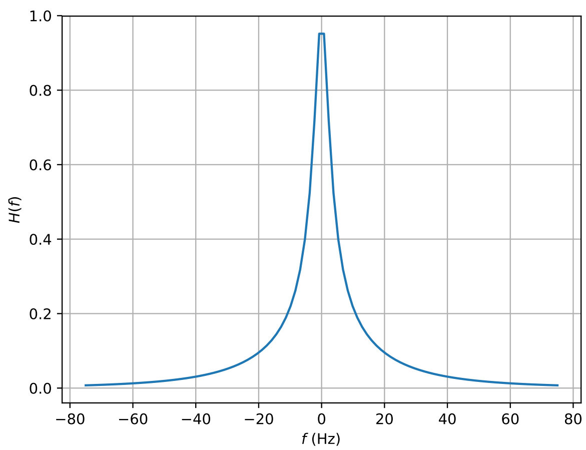

Fig. 2.2: Frequency response of the $R - C$ filter

Problem 2.3. Find the frequency at which $|H(f)|^2 = \frac{1}{2}$. This frequency is known as the 3-dB bandwidth of $H(f)$.

Solution: Substituting sCR = ȷx in (2.9),

$$\left|H(\jmath x)\right| = \frac{1}{\sqrt{2}} \tag{2.10}$$

$$\Rightarrow -jx^3 + 5x^2 + j6x + 1 = \sqrt{2} \tag{2.11}$$

$$\Rightarrow x^2(6 - x^2)^2 + (1 + 5x^2)^2 = 2 \tag{2.12}$$

$$\Rightarrow x^6 + 13x^4 + 46x^2 - 1 = 0 \tag{2.13}$$

Letting $y = x^2$, we obtain the cubic equation

$$y^3 + 13y^2 + 46y - 1 = 0 \tag{2.14}$$

The following script gives the 3 dB bandwidth for the filter H by choosing the real root.

```
import numpy as np
#Filter Characteristics
R = 1e3; #1K ohm resistance
C = 10e-6;#10 uF capacitance

#finding 3 dB bandwidth
    numerically
print(np.sqrt(np.roots([1,  13,
    46,  -1]))/(2*np.pi*R*C))
```

This yields the value $f_{3dB} = 2.3395$ Hz.

Problem 2.4. Obtain the 3 dB bandwidth by solving the cubic equation in the previous problem

Solution: In the above, let $y = z - \frac{13}{3}$. Then the equation becomes

$$\Rightarrow z^3 - (31/3)z - 1015/27 = 0 \tag{2.15}$$

This equation has the theoretical solution evaluated by the following script

```
import numpy as np

#Filter Characteristics

R = 1e3; #1K ohm resistance
C = 10e-6;#10 uF capacitance


#finding 3 dB bandwidth
    theoretically

q = -31/3;
r = -1015/27;

print(np.sqrt((-r/2 + np.sqrt(r
    **2/4 +q**3/27))**(1/3) + (-r/2
    - np.sqrt(r**2/4 +q**3/27))
    **(1/3) - 13/3)/(2*np.pi*R*C))
```

Note that this script gives the same result as the one in the previous problem.

Problem 2.5. Suppose the square wave in Fig. 1.1 is given as input to the filter in Fig. 2.2. Find and plot the filter output.

Solution: Using sinusoidal steady state analysis, if the input to the filter is $\cos 2\pi n f t$, the output is given by

$$|H(nf)| \cos\left\{2\pi n f t + \angle H(nf)\right\} \tag{2.16}$$

Using the principle of superposition, for the input

$$\sum_{n=0}^{\infty} a_n \cos 2\pi n f t + b_n \sin 2\pi n f t \tag{2.17}$$

the output will be

$$\sum_{n=0}^{\infty} a_n |H(nf)| \cos\{2\pi nft + \angle H(nf)\} + b_n |H(nf)| \sin\{2\pi nft + \angle H(nf)\} \quad (2.18)$$

Suitably modifying the program in Problem 1.4,

```
import numpy as np
import matplotlib.pyplot as plt


T_0 = 0.01
T = 0.02
f = 1/T
A = 5
simlen = 1e3
R = 1e3; #10 k resistance
C = 10e-6;#10 uF capacitance

 #generating points in t-axis

t = np.linspace(0,0.1,simlen); #
    generating points in t-axis

for n in range(21):
        if n == 0:
                g = A*T_0/T;
        else:
                an = 2*A*np.sin(2*
                    np.pi*n*f*T_0)
                    /(2*np.pi*n*f*T)
                    ; #Computing
                    coefficients
                bn = 2*A*(1 - np.
                    cos(2*np.pi*n*f*
                    T_0))/(2*np.pi*n
                    *f*T); #
                    Computing
                    coefficients
                s = 1j*2*np.pi*n*f
                    ;
                den = np.polyval
                    ([1, -5, 6, 1],s
                    *C*R);
                Hn = 1/den; #
                    Frequency
                    response
                thetan = np.angle(
                    Hn);
                cost = np.cos(2*np
                    .pi*n*f*t +
                    thetan);#
                    Computing cosine
                     term
                sint = np.sin(2*np
                    .pi*n*f*t+
                    thetan);#
                    Computing sine
                    term


                g = g + np.abs(Hn)
                    *an*cost + np.
                    abs(Hn)*bn*sint;
                     #evaluating the
                     summation



plt.plot(t,g)
plt.grid ()
plt.xlabel('$t$')
plt.ylabel('$g(t)$')
#Save figure, remove the following
    line while running the program
plt.savefig('../figs/2.5.eps')
plt.show()
```

The output of the filter is shown in Fig. 2.5

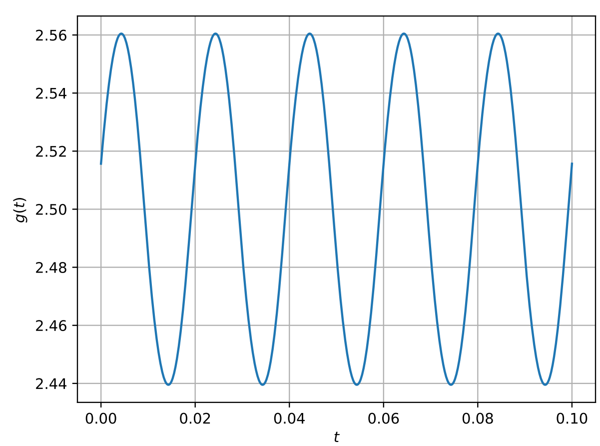

Fig. 2.5: Output of the $R-C$ filter

Problem 2.6. Run the program in problem 2.5 by changing the for loop to

```
for n in range(2):
```

Compare this output with the one in Fig. 2.5 by plotting in the same graph.

Problem 2.7. Interpret the result in problem 2.5.

Solution: In Fig. 2.2, $|H(0)| = 1$ and $|H(50)| = 0.02$. All other values of H are very small. $|H(0)| = 1$ contributes the DC component and $|H(50)| = 0.02$ yields the sinusoidal component of 50 Hz. Thus, $H(f)$ filters all higher harmonics in the square wave in Fig. 1.1.