Audio Filtering

EE23BTECH11042-M KHUSINADHA NAIK*

I. Digital Filter

I.1 The sound file used for this code is obtained from the below link

```
https://github.com/Meowfessor96/
    audio_filtering/blob/main/audio_filtering/
    codes/audio.wav
```

I.2 Audio filtering is carried out using the following Python Code

```
import soundfile as sf
from scipy import signal

#read .wav file
input_signal,fs = sf.read('audio.wav')

#sampling frequency of Input signal
sampl_freq=fs

#order of the filter
order=4

#cutoff frquency 1000Hz
cutoff_freq=1000.0

#digital frequency
Wn=2*cutoff_freq/sampl_freq

# b and a are numerator and denominator
    polynomials respectively
b, a = signal.butter(order,Wn, 'low')

#filter the input signal with butterworth filter
output_signal = signal.filtfilt(b, a,
    input_signal)
#output_signal = signal.lfilter(b, a,
    input_signal)

#write the output signal into .wav file
sf.write('filtered_audio.wav', output_signal,
    fs)
```

I.3 Audio is analysed using a spectrogram found in the following website :

The figures represent the distribution of sound among frequencies in the x axis and similarly the y-axis represents the intensity of the sound

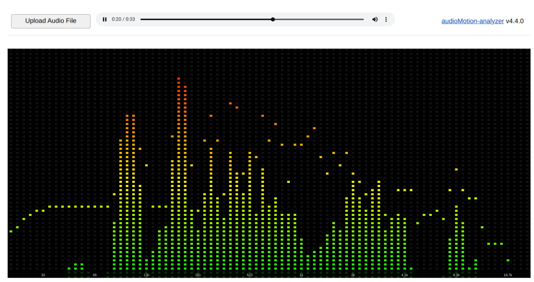

Fig. 1. frequenxy distribution of the audio file before Filtering(at 0:20)

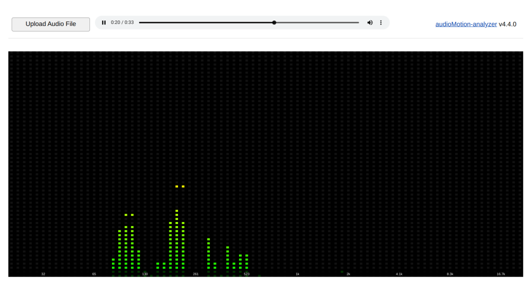

Fig. 2. frequenxy distribution of the audio file after Filtering(at 0:20)

II. Difference Equation

II.1 Let

$$x(n) = \left\{ \underset{\uparrow}{1}, 2, 3, 4, 2, 1 \right\} \tag{1}$$

Sketch $x(n)$.

II.2 Let

$$y(n) + \frac{1}{2}y(n-1) = x(n) + x(n-2), \quad y(n) = 0, n < 0 \tag{2}$$

Sketch $y(n)$.
Solve

Solution: $x(n)$ and $y(n)$ values for different values of n are generated in a text file using C file.

https://github.com/Meowfessor96/audio_filtering/blob/main/audio_filtering/codes/2_2.c

The following code plots (1) and (2)

https://github.com/Meowfessor96/audio_filtering/blob/main/audio_filtering/codes/2.2.py

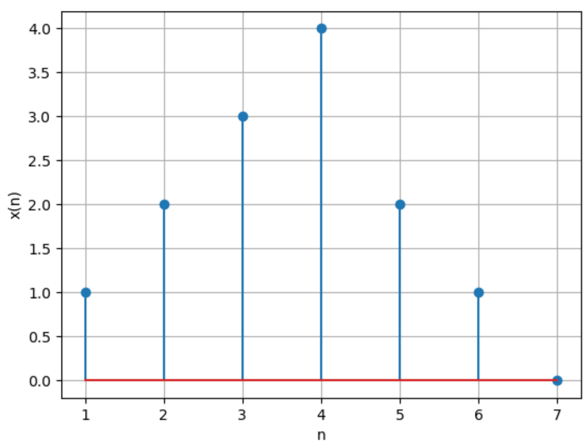

Fig. 3. Plot of $x(n)$ vs n

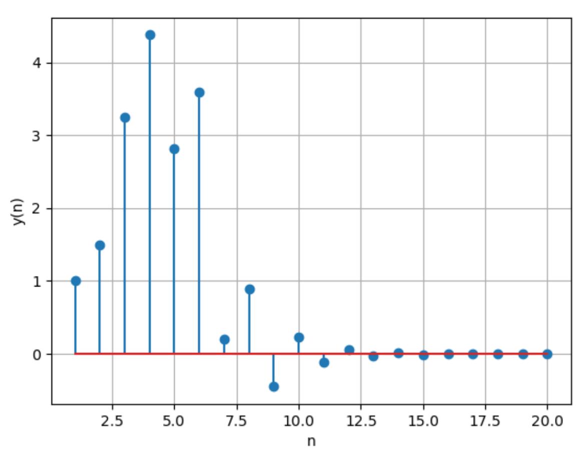

Fig. 4. Plot of $y(n)$ vs n

III. Z-Transform

III.1 The Z-transform of $x(n)$ is defined as

$$X(z) = \mathcal{Z}\{x(n)\} = \sum_{n=-\infty}^{\infty} x(n)z^{-n} \tag{3}$$

Show that

$$\mathcal{Z}\{x(n-1)\} = z^{-1}X(z) \tag{4}$$

and find

$$\mathcal{Z}\{x(n-k)\} \tag{5}$$

Solution: From (3),

$$\mathcal{Z}\{x(n-k)\} = \sum_{n=-\infty}^{\infty} x(n-1)z^{-n} \tag{6}$$

$$= \sum_{n=-\infty}^{\infty} x(n)z^{-n-1} = z^{-1}\sum_{n=-\infty}^{\infty} x(n)z^{-n} \tag{7}$$

$$= z^{-1}X(z) \tag{8}$$

resulting in (4). Similarly, it can be shown that

$$\mathcal{Z}\{x(n-k)\} = z^{-k}X(z) \tag{9}$$

III.2 Find

$$H(z) = \frac{Y(z)}{X(z)} \tag{10}$$

from (2) assuming that the Z-transform is a linear operation.
Solution: Applying (9) in (2),

$$Y(z) + \frac{1}{2}z^{-1}Y(z) = X(z) + z^{-2}X(z) \tag{11}$$

$$\implies \frac{Y(z)}{X(z)} = \frac{1+z^{-2}}{1+\frac{1}{2}z^{-1}} \tag{12}$$

III.3 Find the Z transform of

$$\delta(n) = \begin{cases} 1 & n = 0 \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

and show that the Z-transform of

$$u(n) = \begin{cases} 1 & n \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

is

$$U(z) = \frac{1}{1-z^{-1}}, \quad |z| > 1 \tag{15}$$

Solution: It is easy to show that

$$\delta(n) \overset{\mathcal{Z}}{\longleftrightarrow} 1 \tag{16}$$

and from (14),

$$U(z) = \sum_{n=0}^{\infty} z^{-n} \tag{17}$$

$$= \frac{1}{1-z^{-1}}, \quad |z| > 1 \tag{18}$$

using the formula for the sum of an infinite geometric progression.

III.4 Show that

$$a^n u(n) \overset{\mathcal{Z}}{\longleftrightarrow} \frac{1}{1 - az^{-1}} \quad |z| > |a| \tag{19}$$

Solution:

$$a^n u(n) \overset{\mathcal{Z}}{\longleftrightarrow} \sum_{n=0}^{\infty} \left(a^n z^{-n}\right) \tag{20}$$

$$= \frac{1}{1 - az^{-1}} \quad |z| > |a| \tag{21}$$

III.5 Let

$$H\left(e^{j\omega}\right) = H\left(z = e^{j\omega}\right). \tag{22}$$

Plot $|H\left(e^{j\omega}\right)|$. Comment. $H(e^{j\omega})$ is known as the *Discrete Time Fourier Transform* (DTFT) of $x(n)$.
Solution: The DTFT of the transfer function is plotted using the following code

```
https://github.com/Meowfessor96/
    audio_filtering/blob/main/audio_filtering/
    codes/3.5.py
```

Substituting $z = e^{j\omega}$ in (12), we get

$$\left|H\left(e^{j\omega}\right)\right| = \left|\frac{1 + e^{-2j\omega}}{1 + \frac{1}{2}e^{-j\omega}}\right| \tag{23}$$

$$= \frac{4|\cos\omega|}{\sqrt{5 + 4\cos\omega}} \tag{24}$$

A function $f(x)$ is said to be periodic of time period T if $f(x + nT) = f(x)$, $n \in \mathcal{Z}$

$$\left|H\left(e^{j(\omega+2\pi)}\right)\right| = \frac{4|\cos(\omega + 2\pi)|}{\sqrt{5 + 4\cos(\omega + 2\pi)}} \tag{25}$$

$$= \frac{4|\cos\omega|}{\sqrt{5 + 4\cos\omega}} \tag{26}$$

$$= \left|H\left(e^{j\omega}\right)\right| \tag{27}$$

Therefore its fundamental period is 2π , which verifies that DTFT of a signal is always periodic.

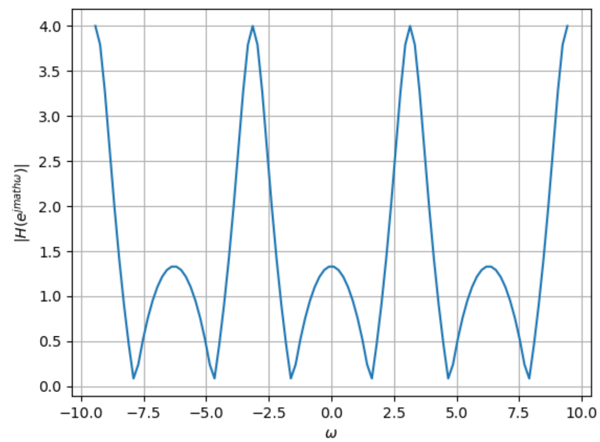

Fig. 5. DTFT, $\left|H\left(e^{j\omega}\right)\right|$

IV. Impulse Response

IV.1 Find an expression for $h(n)$ using $H(z)$, given that

$$h(n) \overset{\mathcal{Z}}{\longleftrightarrow} H(z) \tag{28}$$

and there is a one to one relationship between $h(n)$ and $H(z)$. $h(n)$ is known as the *impulse response* of the system defined by (2).
Solution: From (12),

$$H(z) = \frac{1}{1 + \frac{1}{2}z^{-1}} + \frac{z^{-2}}{1 + \frac{1}{2}z^{-1}} \tag{29}$$

$$\implies h(n) = \left(-\frac{1}{2}\right)^n u(n) + \left(-\frac{1}{2}\right)^{n-2} u(n-2) \tag{30}$$

using (19) and (9).

IV.2 Sketch $h(n)$. Is it bounded? Convergent?
Solution: The following code plots $h(n)$

```
https://github.com/Meowfessor96/
    audio_filtering/blob/main/audio_filtering/
    codes/4.2.py
```

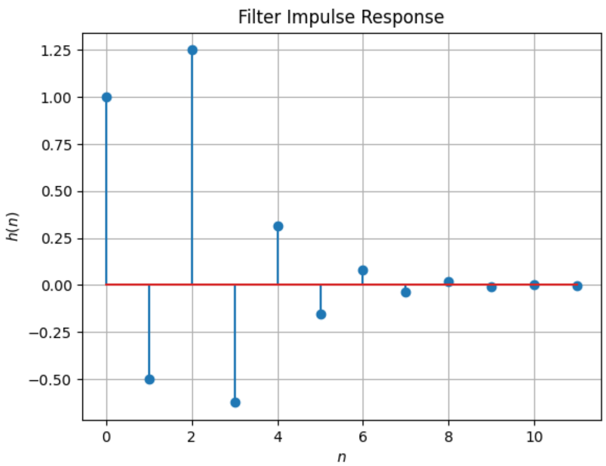

Fig. 6. $h(n)$ as the inverse of $H(z)$

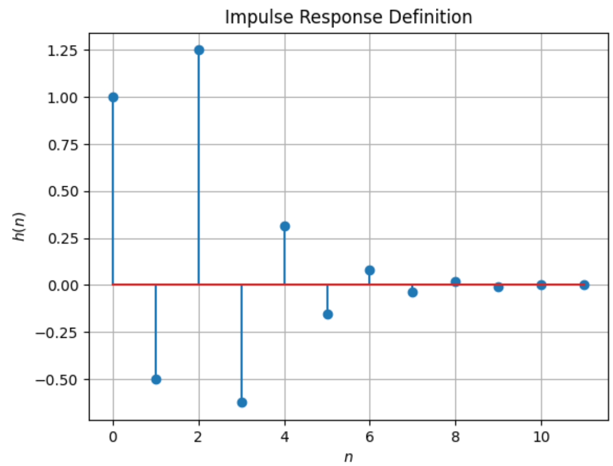

Fig. 7. $h(n)$ from the definition is same as Fig. 6

IV.3 The system with $h(n)$ is defined to be stable if

$$\sum_{n=-\infty}^{\infty} h(n) < \infty \tag{31}$$

Is the system defined by (2) stable for the impulse response in (28)?

Solution: For stable system (31) should converge.

By using ratio test

$$\lim_{n\to\infty} \left| \frac{h(n+1)}{h(n)} \right| < 1 \tag{32}$$

$$\tag{33}$$

For large n

$$u(n) = u(n-2) = 1 \tag{34}$$

$$\lim_{n\to\infty} \left(\frac{h(n+1)}{h(n)} \right) = 1/2 < 1 \tag{35}$$

Therefore it converges. Hence it is stable.

IV.4 Compute and sketch $h(n)$ using

$$h(n) + \frac{1}{2} h(n-1) = \delta(n) + \delta(n-2), \tag{36}$$

This is the definition of $h(n)$.

Solution:

Definition of $h(n)$: The output of the system when $\delta(n)$ is given as input.

The following code plots Fig. 7. Note that this is the same as Fig. 6.

```
https://github.com/Meowfessor96/
    audio_filtering/blob/main/audio_filtering/
    codes/4.4.py
```

IV.5 Compute

$$y(n) = x(n) * h(n) = \sum_{n=-\infty}^{\infty} x(k) h(n-k) \tag{37}$$

Comment. The operation in (37) is known as *convolution*.

Solution: The following code plots Fig. 8. Note that this is the same as $y(n)$ in Fig. **??**.

```
https://github.com/Meowfessor96/
    audio_filtering/blob/main/audio_filtering/
    codes/4.5.py
```

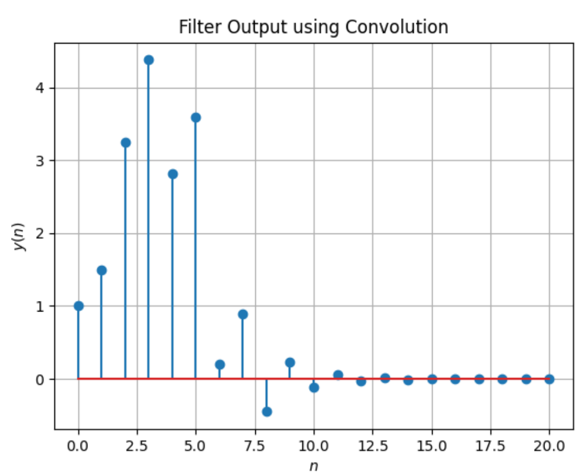

Fig. 8. $y(n)$ from the definition of convolution

IV.6 Show that

$$y(n) = \sum_{n=-\infty}^{\infty} x(n-k)h(k) \tag{38}$$

Solution: In (37), we substitute $k = n - k$ to get

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)\,h(n-k) \tag{39}$$
$$= \sum_{n-k=-\infty}^{\infty} x(n-k)\,h(k) \tag{40}$$
$$= \sum_{k=-\infty}^{\infty} x(n-k)\,h(k) \tag{41}$$

V. DFT and FFT

V.1 Compute

$$X(k) \triangleq \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N}, \quad k = 0, 1, \ldots, N-1 \tag{42}$$

and $H(k)$ using $h(n)$.

V.2 Compute

$$Y(k) = X(k)H(k) \tag{43}$$

V.3 Compute

$$y(n) = \frac{1}{N}\sum_{k=0}^{N-1} Y(k)\cdot e^{j2\pi kn/N}, \quad n = 0, 1, \ldots, N-1 \tag{44}$$

Solution: The above three questions are solved using the code below.

```
https://github.com/Meowfessor96/
    audio_filtering/blob/main/audio_filtering/
    codes/5_sol.py
```

V.4 Repeat the previous exercise by computing $X(k), H(k)$ and $y(n)$ through FFT and IFFT.
Solution: The solution of this question can be found in the code below.

```
https://github.com/Meowfessor96/
    audio_filtering/blob/main/audio_filtering/
    codes/5.4.py
```

This code verifies the result by plotting the obtained result with the result obtained by IDFT.

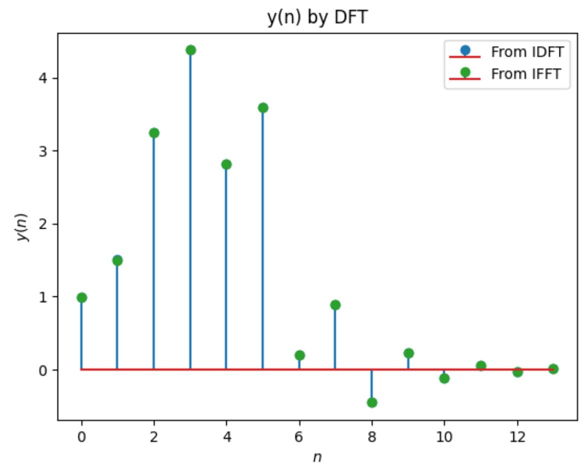

Fig. 9. $y(n)$ obtained from IDFT and IFFT is plotted and verified

V.5 Wherever possible, express all the above equations as matrix equations.
Solution: The DFT matrix is defined as :

$$\mathbf{W} = \begin{pmatrix} \omega^0 & \omega^0 & \ldots & \omega^0 \\ \omega^0 & \omega^1 & \ldots & \omega^{N-1} \\ \vdots & \vdots & \ddots & \vdots \\ \omega^0 & \omega^{N-1} & \ldots & \omega^{(N-1)(N-1)} \end{pmatrix} \tag{45}$$

where $\omega = e^{-\frac{j2\pi}{N}}$. Now any DFT equation can be written as

$$\mathbf{X} = \mathbf{W}\mathbf{x} \tag{46}$$

where

$$\mathbf{x} = \begin{pmatrix} x(0) \\ x(1) \\ \vdots \\ x(n-1) \end{pmatrix} \tag{47}$$

$$\mathbf{X} = \begin{pmatrix} X(0) \\ X(1) \\ \vdots \\ X(n-1) \end{pmatrix} \tag{48}$$

Thus we can rewrite (43) as:

$$\mathbf{Y} = \mathbf{X} \odot \mathbf{H} = (\mathbf{W}\mathbf{x}) \odot (\mathbf{W}\mathbf{h}) \tag{49}$$

where the $\odot$ represents the Hadamard product which performs element-wise multiplication.

The below code computes $y(n)$ by DFT Matrix and then plots it.

```
https://github.com/Meowfessor96/audio_filtering/
    blob/main/audio_filtering/codes/5.5.py
```

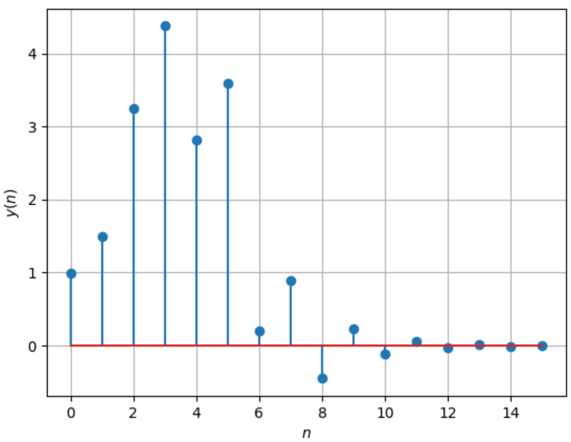

Fig. 10. $y(n)$ obtained from DFT Matrix

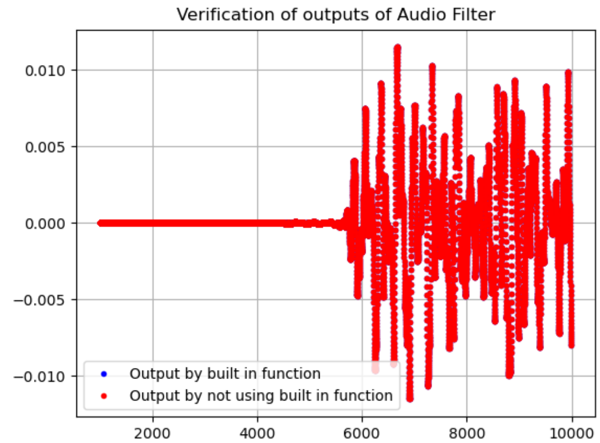

Fig. 11. Both the outputs using and without using function overlap

VI. EXERCISES

Answer the following questions by looking at the python code in Problem I.2.

VI.1 The command

```
output_signal = signal.lfilter(b, a,
    input_signal)
```

in Problem I.2 is executed through the following difference equation

$$\sum_{m=0}^{M} a(m)\, y(n-m) = \sum_{k=0}^{N} b(k)\, x(n-k) \quad (50)$$

where the input signal is $x(n)$ and the output signal is $y(n)$ with initial values all 0. Replace **signal. filtfilt** with your own routine and verify.

Solution: The below code gives the output of an Audio Filter without using the built in function signal.lfilter.

```
https://github.com/Meowfessor96/
    audio_filtering/blob/main/audio_filtering/
    codes/6.1.py
```

VI.2 Repeat all the exercises in the previous sections for the above a and b.

Solution: From the output of a and b from code in Problem I.2, and from 50

$$\begin{aligned} &a(0)\, y(n) + a(1)\, y(n-1) + a(2)\, y(n-2) + a(3) \\ &y(n-3) + a(4)\, y(n-4) = b(0)\, x(n) + b(1)\, x(n-1) \\ &+ b(2)\, x(n-2) + b(3)\, x(n-3) + b(4)\, x(n-4) \end{aligned} \quad (51)$$

Difference Equation is given by :

$$\begin{aligned} &y(n) - (3.66)\, y(n-1) + (5.05)\, y(n-2) \\ &- (3.099)\, y(n-3) + (0.715)\, y(n-4) \\ &= \left(1.45 \times 10^{-5}\right) x(n) + \left(5.74 \times 10^{-5}\right) x(n-1) \\ &+ \left(8.62 \times 10^{-5}\right) x(n-2) + \left(5.74 \times 10^{-5}\right) x(n-3) \\ &+ \left(1.43 \times 10^{-5}\right) x(n-4) \end{aligned} \quad (52)$$

From (50)

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \ldots + b_M z^{-N}}{a_0 + a_1 z^{-1} + a_2 z^{-2} + \ldots + a_N z^{-M}} \quad (53)$$

$$H(z) = \frac{\sum_{k=0}^{N} b(k) z^{-k}}{\sum_{k=0}^{M} a(k) z^{-k}} \quad (54)$$

M = N = 5

Looking at scipy documentation at https://docs.scipy.org/doc/scipy/reference/generated/scipy. the partial fraction on (54) can be generalised as:

$$H(z) = \sum_{i} \frac{r(i)}{1 - p(i) z^{-1}} + \sum_{j}^{k} (j) z^{-j} \quad (55)$$

Whose values can be obtained using the scipy.signal.residuez function. Now,

$$r^n u(n) \stackrel{\mathcal{Z}}{\longleftrightarrow} \frac{1}{1 - rz^{-1}}, \quad |z| > |r| \tag{56}$$

$$\delta(n-k) \stackrel{\mathcal{Z}}{\longleftrightarrow} z^{-k} \tag{57}$$

Taking inverse z transform of (55) by using (56) and (57)

$$h(n) = \sum_i r(i)[p(i)]^n u(n) + \sum_j k(j)\delta(n-j) \tag{58}$$

https://github.com/Meowfessor96/audio_filtering/blob/main/audio_filtering/codes/6.2.py

$r(i)$	$p(i)$	$k(i)$
$0.0590681 - 0.14379042j$	0.8869974 +0.04376584j	2.00×10^{-5}
$0.0590681 + 0.14379042j$	0.8869974 -0.04376584j	–
$-0.05907096 + 0.02466936j$	0.94551962+0.11263131j	–
$-0.05907096 - 0.02466936j$	0.94551962-0.11263131j	–

TABLE 1
VALUES OF $r(i), p(i), k(i)$

Stability of h(n):
According to (31)

$$H(z) = \sum_{n=0}^{\infty} h(n) z^{-n} \tag{59}$$

$$H(1) = \sum_{n=0}^{\infty} h(n) = \frac{\sum_{k=0}^{N} b(k)}{\sum_{k=0}^{M} a(k)} < \infty \tag{60}$$

As both $a(k)$ and $b(k)$ are finite length sequences they converge.
The below code plots Filter frequency response

https://github.com/Meowfessor96/audio_filtering/blob/main/audio_filtering/codes/6_filter_response.py

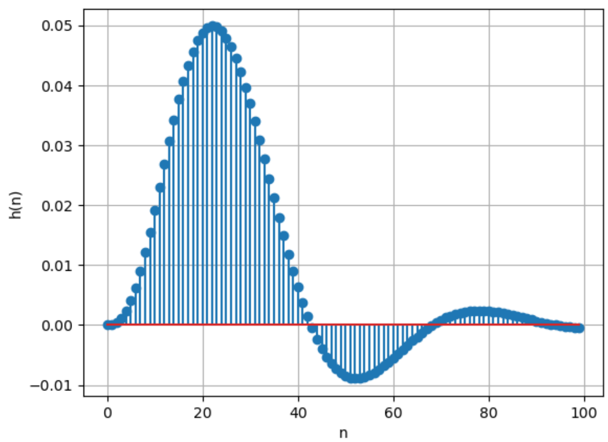

Fig. 12. Plot of h(n)

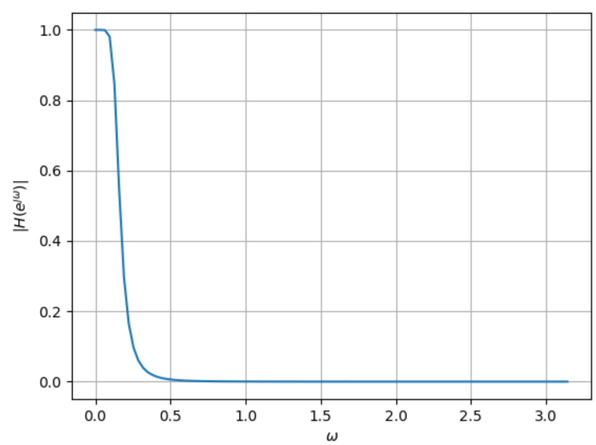

Fig. 13. Frequency Response of Audio Filter

VI.3 What is the sampling frequency of the input signal?
Solution: The Sampling Frequency is 44.1KHz

VI.4 What is type, order and cutoff-frequency of the above butterworth filter
Solution: The given butterworth filter is low-pass with order=4 and cutoff-frequency=1kHz.

VI.5 Modify the code with different input parameters and get the best possible output.
Solution: A better filtering was found on setting the order of the filter to be 5.