Fourier Series

Karthik Ravula
AI21BTECH11024

Contents

***Abstract*—This manual provides a simple introduction to Fourier Series**

1 Periodic Function

Let

$$x(t) = A_0 \left|\sin\left(2\pi f_0 t\right)\right| \tag{1.1}$$

1.1 Plot $x(t)$.
Solution:

```
wget https://github.com/karthik6281/Signal−
    Processing/tree/master/fourier/codes/1.1.
    py
```

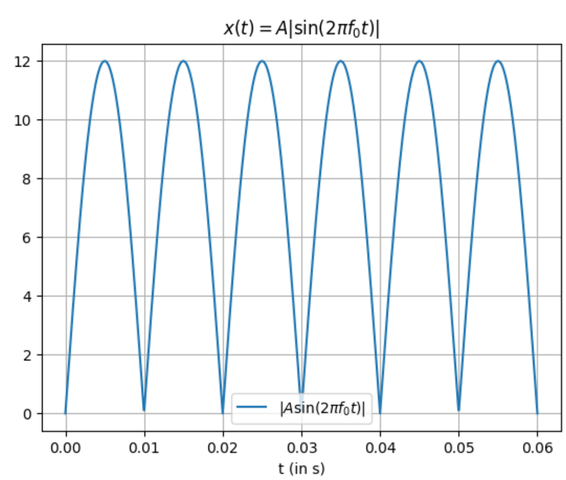

Fig. 1.1

1.2 Show that $x(t)$ is periodic and find its period.
Solution: If a signal $x(t)$ is periodic then

$$x(t+T) = x(t) \tag{1.2}$$

where T is known as fundamental period. Since $|sin\theta|$ function is periodic, x(t) is also periodic.

$$\text{Fundamental Period} = T = \frac{1}{2}\left(\frac{2\pi}{2\pi f_0}\right) \tag{1.3}$$

$$= \frac{1}{2f_0} \tag{1.4}$$

2 Fourier Series

Consider $A_0 = 12$ and $f_0 = 50$ for all numerical calculations.

2.1 If

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{\jmath 2\pi k f_0 t} \tag{2.1}$$

show that

$$c_k = f_0 \int_{-\frac{1}{2f_0}}^{\frac{1}{2f_0}} x(t) e^{-\jmath 2\pi k f_0 t}\, dt \tag{2.2}$$

Solution: From (2.1),

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{\jmath 2\pi k f_0 t} \tag{2.3}$$

Mulitply $e^{-\jmath 2\pi l f_0 t}$ on both sides

$$x(t)e^{-\jmath 2\pi l f_0 t} = \sum_{k=-\infty}^{\infty} c_k e^{\jmath 2\pi k f_0 t} e^{-\jmath 2\pi l f_0 t} \tag{2.4}$$

Integrate on both sides with respect to ’t’ between $-T$ to T where T is fundamental time period of x(t).
Using (1.4),

$$T = \frac{1}{2f_0} \tag{2.5}$$

$$\int_{-\frac{1}{2f_0}}^{\frac{1}{2f_0}} x(t)e^{-\jmath 2\pi k f_0 t}\, dt = \int_{-\frac{1}{2f_0}}^{\frac{1}{2f_0}} \sum_{k=-\infty}^{\infty} c_k e^{\jmath 2\pi (k-l) f_0 t}\, dt \tag{2.6}$$

$$= \sum_{k=-\infty}^{\infty} c_k \int_{-\frac{1}{2f_0}}^{\frac{1}{2f_0}} e^{\jmath 2\pi (k-l) f_0 t}\, dt \tag{2.7}$$

The above integral:

$$\int_{-\frac{1}{2f_0}}^{\frac{1}{2f_0}} e^{\jmath 2\pi (k-l) f_0 t}\, dt = \begin{cases} 0 & k \neq l \\ \int_{-\frac{1}{2f_0}}^{\frac{1}{2f_0}} 1\, dt & k = l \end{cases} \tag{2.8}$$

$$\therefore \int_{-\frac{1}{2f_0}}^{\frac{1}{2f_0}} x(t)e^{-\jmath 2\pi k f_0 t}\, dt = \left(\frac{1}{f_0}\right) c_k \tag{2.9}$$

$$\therefore c_k = f_0 \int_{-\frac{1}{2f_0}}^{\frac{1}{2f_0}} x(t)e^{-\jmath 2\pi k f_0 t}\, dt \tag{2.10}$$

2.2 Find c_k for (1.1)
Solution: c_k can be calculated even simpler by using

$$c_k = 2f_0 \int_0^{\frac{1}{2f_0}} x(t)e^{-\jmath 2\pi k f_0 t}\, dt \tag{2.11}$$

$x(t) = A_0 \sin(2\pi f_0 t)$ in 0 to $\frac{1}{2f_0}$ region.
Also,

$$\sin\theta = \frac{e^{\jmath\theta} - e^{-\jmath\theta}}{2\jmath} \tag{2.12}$$

Using (2.12),

$$c_k = 2f_0 \int_0^{\frac{1}{2f_0}} A_0 \left(\frac{e^{\jmath 2\pi f_0 t} - e^{-\jmath 2\pi f_0 t}}{2\jmath}\right) e^{-\jmath 2\pi k f_0 t}\, dt \tag{2.13}$$

$$= A_0 f_0 \int_0^{\frac{1}{2f_0}} \left(\frac{e^{\jmath 2\pi (1-k) f_0 t} - e^{\jmath 2\pi (-1-k) f_0 t}}{\jmath}\right) dt \tag{2.14}$$

$$= A_0 f_0 \left[\frac{e^{\jmath 2\pi (1-k) f_0 t}}{-2\pi(1-k) f_0}\Bigg|_0^{\frac{1}{2f_0}} - \frac{e^{\jmath 2\pi (-1-k) f_0 t}}{-2\pi(-1-k) f_0}\Bigg|_0^{\frac{1}{2f_0}}\right] \tag{2.15}$$

$$= A_0 \left[\frac{e^{j\pi(1-k)} - 1}{2\pi(k-1)} - \frac{e^{-j\pi(1+k)} - 1}{2\pi(k+1)}\right] \tag{2.16}$$

$$= \begin{cases} \frac{2A_0}{\pi(1-k^2)} & k = even \\ 0 & k = odd \end{cases} \tag{2.17}$$

2.3 Verify (1.1) using python.

Solution:

```
wget https://github.com/karthik6281/Signal-Processing/tree/master/fourier/codes/2.3.py
```

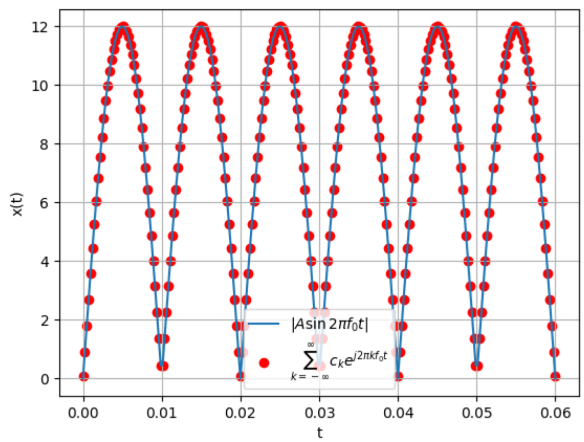

Fig. 2.3

2.4 Show that

$$x(t) = \sum_{k=0}^{\infty} \left(a_k \cos \jmath 2\pi k f_0 t + b_k \sin \jmath 2\pi k f_0 t\right) \tag{2.18}$$

and obtain the formulae for a_k and b_k. **Solution:** Using (2.1),

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{\jmath 2\pi k f_0 t} \tag{2.19}$$

As,

$$e^{\jmath 2\pi k f_0 t} = \cos(2\pi k f_0 t) + j\sin(2\pi k f_0 t) \tag{2.20}$$

Substituting leads to

$$x(t) = \sum_{k=-\infty}^{\infty} c_k \left[\cos(2\pi k f_0 t) + j\sin(2\pi k f_0 t)\right] \tag{2.21}$$

$$= \sum_{k=-\infty}^{\infty} c_k \cos(2\pi k f_0 t) + j c_k \sin(2\pi k f_0 t) \tag{2.22}$$

$$= \sum_{k=-\infty}^{-1} \left[c_k \cos(2\pi k f_0 t) + j c_k \sin(2\pi k f_0 t)\right]$$
$$+ c_0 + \sum_{k=1}^{\infty} \left[c_k \cos(2\pi k f_0 t) + j c_k \sin(2\pi k f_0 t)\right] \tag{2.23}$$

$$= \sum_{k=1}^{\infty} \left[c_{-k} \cos(2\pi k f_0 t) - j c_{-k} \sin(2\pi k f_0 t)\right]$$
$$+ c_0 + \sum_{k=1}^{\infty} \left[c_k \cos(2\pi k f_0 t) + j c_k \sin(2\pi k f_0 t)\right] \tag{2.24}$$

$$= c_0 + \sum_{k=1}^{\infty} \left[(c_k + c_{-k}) \cos(2\pi k f_0 t) + j(c_k - c_{-k}) \sin(2\pi k f_0 t)\right] \tag{2.25}$$

Replacing $(c_k + c_{-k}) \to a_k$ and $j(c_k - c_{-k}) \to b_k$,

$$= c_0 + \sum_{k=1}^{\infty} (a_k \cos 2\pi k f_0 t + b_k \sin 2\pi k f_0 t) \tag{2.26}$$

$$= \sum_{k=0}^{\infty} (a_k \cos 2\pi k f_0 t + b_k \sin 2\pi k f_0 t) \tag{2.27}$$

$$\therefore a_k = \begin{cases} c_k + c_{-k} & k \neq 0 \\ c_0 & k = 0 \end{cases} \tag{2.28}$$

$$b_k = j(c_k - c_{-k}) \tag{2.29}$$

Using (2.2),

$$c_k = f_0 \int_{-\frac{1}{2f_0}}^{\frac{1}{2f_0}} x(t) e^{-\jmath 2\pi k f_0 t}\, dt \tag{2.30}$$

$$c_{-k} = f_0 \int_{-\frac{1}{2f_0}}^{\frac{1}{2f_0}} x(t) e^{\jmath 2\pi k f_0 t}\, dt \tag{2.31}$$

$$a_k = c_k + c_{-k} = f_0 \int_{-\frac{1}{2f_0}}^{\frac{1}{2f_0}} x(t) \left[e^{-\jmath 2\pi k f_0 t} + e^{\jmath 2\pi k f_0 t}\right] dt \tag{2.32}$$

$$= 2 f_0 \int_{-\frac{1}{2f_0}}^{\frac{1}{2f_0}} x(t) \cos(2\pi k f_0 t)\, dt \tag{2.33}$$

Parallely,

$$b_k = -2 j f_0 \int_{-\frac{1}{2f_0}}^{\frac{1}{2f_0}} x(t) \sin(2\pi k f_0 t)\, dt \tag{2.34}$$

2.5 Find a_k and b_k for (1.1)

Solution: Using (2.28) and (2.29) with (2.17),

$$a_k = c_k + c_{-k} = \begin{cases} \frac{4A_0}{\pi(1-k^2)} & k = even \\ \frac{2A_0}{\pi} & k = 0 \\ 0 & k = odd \end{cases} \tag{2.35}$$

$$b_k = j(c_k - c_{-k}) = 0 \tag{2.36}$$

2.6 Verify (2.18) using python.

Solution:

```
wget https://github.com/karthik6281/Signal
    −Processing/tree/master/fourier/codes
    /2.6.py
```

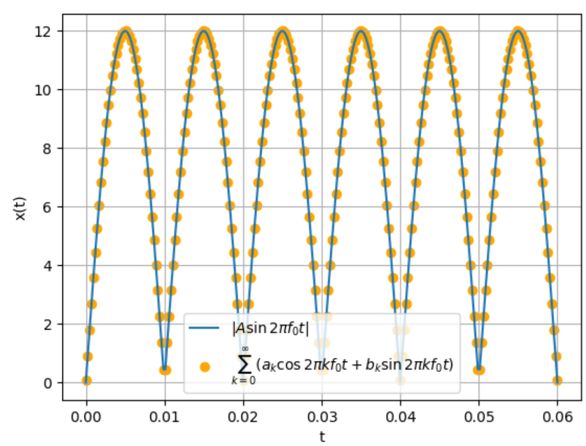

Fig. 2.6

3 Fourier Transform

3.1

$$\delta(t) = 0, \quad t \neq 0 \tag{3.1}$$

$$\int_{-\infty}^{\infty} \delta(t)\, dt = 1 \tag{3.2}$$

3.2 The Fourier Transform of $g(t)$ is

$$G(f) = \int_{-\infty}^{\infty} g(t)e^{-j2\pi ft}\, dt \tag{3.3}$$

3.3 Show that

$$g(t-t_0) \overset{\mathcal{F}}{\longleftrightarrow} G(f)e^{-j2\pi f t_0} \tag{3.4}$$
$$\tag{3.5}$$

Solution:

$$\mathcal{F}\{g(t-t_0)\} = \int_{-\infty}^{\infty} g(t-t_0)e^{-j2\pi ft}\, dt \tag{3.6}$$
$$= e^{-j2\pi f t_0}\int_{-\infty}^{\infty} g(t-t_0)e^{-j2\pi f(t-t_0)}\, dt \tag{3.7}$$
$$= G(f)e^{-j2\pi f t_0} \tag{3.8}$$

3.4 Show that

$$G(t) \overset{\mathcal{F}}{\longleftrightarrow} g(-f) \tag{3.9}$$

Solution: From the definition of Inverse Fourier Transform

$$g(t) = \int_{-\infty}^{\infty} G(f)e^{j2\pi ft}\, df \tag{3.10}$$

Replace t → f,

$$g(f) = \int_{-\infty}^{\infty} G(t)e^{j2\pi ft}\, dt \tag{3.11}$$

Replace f → -f,

$$g(-f) = \int_{-\infty}^{\infty} G(t)e^{-j2\pi ft}\, dt \tag{3.12}$$
$$= \mathcal{F}\{G(t)\} \tag{3.13}$$

$$\therefore G(t) \overset{\mathcal{F}}{\longleftrightarrow} g(-f) \tag{3.14}$$

3.5 $\delta(t) \overset{\mathcal{F}}{\longleftrightarrow} ?$
Solution:

$$\mathcal{F}\{\delta(t)\} = \int_{-\infty}^{\infty} \delta(t)e^{-j2\pi ft}\, dt \tag{3.15}$$
$$= \int_{-\infty}^{\infty} \delta(t)\, dt \tag{3.16}$$
$$\tag{3.17}$$

Since $e^{-j2\pi ft} = 1$ for t=0 and remaining integrand is zero for t $\neq$ 0.

$$= \int_{-\infty}^{\infty} \delta(t)\, dt \tag{3.18}$$
$$= 1 \tag{3.19}$$

3.6 $e^{-j2\pi f_0 t} \overset{\mathcal{F}}{\longleftrightarrow} ?$
Solution:

$$\mathcal{F}\left\{e^{-j2\pi f_0 t}\right\} = \int_{-\infty}^{\infty} 1 \cdot e^{-j2\pi(f+f_0)t}\, dt \tag{3.20}$$
$$= \int_{-\infty}^{\infty} \mathcal{F}\{\delta(t)\}\, e^{-j2\pi(f+f_0)t}\, dt \tag{3.21}$$

Using (3.9),

$$= \delta(-(f+f_0)) = \delta(f+f_0) \tag{3.22}$$

3.7 $\cos(2\pi f_0 t) \overset{\mathcal{F}}{\longleftrightarrow} ?$
Solution:

$$\mathcal{F}\{\cos(2\pi f_0 t)\} = \mathcal{F}\left\{\frac{e^{j2\pi f_0 t} + e^{-j2\pi f_0 t}}{2}\right\} \tag{3.23}$$

Using (3.22),

$$= \frac{\delta(f+f_0) + \delta(f-f_0)}{2} \tag{3.24}$$

3.8 Find the Fourier Transform of $x(t)$ and plot it. Verify using python.
Solution:

$$\mathcal{F}\{x(t)\} = \mathcal{F}\left\{\sum_{k=-\infty}^{\infty} c_k e^{j2\pi k f_0 t}\right\} \tag{3.25}$$
$$X(f) = \sum_{k=-\infty}^{\infty} c_k \delta(f - k f_0) \tag{3.26}$$

```
wget https://github.com/karthik6281/Signal
    −Processing/tree/master/fourier/codes
    /3.8.py
```

3.9 Show that

$$\text{rect}\,(t) \overset{\mathcal{F}}{\longleftrightarrow} \text{sinc}\,(t) \tag{3.27}$$

Verify using python.
Solution:

$$\mathcal{F}\{\text{rect}\,(t)\} = \int_{-\infty}^{\infty} \text{rect}\,(t)e^{-j2\pi ft}\, dt \tag{3.28}$$
$$= \int_{-\frac{1}{2}}^{\frac{1}{2}} 1 \cdot e^{-j2\pi ft}\, dt \tag{3.29}$$

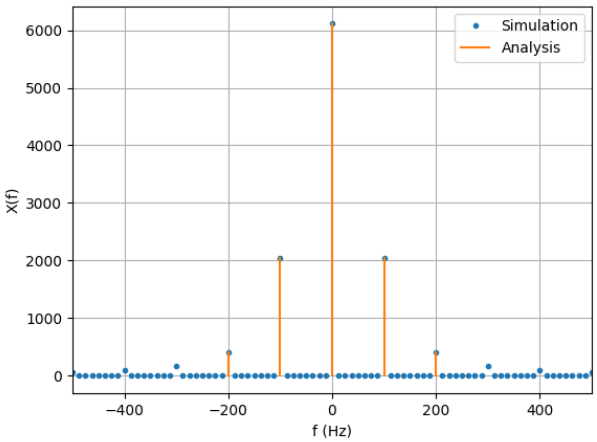

Fig. 3.8

$$= \frac{e^{-j2\pi f t}}{-j2\pi f}\Big|_{-\frac{1}{2}}^{\frac{1}{2}} \tag{3.30}$$
$$= \operatorname{sinc}(t) \tag{3.31}$$

```
wget https://github.com/karthik6281/
    Signal−Processing/tree/master/fourier/
    codes/3.9.py
```

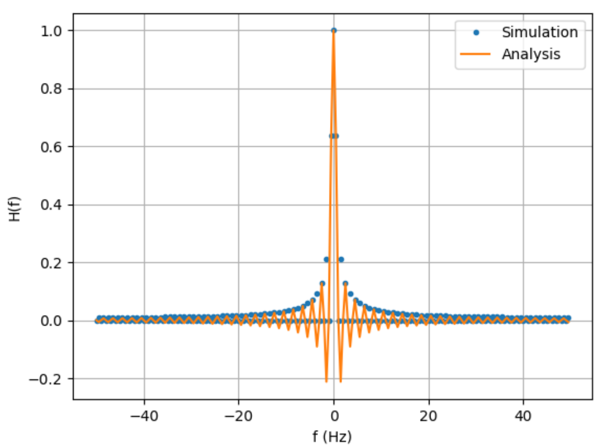

Fig. 3.9

3.10 $\operatorname{sinc}(t) \overset{\mathcal{F}}{\longleftrightarrow} ?$. Verify using python.
Solution: Using (3.31), (3.14) and even property of rect function,

$$\operatorname{sinc}(t) \overset{\mathcal{F}}{\longleftrightarrow} \operatorname{rect}(t) \tag{3.32}$$

```
wget https://github.com/karthik6281/Signal
    −Processing/tree/master/fourier/codes
    /3.10.py
```

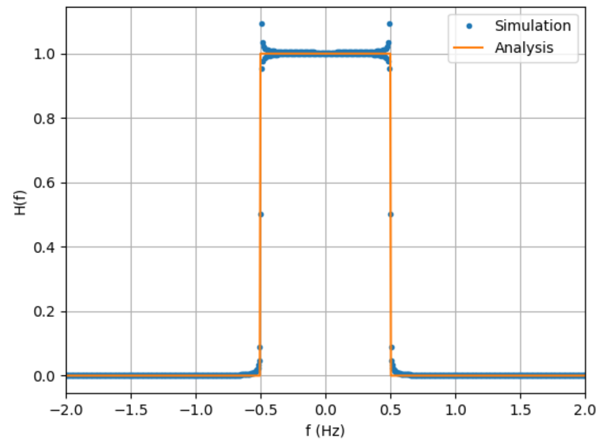

Fig. 3.10

4 Filter

4.1 Find $H(f)$ which transforms $x(t)$ to DC 5V. **Solution:**

$$x[t] \longrightarrow \boxed{h[t]} \longrightarrow x[t] * h[t]$$

$$X(f) \longrightarrow \boxed{H(f)} \longrightarrow X(f)H(f)$$

$$X(f)H(f) = V_0\delta(f) \tag{4.1}$$

Above equation indicates that H(f) will pass X(f) for f=0.
∴ H(f) should be a low pass filter.

$$|H(f)| = \frac{V_0}{\left(\frac{2A_0}{\pi}\right)} = \frac{V_0\pi}{2A_0} \tag{4.2}$$
$$H(f) = \frac{V_0\pi}{2A_0} \quad in \quad -2f_0 \le f \le 2f_0 \tag{4.3}$$
$$H(f) = \frac{V_0\pi}{2A_0}\operatorname{rect}\left(\frac{f}{4f_0}\right) \tag{4.4}$$

4.2 Find $h(t)$. **Solution:** Using (4.4) and (3.32),

$$h(t) = \frac{2V_0\pi f_0}{A_0}\operatorname{sinc}(4f_0t) \tag{4.5}$$

4.3 Verify your result using through convolution. **Solution:**

```
wget https://github.com/karthik6281/Signal
    -Processing/tree/master/fourier/codes
    /4.3.py
```

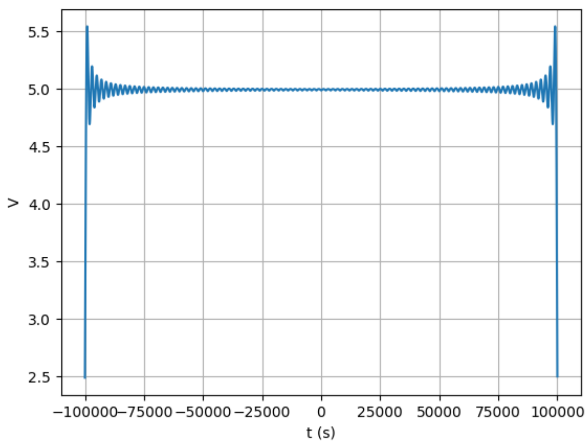

Fig. 4.3

5 Filter Design

5.1 Design a Butterworth filter for $H(f)$.
Solution: The Butterworth filter has an amplitude response given by

$$|H(f)|^2 = \frac{1}{\left(1 + \left(\frac{f}{f_c}\right)^{2n}\right)} \tag{5.1}$$

where n is the order of the filter and f_c is the cutoff frequency. The attenuation at frequency f is given by

$$A = -10\log_{10}|H(f)|^2 \tag{5.2}$$
$$= -20\log_{10}|H(f)| \tag{5.3}$$

We consider the following design parameters for our lowpass analog Butterworth filter:

a) Passband edge, $f_p = 50$ Hz
b) Stopband edge, $f_s = 100$ Hz
c) Passband attenuation, $A_p = -1$ dB
d) Stopband attenuation, $A_s = -20$ dB

We are required to find a desriable order n and cutoff frequency f_c for the filter. From (5.3),

$$A_p = -10\log_{10}\left[1 + \left(\frac{f_p}{f_c}\right)^{2n}\right] \tag{5.4}$$

$$A_s = -10\log_{10}\left[1 + \left(\frac{f_s}{f_c}\right)^{2n}\right] \tag{5.5}$$

Thus,

$$\left(\frac{f_p}{f_c}\right)^{2n} = 10^{-\frac{A_p}{10}} - 1 \tag{5.6}$$

$$\left(\frac{f_s}{f_c}\right)^{2n} = 10^{-\frac{A_s}{10}} - 1 \tag{5.7}$$

Therefore, on dividing the above equations and solving for n,

$$n = \frac{\log\left(10^{-\frac{A_s}{10}} - 1\right) - \log\left(10^{-\frac{A_p}{10}} - 1\right)}{2\left(\log f_s - \log f_p\right)} \tag{5.8}$$

In this case, making appropriate susbstitutions gives $n = 4.29$. Hence, we take $n = 5$. Solving for f_c in (5.6) and (5.7),

$$f_{c1} = f_p\left[10^{-\frac{A_p}{10}} - 1\right]^{-\frac{1}{2n}} = 57.23\,\text{Hz} \tag{5.9}$$

$$f_{c2} = f_s\left[10^{-\frac{A_s}{10}} - 1\right]^{-\frac{1}{2n}} = 63.16\,\text{Hz} \tag{5.10}$$

Hence, we take $f_c = \sqrt{f_{c1}f_{c2}} = 60\,\text{Hz}$ approximately.

5.2 Design a Chebyshev filter for $H(f)$.bjjk,bv,kgg
Solution: The Chebyshev filter has an amplitude response given by

$$|H(f)|^2 = \frac{1}{\left(1 + \epsilon^2 C_n^2\left(\frac{f}{f_c}\right)\right)} \tag{5.11}$$

where

a) n is the order of the filter
b) ϵ is the ripple
c) f_c is the cutoff frequency
d) $C_n = \cosh^{-1}(n\cosh x)$ denotes the n$^{\text{th}}$ order Chebyshev polynomial, given by

$$c_n(x) = \begin{cases} \cos\left(n\cos^{-1}x\right) & |x| \le 1 \\ \cosh\left(n\cosh^{-1}x\right) & \text{otherwise} \end{cases} \tag{5.12}$$

We are given the following specifications:

a) Passband edge (which is equal to cutoff frequency), $f_p = f_c$
b) Stopband edge, f_s
c) Attenuation at stopband edge, A_s
d) Peak-to-peak ripple δ in the passband. It is given in dB and is related to ϵ as

$$\delta = 10\log_{10}\left(1 + \epsilon^2\right) \tag{5.13}$$

and we must find a suitable n and ϵ. From (5.13),

$$\epsilon = \sqrt{10^{\frac{\delta}{10}} - 1} \tag{5.14}$$

At $f_s > f_p = f_c$, using (5.12), A_s is given by

$$A_s = -10 \log_{10} \left[1 + \epsilon^2 c_n^2 \left(\frac{f_s}{f_p} \right) \right] \tag{5.15}$$

$$\implies c_n \left(\frac{f_s}{f_p} \right) = \frac{\sqrt{10^{-\frac{A_s}{10}} - 1}}{\epsilon} \tag{5.16}$$

$$\implies n = \frac{\cosh^{-1} \left(\frac{\sqrt{10^{-\frac{A_s}{10}} - 1}}{\epsilon} \right)}{\cosh^{-1} \left(\frac{f_s}{f_p} \right)} \tag{5.17}$$

We consider the following specifications:

a) Passband edge/cutoff frequency, $f_p = f_c = 60\,\text{Hz}$.
b) Stopband edge, $f_s = 100\,\text{Hz}$.
c) Passband ripple, $\delta = 0.5\,\text{dB}$
d) Stopband attenuation, $A_s = -20\,\text{dB}$

$\epsilon = 0.35$ and $n = 3.68$. Hence, we take $n = 4$ as the order of the Chebyshev filter.

5.3 Design a circuit for your Butterworth filter.
Solution: Looking at the table of normalized element values L_k, C_k, of the Butterworth filter for order 5, and noting that de-normalized values L'_k and C'_k are given by

$$C'_k = \frac{C_k}{\omega_c} \qquad L'_k = \frac{L_k}{\omega_c} \tag{5.18}$$

De-normalizing these values, taking $f_c = 60$ Hz,

$$C'_1 = C'_5 = 1.64\,\text{mF} \tag{5.19}$$
$$L'_2 = L'_4 = 4.29\,\text{mH} \tag{5.20}$$
$$C'_3 = 5.31\,\text{mF} \tag{5.21}$$

The L-C network is shown in Fig. 5.3.

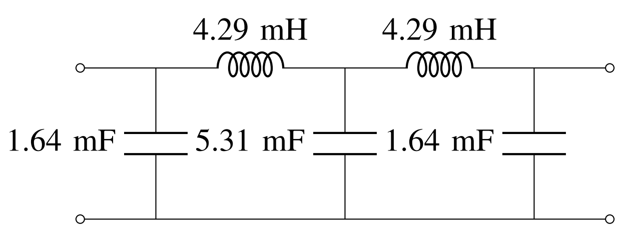

Fig. 5.3: L-C Butterworth Filter

Below python code plot the figure 5.3

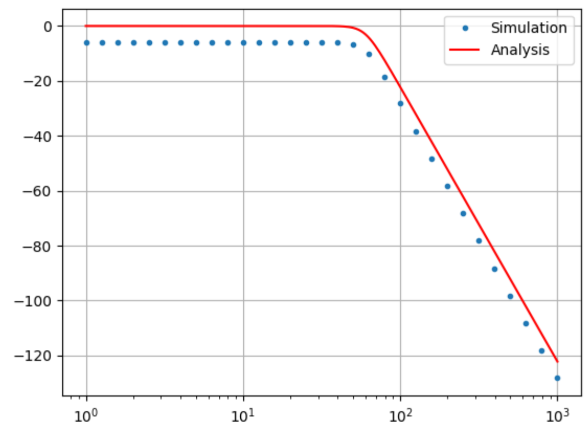

Fig. 5.3: Simulation of Butterworth filter.

```
wget https://github.com/
    karthik6281/Signal–
    Processing/tree/master/
    fourier/codes/5.3.py
```

5.4 Design a circuit for your Chebyshev filter.
Solution: Looking at the table of normalized element values of the Chebyshev filter for order 3 and 0.5 dB ripple, and de-nomrmalizing those values, taking $f_c = 50\,\text{Hz}$,

$$C'_1 = 4.43\,\text{mF} \tag{5.22}$$
$$L'_2 = 3.16\,\text{mH} \tag{5.23}$$
$$C'_3 = 6.28\,\text{mF} \tag{5.24}$$
$$L'_4 = 2.23\,\text{mH} \tag{5.25}$$

The L-C network is shown in Fig. 5.4.

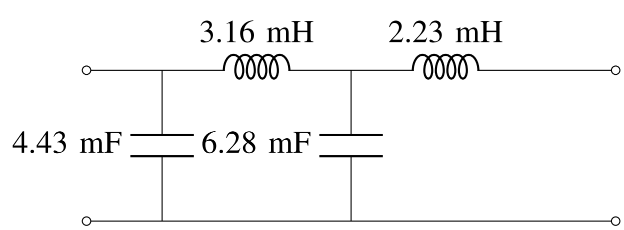

Fig. 5.4: L-C Chebyshev Filter

Below python code plot the figure 5.4

```
wget https://github.com/
    karthik6281/Signal–
    Processing/tree/master/
    fourier/codes/5.4.py
```

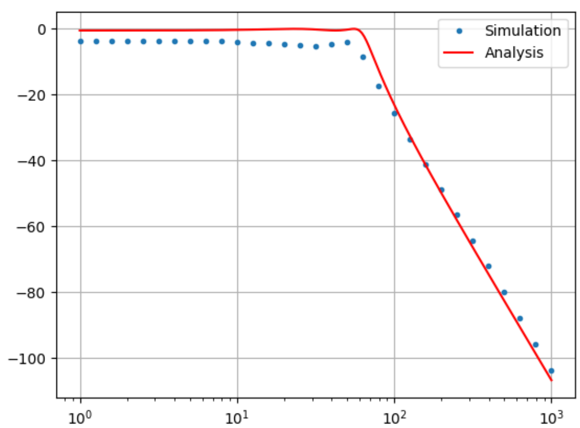

Fig. 5.4: Simulation of Chebyshev filter.