Digital Signal Processing

G V V Sharma*

Contents

Abstract—**This manual provides a simple introduction to digital signal processing.**

1 Software Installation

Run the following commands

```
sudo apt-get update
sudo apt-get install libffi-dev libsndfile1 python3
    -scipy python3-numpy python3-matplotlib
sudo pip install cffi pysoundfile
```

2 Digital Filter

2.1 Download the sound file from

```
wget https://raw.githubusercontent.com/
    gadepall/
EE1310/master/filter/codes/Sound_Noise.wav
```

2.2 You will find a spectrogram at https://academo.org/demos/spectrum-analyzer. Upload the sound file that you downloaded in Problem 2.1 in the spectrogram and play. Observe the spectrogram. What do you find?
Solution: There are a lot of yellow lines between 440 Hz to 5.1 KHz. These represent the synthesizer key tones. Also, the key strokes are audible along with background noise.

2.3 Write the python code for removal of out of band noise and execute the code.
Solution:

```
import soundfile as sf
from scipy import signal
#read .wav file
input_signal,fs = sf.read('
    filter_codes_Sound_Noise.wav')
#sampling frequency of Input signal
sampl_freq=fs
#order of the filter
order=4
#cutoff frquency 4kHz
cutoff_freq=4000.0
#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('Sound_With_ReducedNoise.wav',
    output_signal, fs)
```

2.4 The output of the python script in Problem 2.3 is the audio file Sound_With_ReducedNoise.wav. Play the file in the spectrogram in Problem 2.2. What do you observe?
Solution: The key strokes as well as background noise is subdued in the audio. Also, the signal is blank for frequencies above 5.1 kHz.

*The author is with the Department of Electrical Engineering, Indian Institute of Technology, Hyderabad 502285 India e-mail: gadepall@iith.ac.in.

3 Difference Equation

3.1 Let

$$x(n) = \left\{\underset{\uparrow}{1}, 2, 3, 4, 2, 1\right\} \tag{3.1}$$

Sketch $x(n)$.

3.2 Let

$$y(n) + \frac{1}{2}y(n-1) = x(n) + x(n-2),$$
$$y(n) = 0, n < 0 \tag{3.2}$$

Sketch $y(n)$.
Solution: The following code yields Fig. 3.2.

```
wget https://github.com/gadepall/EE1310/raw/
    master/filter/codes/xnyn.py
```

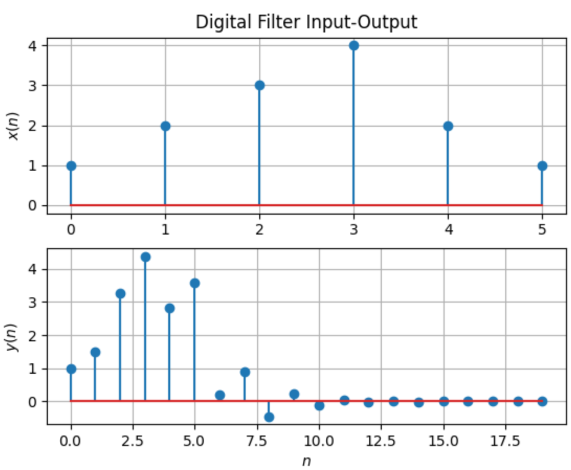

Fig. 3.2

3.3 Repeat the above exercise using a C code.

4 Z-transform

4.1 The Z-transform of $x(n)$ is defined as

$$X(z) = \mathcal{Z}\{x(n)\} = \sum_{n=-\infty}^{\infty} x(n)z^{-n} \tag{4.1}$$

Show that

$$\mathcal{Z}\{x(n-1)\} = z^{-1}X(z) \tag{4.2}$$

and find

$$\mathcal{Z}\{x(n-k)\} \tag{4.3}$$

Solution: From (4.1),

$$\mathcal{Z}\{x(n-k)\} = \sum_{n=-\infty}^{\infty} x(n-1)z^{-n} \tag{4.4}$$
$$= \sum_{n=-\infty}^{\infty} x(n)z^{-n-1} = z^{-1}\sum_{n=-\infty}^{\infty} x(n)z^{-n} \tag{4.5}$$

resulting in (4.2). Similarly, it can be shown that

$$\mathcal{Z}\{x(n-k)\} = z^{-k}X(z) \tag{4.6}$$

4.2 Obtain $X(z)$ for $x(n)$ defined in problem 3.1.
Solution: from 3.1

$$x(n) = \{1, 2, 3, 4, 2, 1\} \tag{4.7}$$
$$X(z) = \mathcal{Z}\{x(n)\} = \sum_{n=-\infty}^{\infty} x(n)z^{-n} \tag{4.8}$$
$$X(z) = 1z^0 + 2z^{-1} + 3z^{-2} + 4z^{-3} + 2z^{-4} + 1z^{-5} \tag{4.9}$$
$$X(z) = 1 + 2z^{-1} + 3z^{-2} + 4z^{-3} + 2z^{-4} + 1z^{-5} \tag{4.10}$$

4.3 Find

$$H(z) = \frac{Y(z)}{X(z)} \tag{4.11}$$

from (3.2) assuming that the Z-transform is a linear operation.
Solution: Applying (4.6) in (3.2),

$$Y(z) + \frac{1}{2}z^{-1}Y(z) = X(z) + z^{-2}X(z) \tag{4.12}$$
$$\implies \frac{Y(z)}{X(z)} = \frac{1 + z^{-2}}{1 + \frac{1}{2}z^{-1}} \tag{4.13}$$

4.4 Find the Z transform of

$$\delta(n) = \begin{cases} 1 & n = 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.14}$$

and show that the Z-transform of

$$u(n) = \begin{cases} 1 & n \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.15}$$

is

$$U(z) = \frac{1}{1 - z^{-1}}, \quad |z| > 1 \tag{4.16}$$

Solution: It is easy to show that

$$\delta(n) \overset{\mathcal{Z}}{\rightleftharpoons} 1 \tag{4.17}$$

and from (4.15),

$$U(z) = \sum_{n=0}^{\infty} z^{-n} \tag{4.18}$$

$$= \frac{1}{1 - z^{-1}}, \quad |z| > 1 \tag{4.19}$$

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

4.5 Show that

$$a^n u(n) \overset{\mathcal{Z}}{\rightleftharpoons} \frac{1}{1 - az^{-1}} \quad |z| > |a| \tag{4.20}$$

Solution:

$$a^n u(n) \overset{\mathcal{Z}}{\rightleftharpoons} \sum_{n=0}^{\infty} \left(az^{-1}\right)^n \tag{4.21}$$

$$= \frac{1}{1 - az^{-1}} \quad |z| > |a| \tag{4.22}$$

4.6 Let

$$H\left(e^{\jmath\omega}\right) = H\left(z = e^{\jmath\omega}\right). \tag{4.23}$$

Plot $\left|H\left(e^{\jmath\omega}\right)\right|$. Comment. $H(e^{\jmath\omega})$ is known as the *Discret Time Fourier Transform* (DTFT) of $x(n)$.

Solution:

$$H\left(e^{\jmath\omega}\right) = \frac{1 + e^{-2\jmath\omega}}{1 + \frac{1}{2}e^{-\jmath\omega}} \tag{4.24}$$

$$\implies \left|H\left(e^{\jmath\omega}\right)\right| = \frac{\left|1 + \cos 2\omega - \jmath \sin 2\omega\right|}{\left|1 + \frac{1}{2}\cos\omega - \frac{1}{2}\sin\omega\right|} \tag{4.25}$$

$$= \sqrt{\frac{(1 + \cos 2\omega)^2 + (\sin 2\omega)^2}{(1 + \frac{1}{2}\cos\omega)^2 + (\frac{1}{2}\sin\omega)^2}} \tag{4.26}$$

$$= \sqrt{\frac{2 + 2\cos 2\omega}{\frac{5}{4} + \cos\omega}} \tag{4.27}$$

$$= \sqrt{\frac{2(2\cos^2\omega)4}{5 + 4\cos\omega}} \tag{4.28}$$

$$\left|H\left(e^{\jmath\omega}\right)\right| = \frac{4\left|\cos\omega\right|}{\sqrt{5 + 4\cos\omega}} \tag{4.29}$$

So,

$$\frac{4\left|\cos(\omega + 2\pi)\right|}{\sqrt{5 + 4\cos(\omega + 2\pi)}} = \frac{4\left|\cos\omega\right|}{\sqrt{5 + 4\cos\omega}} \tag{4.30}$$

It is clear that $\left|H\left(e^{\jmath\omega}\right)\right|$ is periodic with period 2π.

Solution: The following code plots Fig. 4.6.

```
wget https://raw.githubusercontent.com/
    gadepall/EE1310/master/filter/codes/dtft.
    py
```

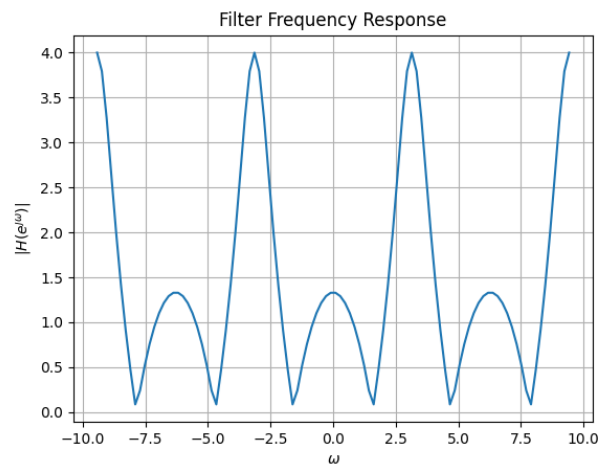

Fig. 4.6: $\left|H\left(e^{\jmath\omega}\right)\right|$
The function $\left|H\left(e^{\jmath\omega}\right)\right|$ is periodic.

4.7 Express $x(n)$ in terms of $H\left(e^{\jmath\omega}\right)$.

Solution:

We have,

$$H(e^{\jmath\omega}) = \sum_{k=-\infty}^{\infty} h(k)e^{-\jmath\omega k} \tag{4.31}$$

However,

$$\int_{-\pi}^{\pi} e^{\jmath\omega(n-k)} d\omega = \begin{cases} 2\pi & n = k \\ 0 & \text{otherwise} \end{cases} \tag{4.32}$$

and so,

$$\frac{1}{2\pi}\int_{-\pi}^{\pi} H(e^{\jmath\omega})e^{j\omega n} d\omega \tag{4.33}$$

$$= \frac{1}{2\pi}\sum_{k=-\infty}^{\infty}\int_{-\pi}^{\pi} h(k)e^{\jmath\omega(n-k)} d\omega \tag{4.34}$$

$$= \frac{1}{2\pi} 2\pi h(n) = h(n) \tag{4.35}$$

which is known as the Inverse Discrete Fourier Transform. Thus,

$$h(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H(e^{\jmath\omega})e^{\jmath\omega n} d\omega \tag{4.36}$$

$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} \frac{1 + e^{-2\jmath\omega}}{1 + \frac{1}{2}e^{-\jmath\omega}} e^{\jmath\omega n} d\omega \tag{4.37}$$

5 Impulse Response

5.1 Using long division, find

$$h(n), n < 5 \tag{5.1}$$

for $H(z)$ in (4.17)
Solution: We substitute $x := z^{-1}$, and perform the long division.
1 + x^{2}1 + $\frac{1}{2}x$
Thus,

$$H(z) = -4 + 2z^{-1} + \frac{5}{1+\frac{1}{2}z^{-1}} \tag{5.2}$$

$$= -4 + 2z^{-1} + 5\sum_{n=0}^{\infty}\left(-\frac{1}{2}\right)^n z^{-n} \tag{5.3}$$

$$= 1 - \frac{1}{2}z^{-1} + 5\sum_{n=2}^{\infty}\left(-\frac{1}{2}\right)^n z^{-n} \tag{5.4}$$

$$= \sum_{n=0}^{\infty}\left(-\frac{1}{2}\right)^n z^{-n} + 4\sum_{n=2}^{\infty}\left(-\frac{1}{2}\right)^n z^{-n} \tag{5.5}$$

$$= \sum_{n=-\infty}^{\infty} u(n)\left(-\frac{1}{2}\right)^n z^{-n} + \sum_{n=-\infty}^{\infty} u(n-2)\left(-\frac{1}{2}\right)^{n-2} z^{-n} \tag{5.6}$$

Therefore, from (4.1),

$$h(n) = \left(-\frac{1}{2}\right)^n u(n) + \left(-\frac{1}{2}\right)^{n-2} u(n-2) \tag{5.7}$$

5.2 Find an expression for $h(n)$ using $H(z)$, given that

$$h(n) \overset{\mathcal{Z}}{\rightleftharpoons} H(z) \tag{5.8}$$

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 (3.2).
Solution: From (4.13),

$$H(z) = \frac{1}{1+\frac{1}{2}z^{-1}} + \frac{z^{-2}}{1+\frac{1}{2}z^{-1}} \tag{5.9}$$

$$\implies h(n) = \left(-\frac{1}{2}\right)^n u(n) + \left(-\frac{1}{2}\right)^{n-2} u(n-2) \tag{5.10}$$

using (4.20) and (4.6).
The ROC will be $(-\infty, 1/2)) \cup (-1/2, \infty)$

5.3 Sketch $h(n)$. Is it bounded? Convergent?
Solution: From (4.13),

$$H(z) = \frac{1}{1+\frac{1}{2}z^{-1}} + \frac{z^{-2}}{1+\frac{1}{2}z^{-1}} \tag{5.11}$$

$$\implies h(n) = \left(-\frac{1}{2}\right)^n u(n) + \left(-\frac{1}{2}\right)^{n-2} u(n-2) \tag{5.12}$$

using (4.20) and (4.6).
Solution: The following code plots Fig. 5.3.

```
wget https://raw.githubusercontent.com/
    gadepall/EE1310/master/filter/codes/hn.py
```

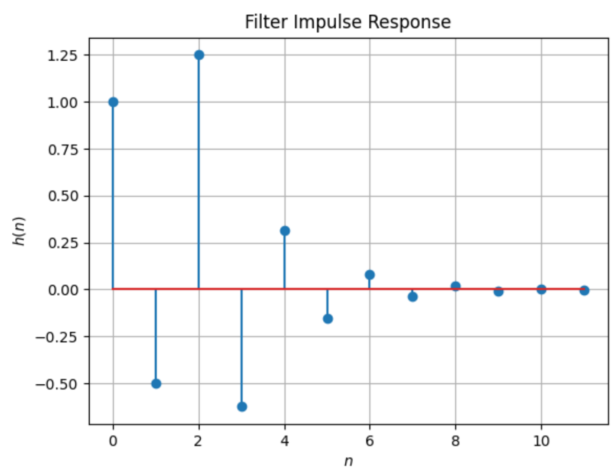

Fig. 5.3: $h(n)$ as the inverse of $H(z)$

5.4 Convergent? Justify using the ratio test.
Solution: for $n > 2$,

$$h(n) = \left(-\frac{1}{2}\right)^n + \left(-\frac{1}{2}\right)^{n-2} \tag{5.13}$$

$$h(n) = 5\left(-\frac{1}{2}\right)^n \tag{5.14}$$

$$\left|\frac{h(n+1)}{h(n)}\right| = \frac{1}{2} < 1 \tag{5.15}$$

Hence, $h(n)$ is convergent.

5.5 The system with $h(n)$ is defined to be stable if

$$\sum_{n=-\infty}^{\infty} h(n) < \infty \tag{5.16}$$

Is the system defined by (3.2) stable for the impulse response in (5.8)?

Solution:

$$|u(n)| \leq 1 \tag{5.17}$$

$$\left|\left(-\frac{1}{2}\right)^n\right| \leq 1 \tag{5.18}$$

$$\implies \left|\left(-\frac{1}{2}\right)^n u(n)\right| \leq 1 \tag{5.19}$$

Similarly,

$$\left|\left(-\frac{1}{2}\right)^{n-2} u(n-2)\right| \leq 1 \tag{5.20}$$

$$\implies h(n) \leq 2 \tag{5.21}$$

Hence, $h(n)$ is bounded.

5.6 Verify result using python code.
Solution:

```
https://github.com/YashSharma/EE3900/
    blob/master/Assignment1/codes/q5/
    q5.6.py
```

We use Toeplitz matrices for convolution

$$\mathbf{y} = \mathbf{x} \circledast \mathbf{h} \tag{5.22}$$

$$\mathbf{y} = \begin{pmatrix} h_1 & 0 & . & . & . & 0 \\ h_2 & h_1 & . & . & . & 0 \\ h_3 & h_2 & h_1 & . & . & 0 \\ . & . & . & . & . & . \\ 0 & . & . & h_3 & h_2 & h_1 \\ 0 & . & . & . & h_2 & h_1 \\ 0 & . & . & . & 0 & h_1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \tag{5.23}$$

5.7 Compute and sketch $h(n)$ using

$$h(n) + \frac{1}{2}h(n-1) = \delta(n) + \delta(n-2), \tag{5.24}$$

This is the definition of $h(n)$.
Solution: for $n > 2$,

$$h(n) = \left(-\frac{1}{2}\right)^n + \left(-\frac{1}{2}\right)^{n-2} \tag{5.25}$$

$$h(n) = 5\left(-\frac{1}{2}\right)^n \tag{5.26}$$

$$\left|\frac{h(n+1)}{h(n)}\right| = \frac{1}{2} < 1 \tag{5.27}$$

Hence, $h(n)$ is convergent. //The following code plots Fig. 5.7. Note that this is the same as Fig. 5.3.

```
wget https://raw.githubusercontent.com/
    gadepall/EE1310/master/filter/codes/hndef
    .py
```

figs/5_5.png

Fig. 5.6: $y(n)$ from the definition

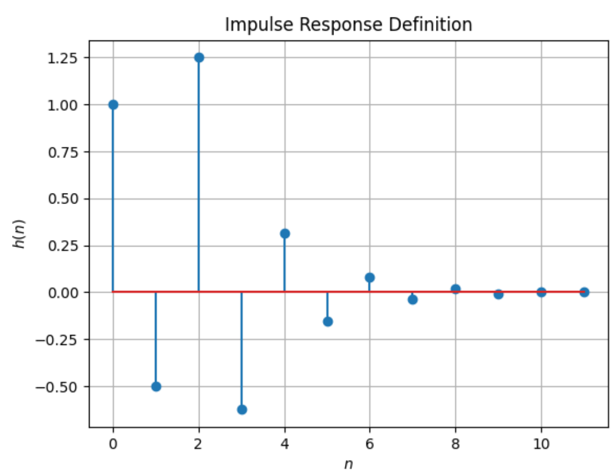

Fig. 5.7: $h(n)$ from the definition

5.8 Compute

$$y(n) = x(n) * h(n) = \sum_{n=-\infty}^{\infty} x(k)h(n-k) \tag{5.28}$$

Comment. The operation in (5.28) is known as *convolution*.

Solution:

$$x(n) * h(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) \tag{5.29}$$

$$= \sum_{k=0}^{5} x(k)h(n-k) \tag{5.30}$$

The following code plots Fig. 5.8. Note that this is the same as $y(n)$ in Fig. **??**.

```
https://github.com/YashSharma/EE3900/
    blob/master/Assignment1/codes/q5/
    ynconv.py
```

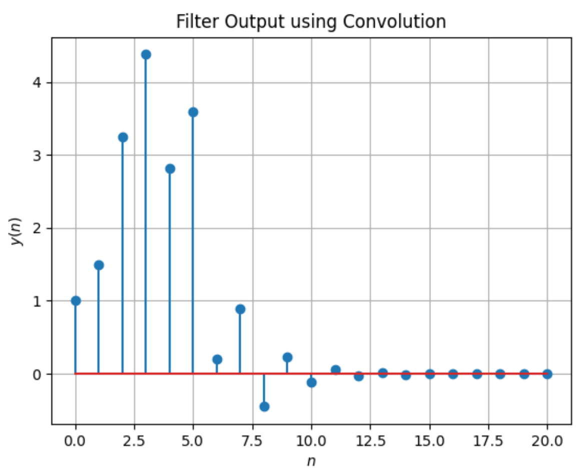

Fig. 5.8: $y(n)$ from the definition of convolution

5.9 Express the above convolution using a Teoplitz matrix. **Solution:**

$$\vec{x} = \begin{pmatrix} 1 & 2 & 3 & 4 & 2 & 1 \end{pmatrix}^\top \tag{5.31}$$

$$\vec{h} = \begin{pmatrix} h_0 & h_1 & \cdots & h_{N-1} \end{pmatrix}^\top \tag{5.32}$$

$$\vec{y} = \vec{x} \circledast \vec{h} \tag{5.33}$$

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N+5} \end{pmatrix} = \begin{pmatrix} h_0 & 0 & 0 & \cdots & 0 \\ h_1 & h_0 & 0 & \cdots & 0 \\ h_2 & h_1 & h_0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{N-1} & h_{N-2} & h_{N-3} & \cdots & h_{N-6} \\ 0 & h_{N-1} & h_{N-2} & \cdots & h_{N-5} \\ 0 & 0 & h_{N-1} & \cdots & h_{N-4} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & h_{N-1} \end{pmatrix} \begin{pmatrix} 1.0 \\ 2.0 \\ 3.0 \\ 4.0 \\ 2.0 \\ 1.0 \end{pmatrix} \tag{5.34}$$

5.10 Show that

$$y(n) = \sum_{n=-\infty}^{\infty} x(n-k)h(k) \tag{5.35}$$

Solution:

$$y(n) = x(n) * h(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) \tag{5.36}$$

Substitute $k = n - i$

$$\sum_{k=-\infty}^{\infty} x(k)h(n-k) = \sum_{n-i=-\infty}^{\infty} x(n-i)h(n-(n-i)) \tag{5.37}$$

$$= \sum_{i=\infty}^{-\infty} x(n-i)h(i) \tag{5.38}$$

$$= \sum_{i=-\infty}^{\infty} x(n-i)h(i) \tag{5.39}$$

since the order of limits does not matter for a summation. Thus,

$$\sum_{k=-\infty}^{\infty} x(k)h(n-k) = \sum_{k=-\infty}^{\infty} x(n-k)h(k) \tag{5.40}$$

$$\implies x(n) * h(n) = h(n) * x(n) \tag{5.41}$$

Therefore, convolution is commutative.

6 DFT and FFT

6.1 Compute

$$X(k) \triangleq \sum_{n=0}^{N-1} x(n)e^{-\jmath 2\pi kn/N}, \quad k = 0, 1, \dots, N-1 \tag{6.1}$$

and $H(k)$ using $h(n)$.
Solution:
The following code plots Fig.6.1

```
https://github.com/sharmayash105/
    EE3900/blob/master/Assignment1/
    codes/q6/6.1.py
```

6.2 Compute

$$Y(k) = X(k)H(k) \tag{6.2}$$

Solution: The following code plots Fig.6.2

```
https://github.com/sharmayash105/
    EE3900/blob/master/Assignment1/
    codes/q6/6.2.py
```

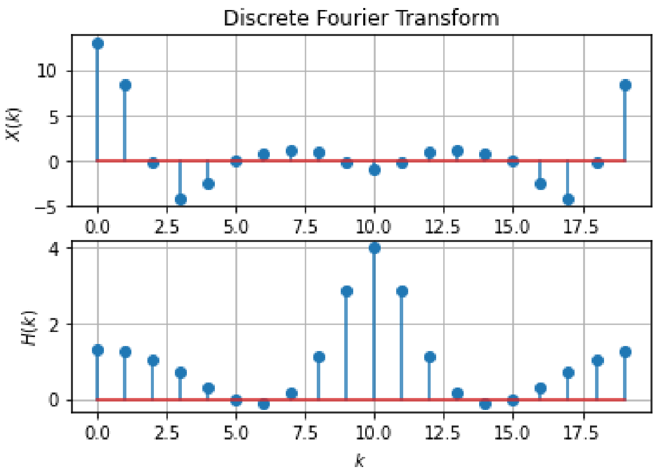

Fig. 6.1: Discret Fourier Transform

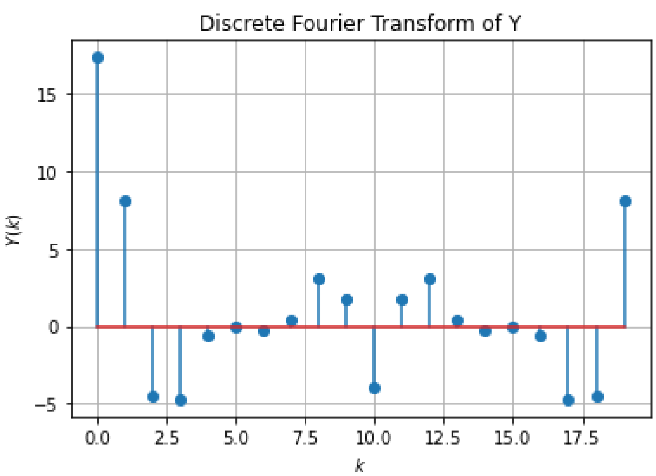

Fig. 6.2: Discret Fourier Transform of $Y(k)$

6.3 Compute

$$y(n) = \frac{1}{N} \sum_{k=0}^{N-1} Y(k) \cdot e^{\jmath 2\pi kn/N}, \quad n = 0, 1, \dots, N-1 \tag{6.3}$$

Solution: The following code plots Fig. 5.8. Note that this is the same as $y(n)$ in Fig. 3.2.

```
wget https://raw.githubusercontent.com/
    gadepall/EE1310/master/filter/codes/yndft.
    py
```

6.4 Repeat the previous exercise by computing $X(k), H(k)$ and $y(n)$ through FFT and IFFT.
Solution: Run the code to generate the Fig. 6.4

```
    code xk
```

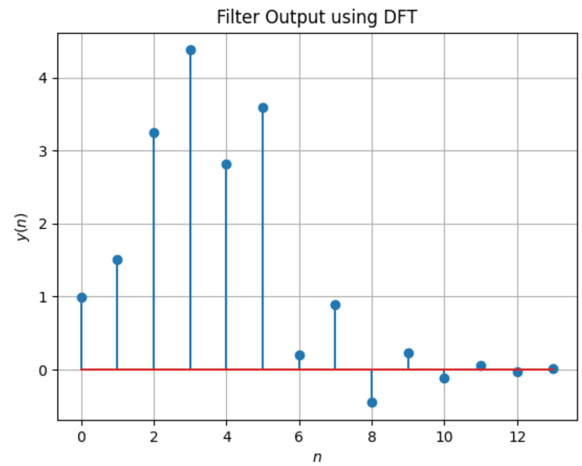

Fig. 6.3: $y(n)$ from the DFT

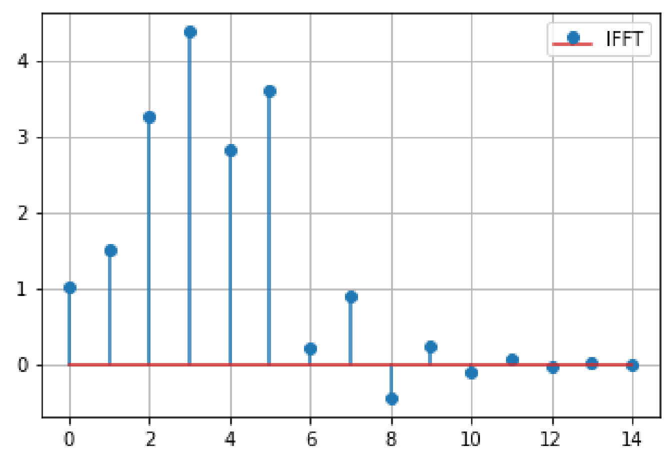

Fig. 6.4: Plot of $y(n)$ by FFT

6.5 Wherever possible, express all the above equations as matrix equations.
Solution:

$$\vec{x} = \begin{pmatrix} x_0 & x_1 & \cdots & x_{N-1} \end{pmatrix}^\top \tag{6.4}$$

$$\vec{h} = \begin{pmatrix} x_0 & x_1 & \cdots & x_{N-1} \end{pmatrix}^\top \tag{6.5}$$

$$\vec{y} = \vec{x} \circledast \vec{h} \tag{6.6}$$

$$\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_{2N-1} \end{pmatrix} = \begin{pmatrix} h_0 & 0 & 0 & \cdots & 0 \\ h_1 & h_0 & 0 & \cdots & 0 \\ h_2 & h_1 & h_0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{N-1} & h_{N-2} & h_{N-3} & \cdots & h_0 \\ 0 & h_{N-1} & h_{N-2} & \cdots & h_1 \\ 0 & 0 & h_{N-1} & \cdots & h_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & h_{N-1} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix} \tag{6.7}$$

The convolution can be written using a Toeplitz matrix.
Consider the DFT matrix

$$\vec{W} = \begin{pmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \omega^3 & \cdots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 & \omega^6 & \cdots & \omega^{2(N-1)} \\ 1 & \omega^3 & \omega^6 & \omega^9 & \cdots & \omega^{3(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{N-1} & \omega^{2(N-1)} & \omega^{3(N-1)} & \cdots & \omega^{(N-1)(N-1)} \end{pmatrix} \tag{6.8}$$

where $\omega = e^{-\jmath 2\pi/N}$ is the N^{th} root of unity
Then the discrete Fourier transforms of $\vec{x}$ and $\vec{h}$ are given by

$$\vec{X} = \vec{W}\vec{x} \tag{6.9}$$

$$\vec{H} = \vec{W}\vec{h} \tag{6.10}$$

$\vec{Y}$ is then given by

$$\vec{Y} = \vec{X} \circ \vec{H} \tag{6.11}$$

where $\circ$ denotes the Hadamard product (element-wise multiplication)
But $\vec{Y}$ is the discrete Fourier transform of the filter output $\vec{y}$

$$\vec{Y} = \vec{W}\vec{y} \tag{6.12}$$

Thus,

$$\vec{W}\vec{y} = \vec{X} \circ \vec{H} \tag{6.13}$$

$$\implies \vec{y} = \vec{W}^{-1}\left(\vec{X} \circ \vec{H}\right) \tag{6.14}$$

$$= \vec{W}^{-1}\left(\vec{W}\vec{x} \circ \vec{W}\vec{h}\right) \tag{6.15}$$

This is the inverse discrete Fourier transform of $\vec{Y}$

6.6 Verify the above equations by generating the DFT matrix in Python.
Solution: Download the following python code that plots Fig.6.6

code of 6.6

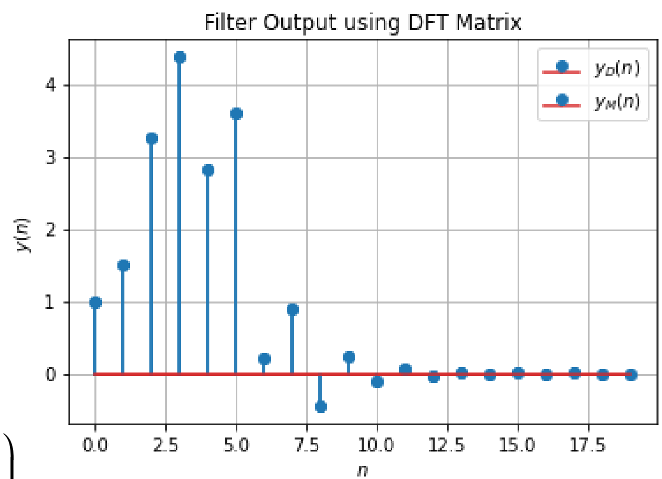

Fig. 6.6: Filter output using DFT matrix

6.7 Find time complexities of computing $y(n)$ using FFT/IFFT and convolution.
Solution: The C code for finding running time of these algorithm.

6.7 c code

This code generates text files that used to plot runtimes of algorithms in following python codes.

6.7.1.py 6.7.2.py

7 FFT

7.1 The DFT of $x(n)$ is given by

$$X(k) \triangleq \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N}, \quad k = 0, 1, \ldots, N-1 \tag{7.1}$$

7.2 Let

$$W_N = e^{-j2\pi/N} \tag{7.2}$$

Then the N-point *DFT matrix* is defined as

$$\vec{F}_N = [W_N^{mn}], \quad 0 \le m, n \le N-1 \tag{7.3}$$

where W_N^{mn} are the elements of $\vec{F}_N$.

7.3 Let

$$\vec{I}_4 = \begin{pmatrix} \vec{e}_4^1 & \vec{e}_4^2 & \vec{e}_4^3 & \vec{e}_4^4 \end{pmatrix} \tag{7.4}$$

be the 4×4 identity matrix. Then the 4 point *DFT permutation matrix* is defined as

$$\vec{P}_4 = \begin{pmatrix} \vec{e}_4^1 & \vec{e}_4^3 & \vec{e}_4^2 & \vec{e}_4^4 \end{pmatrix} \tag{7.5}$$

7.4 The 4 point *DFT diagonal matrix* is defined as

$$\vec{D}_4 = diag\begin{pmatrix} W_8^0 & W_8^1 & W_8^2 & W_8^3 \end{pmatrix} \tag{7.6}$$

7.5 Show that

$$W_N^2 = W_{N/2} \tag{7.7}$$

Solution:

$$W_N = e^{-j2\pi/N} \tag{7.8}$$

$$W_{N/2} = e^{-j2\pi*2/N} \tag{7.9}$$

$$W_{N/2} = \left(e^{-j2\pi/N}\right)^2 \tag{7.10}$$

$$W_{N/2} = W_{N/2}^2 \tag{7.11}$$

$$W_N^2 = W_{N/2} \tag{7.12}$$

7.6 Show that

$$\vec{F}_4 = \begin{bmatrix} \vec{I}_2 & \vec{D}_2 \\ \vec{I}_2 & -\vec{D}_2 \end{bmatrix} \begin{bmatrix} \vec{F}_2 & 0 \\ 0 & \vec{F}_2 \end{bmatrix} \vec{P}_4 \tag{7.13}$$

solution Observe that for $n \in \mathbb{N}$, $W_4^{4n} = 1$ and $W_4^{4n+2} = -1$. Using (7.7),

$$\vec{D}_2\vec{F}_2 = \begin{bmatrix} W_4^0 & 0 \\ 0 & W_4^1 \end{bmatrix} \begin{bmatrix} W_2^0 & W_2^0 \\ W_2^0 & W_2^1 \end{bmatrix} \tag{7.14}$$

$$= \begin{bmatrix} W_4^0 & 0 \\ 0 & W_4^1 \end{bmatrix} \begin{bmatrix} W_4^0 & W_4^0 \\ W_4^0 & W_4^2 \end{bmatrix} \tag{7.15}$$

$$= \begin{bmatrix} W_4^0 & W_4^0 \\ W_4^1 & W_4^3 \end{bmatrix} \tag{7.16}$$

$$\implies -\vec{D}_2\vec{F}_2 = \begin{bmatrix} W_4^2 & W_4^6 \\ W_4^3 & W_4^9 \end{bmatrix} \tag{7.17}$$

and

$$\vec{F}_2 = \begin{pmatrix} W_2^0 & W_2^0 \\ W_2^0 & W_2^1 \end{pmatrix} \tag{7.18}$$

$$= \begin{pmatrix} W_4^0 & W_4^0 \\ W_4^0 & W_4^2 \end{pmatrix} \tag{7.19}$$

Hence,

$$\vec{W}_4 = \begin{pmatrix} W_4^0 & W_4^0 & W_4^0 & W_4^0 \\ W_4^0 & W_4^2 & W_4^1 & W_4^3 \\ W_4^0 & W_4^4 & W_4^2 & W_4^6 \\ W_4^0 & W_4^6 & W_4^3 & W_4^9 \end{pmatrix} \tag{7.20}$$

$$= \begin{bmatrix} \vec{I}_2\vec{F}_2 & \vec{D}_2F_2 \\ \vec{I}_2\vec{F}_2 & -\vec{D}_2F_2 \end{bmatrix} \tag{7.21}$$

$$= \begin{bmatrix} \vec{I}_2 & \vec{D}_2 \\ \vec{I}_2 & \vec{D}_2 \end{bmatrix} \begin{bmatrix} \vec{F}_2 & 0 \\ 0 & \vec{F}_2 \end{bmatrix} \tag{7.22}$$

Multiplying (7.22) by $\vec{P}_4$ on both sides, and noting that $\vec{W}_4\vec{P}_4 = \vec{F}_4$ gives us (**??**).

7.7 Show that

$$\vec{F}_N = \begin{bmatrix} \vec{I}_{N/2} & \vec{D}_{N/2} \\ \vec{I}_{N/2} & -\vec{D}_{N/2} \end{bmatrix} \begin{bmatrix} \vec{F}_{N/2} & 0 \\ 0 & \vec{F}_{N/2} \end{bmatrix} \vec{P}_N \tag{7.23}$$

Solution: Observe that for even N and letting $\vec{f}_N^i$ denote the i^{th} column of $\vec{F}_N$, from (7.16) and (7.17),

$$\begin{pmatrix} \vec{D}_{N/2}\vec{F}_{N/2} \\ -\vec{D}_{N/2}\vec{F}_{N/2} \end{pmatrix} = \begin{pmatrix} \vec{f}_N^2 & \vec{f}_N^4 & \dots & \vec{f}_N^N \end{pmatrix} \tag{7.24}$$

and

$$\begin{pmatrix} \vec{I}_{N/2}\vec{F}_{N/2} \\ \vec{I}_{N/2}\vec{F}_{N/2} \end{pmatrix} = \begin{pmatrix} \vec{f}_N^1 & \vec{f}_N^3 & \dots & \vec{f}_N^{N-1} \end{pmatrix} \tag{7.25}$$

Thus,

$$\begin{bmatrix} \vec{I}_2\vec{F}_2 & \vec{D}_2\vec{F}_2 \\ \vec{I}_2\vec{F}_2 & -\vec{D}_2\vec{F}_2 \end{bmatrix} = \begin{bmatrix} \vec{I}_{N/2} & \vec{D}_{N/2} \\ \vec{I}_{N/2} & -\vec{D}_{N/2} \end{bmatrix} \begin{bmatrix} \vec{F}_{N/2} & 0 \\ 0 & \vec{F}_{N/2} \end{bmatrix}$$

$$= \begin{pmatrix} \vec{f}_N^1 & \dots & \vec{f}_N^{N-1} & \vec{f}_N^2 & \dots & \vec{f}_N^N \end{pmatrix} \tag{7.26}$$

and so,

$$\begin{bmatrix} \vec{I}_{N/2} & \vec{D}_{N/2} \\ \vec{I}_{N/2} & -\vec{D}_{N/2} \end{bmatrix} \begin{bmatrix} \vec{F}_{N/2} & 0 \\ 0 & \vec{F}_{N/2} \end{bmatrix} \vec{P}_N$$

$$= \begin{pmatrix} \vec{f}_N^1 & \vec{f}_N^2 & \dots & \vec{f}_N^N \end{pmatrix} = \vec{F}_N \tag{7.27}$$

7.8 Find

$$\vec{P}_4\vec{x} \tag{7.28}$$

Solution: We have,

$$\vec{P}_4\vec{x} = \begin{pmatrix} \vec{e}_4^1 & \vec{e}_4^3 & \vec{e}_4^2 & \vec{e}_4^4 \end{pmatrix} \begin{pmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \end{pmatrix} = \begin{pmatrix} x(0) \\ x(2) \\ x(1) \\ x(3) \end{pmatrix} \tag{7.29}$$

7.9 Show that

$$\vec{X} = \vec{F}_N\vec{x} \tag{7.30}$$

where $\vec{x}, \vec{X}$ are the vector representations of $x(n), X(k)$ respectively.

Solution: Writing the terms of X,

$$X(0) = x(0) + x(1) + \ldots + x(N-1) \tag{7.31}$$

$$X(1) = x(0) + x(1)e^{-\frac{\jmath 2\pi}{N}} + \ldots + + x(N-1)e^{-\frac{\jmath 2(N-1)\pi}{N}} \tag{7.32}$$

$$\vdots$$

$$X(N-1) = x(0) + x(1)e^{-\frac{\jmath 2(N-1)\pi}{N}} + \ldots + + x(N-1)e^{-\frac{\jmath 2(N-1)(N-1)\pi}{N}} \tag{7.33}$$

Clearly, the term in the m^{th} row and n^{th} column is given by $(0 \le m \le N-1$ and $0 \le n \le N-1)$

$$T_{mn} = x(n)e^{-\frac{\jmath 2mn\pi}{N}} \tag{7.34}$$

and so, we can represent each of these terms as a matrix product

$$\vec{X} = \vec{F}_N\vec{x} \tag{7.35}$$

where $\vec{F}_N = \left[e^{-\frac{-\jmath 2mn\pi}{N}}\right]_{mn}$ for $0 \le m \le N-1$ and $0 \le n \le N-1$.

7.10 Let

$$\begin{pmatrix} X_1(0) \\ X_1(1) \\ X_1(2) \end{pmatrix} = F_3 \begin{pmatrix} x(0) \\ x(2) \\ x(4) \end{pmatrix} \tag{7.36}$$

$$\begin{pmatrix} X_2(0) \\ X_2(1) \\ X_2(2) \end{pmatrix} = F_3 \begin{pmatrix} x(1) \\ x(3) \\ x(5) \end{pmatrix} \tag{7.37}$$

Show that

$$\begin{pmatrix} X(0) \\ X(1) \\ X(2) \end{pmatrix} = \begin{pmatrix} X_1(0) \\ X_1(1) \\ X_1(2) \end{pmatrix} + \begin{pmatrix} W_6^0 & 0 & 0 \\ 0 & W_6^1 & 0 \\ 0 & 0 & W_6^2 \end{pmatrix} \begin{pmatrix} X_2(0) \\ X_2(1) \\ X_2(2) \end{pmatrix} \tag{7.38}$$

$$\begin{pmatrix} X(3) \\ X(4) \\ X(5) \end{pmatrix} = \begin{pmatrix} X_1(0) \\ X_1(1) \\ X_1(2) \end{pmatrix} - \begin{pmatrix} W_6^0 & 0 & 0 \\ 0 & W_6^1 & 0 \\ 0 & 0 & W_6^2 \end{pmatrix} \begin{pmatrix} X_2(0) \\ X_2(1) \\ X_2(2) \end{pmatrix} \tag{7.39}$$

Solution: We write out the values of performing an 8-point FFT on $\vec{x}$ as follows.

$$X(k) = \sum_{n=0}^{7} x(n)e^{-\frac{\jmath 2kn\pi}{8}} \tag{7.40}$$

$$= \sum_{n=0}^{3} \left(x(2n)e^{-\frac{\jmath 2kn\pi}{4}} + e^{-\frac{\jmath 2k\pi}{8}} x(2n+1)e^{-\frac{\jmath 2kn\pi}{4}}\right) \tag{7.41}$$

$$= X_1(k) + e^{-\frac{\jmath 2k\pi}{4}} X_2(k) \tag{7.42}$$

where $\vec{X}_1$ is the 4-point FFT of the even-numbered terms and $\vec{X}_2$ is the 4-point FFT of the odd numbered terms. Noticing that for $k \ge 4$,

$$X_1(k) = X_1(k-4) \tag{7.43}$$

$$e^{-\frac{\jmath 2k\pi}{8}} = -e^{-\frac{\jmath 2(k-4)\pi}{8}} \tag{7.44}$$

we can now write out $X(k)$ in matrix form as in (**??**) and (**??**). We also need to solve the two 4-point FFT terms so formed.

$$X_1(k) = \sum_{n=0}^{3} x_1(n)e^{-\frac{\jmath 2kn\pi}{8}} \tag{7.45}$$

$$= \sum_{n=0}^{1} \left(x_1(2n)e^{-\frac{\jmath 2kn\pi}{4}} + e^{-\frac{\jmath 2k\pi}{8}} x_2(2n+1)e^{-\frac{\jmath 2kn\pi}{4}}\right) \tag{7.46}$$

$$= X_3(k) + e^{-\frac{\jmath 2k\pi}{4}} X_4(k) \tag{7.47}$$

using $x_1(n) = x(2n)$ and $x_2(n) = x(2n+1)$. Thus we can write the 2-point FFTs

$$\begin{bmatrix} X_3(0) \\ X_3(1) \end{bmatrix} = F_2 \begin{bmatrix} x(0) \\ x(4) \end{bmatrix} \tag{7.48}$$

$$\begin{bmatrix} X_4(0) \\ X_4(1) \end{bmatrix} = F_2 \begin{bmatrix} x(2) \\ x(6) \end{bmatrix} \tag{7.49}$$

Using a similar idea for the terms X_2,

$$\begin{bmatrix} X_5(0) \\ X_5(1) \end{bmatrix} = F_2 \begin{bmatrix} x(1) \\ x(5) \end{bmatrix} \tag{7.50}$$

$$\begin{bmatrix} X_6(0) \\ X_6(1) \end{bmatrix} = F_2 \begin{bmatrix} x(3) \\ x(7) \end{bmatrix} \tag{7.51}$$

But observe that from (7.29),

$$\vec{P}_8\vec{x} = \begin{pmatrix}\vec{x}_1\\ \vec{x}_2\end{pmatrix} \tag{7.52}$$

$$\vec{P}_4\vec{x}_1 = \begin{pmatrix}\vec{x}_3\\ \vec{x}_4\end{pmatrix} \tag{7.53}$$

$$\vec{P}_4\vec{x}_2 = \begin{pmatrix}\vec{x}_5\\ \vec{x}_6\end{pmatrix} \tag{7.54}$$

where we define $x_3(k) = x(4k)$, $x_4(k) = x(4k + 2)$, $x_5(k) = x(4k + 1)$, and $x_6(k) = x(4k + 3)$ for $k = 0, 1$.

7.11 For

$$\vec{x} = \begin{pmatrix}1\\2\\3\\4\\2\\1\end{pmatrix} \tag{7.55}$$

compte the DFT using (7.30) **Solution:** Download the Python code from

```
7.11.py
```

and run it using

```
$ python3 7_11.py
```

7.12 Repeat the above exercise using (7.39)

7.13 Write a C program to compute the 8-point FFT. **Solution:** The C code for the above two problems can be downloaded from

```
7.13.c
```

Compile and run the code using

```
$ gcc −lm −Wall −O2 7_13.c
$ ./a.out
```

8 Exercises

Answer the following questions by looking at the python code in Problem 2.3.

8.1 The command

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

in Problem 2.3 is executed through the following difference equation

$$\sum_{m=0}^{M} a(m)\, y(n-m) = \sum_{k=0}^{N} b(k)\, x(n-k) \tag{8.1}$$

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:** Download the source code by typing the next command

```
$ wget https://raw.githubusercontent.com/
    goats−9/ee3900−assignments/main/filter/
    codes/8_1.py
```

and run it using

```
$ python3 8_1.py
```

8.2 Repeat all the exercises in the previous sections for the above a and b.
Solution: For the given values, the difference equation is

$$\begin{aligned}
&y(n) - (2.52)\, y(n-1) + (2.56)\, y(n-2)\\
&- (1.21)\, y(n-3) + (0.22)\, y(n-4)\\
&= \left(3.45 \times 10^{-3}\right) x(n) + \left(1.38 \times 10^{-2}\right) x(n-1)\\
&+ \left(2.07 \times 10^{-2}\right) x(n-2) + \left(1.38 \times 10^{-2}\right) x(n-3)\\
&+ \left(3.45 \times 10^{-3}\right) x(n-4)
\end{aligned} \tag{8.2}$$

From (8.1), we see that the transfer function can be written as follows

$$H(z) = \frac{\sum_{k=0}^{N} b(k) z^{-k}}{\sum_{k=0}^{M} a(k) z^{-k}} \tag{8.3}$$

$$= \sum_i \frac{r(i)}{1 - p(i) z^{-1}} + \sum_j k(j) z^{-j} \tag{8.4}$$

where $r(i)$, $p(i)$, are called residues and poles respectively of the partial fraction expansion of $H(z)$. $k(i)$ are the coefficients of the direct polynomial terms that might be left over. We can now take the inverse z-transform of (8.4) and get using (4.20),

$$h(n) = \sum_i r(i)[p(i)]^n u(n) + \sum_j k(j)\delta(n-j) \tag{8.5}$$

Substituting the values,

$$h(n) = [(-0.24 - 0.71\jmath)(0.56 + 0.14\jmath)^n + (-0.24 + 0.71\jmath)(0.56 - 0.14\jmath)^n + (-0.25 + 0.12\jmath)(0.70 + 0.41\jmath)^n + (-0.25 - 0.12\jmath)(0.70 - 0.41\jmath)^n]u(n) + \left(1.6 \times 10^{-2}\right)\delta(n) \tag{8.6}$$

$$\implies h(n) = (1.5)(0.58)^n \cos(n\alpha_1 + \beta_1) + (0.55)(0.81)^n \cos(n\alpha_2 + \beta_2) + \left(1.6 \times 10^{-2}\right)\delta(n) \tag{8.7}$$

where

$$\tan\alpha_1 = 0.25 \tag{8.8}$$

$$\tan\beta_1 = 2.96 \tag{8.9}$$

$$\tan\alpha_2 = 0.59 \tag{8.10}$$

$$\tan\beta_2 = -0.48 \tag{8.11}$$

The values $r(i)$, $p(i)$, $k(i)$ and thus the impulse response function are computed and plotted at

```
$ wget https://raw.githubusercontent.com/
    goats-9/ee3900-assignments/main/filter/
    codes/8_2_1.py
```

The filter frequency response is plotted at

```
$ wget https://raw.githubusercontent.com/
    goats-9/ee3900-assignments/main/filter/
    codes/8_2_2.py
```

Observe that for a series $t_n = r^n$, $\frac{t_{n+1}}{t_n} = r$. By the ratio test, t_n converges if $|r| < 1$. We observe that for all i, $|p(i)| < 1$ and so, as $h(n)$ is the sum of many convergent series, we see that $h(n)$ converges and is bounded. From (4.1),

$$\sum_{n=0}^{\infty} h(n) = H(1) = \frac{\sum_{k=0}^{N} b(k)}{\sum_{k=0}^{M} a(k)} = 1 < \infty \tag{8.12}$$

Therefore, the system is stable. From Fig. (**??**), $h(n)$ is negligible after $n \geq 64$, and we can apply a 64-bit FFT to get y(n). The following code uses the DFT matrix to generate $y(n)$ in Fig. (**??**).

```
$ wget https://raw.githubusercontent.com/
    goats-9/ee3900-assignments/main/filter/
    codes/8_2_3.py
```

The codes can be run all at once by typing a small shell script

```
$ for file in 8_2_*.py; do python ${file};
    done
```

8.3 What is the sampling frequency of the input signal?
Solution: Sampling frequency $f_s = 44.1$ kHZ.

8.4 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 4 kHz.

8.5 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 7.