Digital Signal Processing

Ravula Karthik

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://github.com/karthik6281/Signal−
    Processing/blob/main/sig−pro/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:

```python
import soundfile as sf
from scipy import signal

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

3 Difference Equation

3.1 Let

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

Sketch $x(n)$.
Solution: The following code yields Fig. 3.1.

```
wget https://github.com/karthik6281/Signal-Processing/blob/main/sig-pro/codes/3_1.py
```

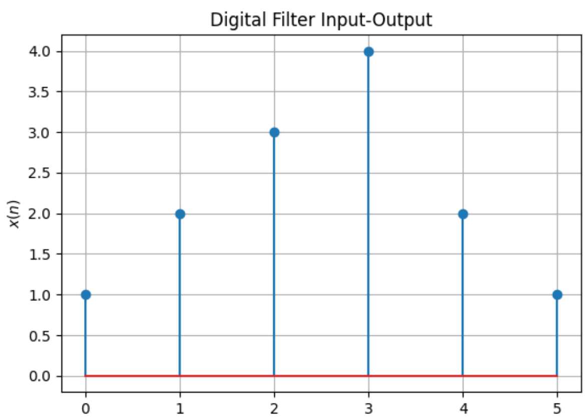

Fig. 3.1

3.2 Let

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

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

```
wget https://github.com/karthik6281/Signal-Processing/blob/main/sig-pro/codes/3_2.py
```

3.3 Repeat the above exercise using a C code.
Solution: The c code can be obtained from

```
wget https://github.com/karthik6281/Signal-Processing/blob/main/sig-pro/codes/3_3.c
```

4 Z-transform

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

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

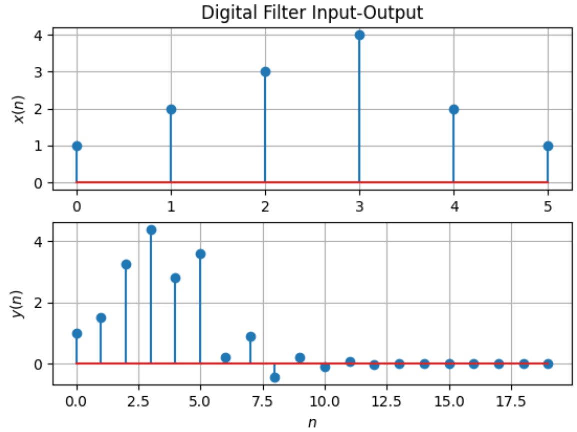

Fig. 3.2

Show that

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

and find

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

Solution: From (4.1),

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

$$= \sum_{n=-\infty}^{\infty} x(n)z^{-n-1} = z^{-1} \sum_{n=-\infty}^{\infty} x(n)z^{-n} \quad (4.5)$$

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

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

4.2 Obtain $X(z)$ for $x(n)$ defined in problem **??**.
Solution: Z-transform of x(n),$X(z)$ is given by

$$\mathcal{Z}\{x(n)\} = \sum_{n=-\infty}^{\infty} x(n)z^{-n} \quad (4.7)$$

$$= \sum_{n=0}^{5} x(n)z^{-n} \quad (4.8)$$

$$= 1 + 2z^{-1} + 3z^{-2} + 4z^{-3} + 2z^{-4} + z^{-5} \quad (4.9)$$

4.3 Find

$$H(z) = \frac{Y(z)}{X(z)} \quad (4.10)$$

from (3.2) assuming that the Z-transform is a linear operation.

Solution: Applying (4.6) in (3.2),

$$Y(z) + \frac{1}{2}z^{-1}Y(z) = X(z) + z^{-2}X(z) \tag{4.11}$$

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

4.4 Find the Z transform of

$$\delta(n) = \begin{cases} 1 & n = 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.13}$$

and show that the Z-transform of

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

is

$$U(z) = \frac{1}{1 - z^{-1}}, \quad |z| > 1 \tag{4.15}$$

Solution: The Z-transform of $\delta(n)$ is defined as

$$\mathcal{Z}\{\delta(n)\} = \sum_{n=-\infty}^{\infty} \delta(n)z^{-n} \tag{4.16}$$

$$= \delta(0)z^{-0} \tag{4.17}$$

$$= 1 \tag{4.18}$$

Hence we can say that

$$\delta(n) \overset{\mathcal{Z}}{\rightleftharpoons} 1 \tag{4.19}$$

and from (4.14),

$$U(z) = \sum_{n=0}^{\infty} z^{-n} \tag{4.20}$$

$$= \frac{1}{1 - z^{-1}}, \quad |z| > 1 \tag{4.21}$$

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

4.5 Show that

$$a^n u(n) \overset{\mathcal{Z}}{\rightleftharpoons} \frac{1}{1 - az^{-1}} \quad |z| > |a| \tag{4.22}$$

Solution:

$$\mathcal{Z}\{a^n u(n)\} = \sum_{n=-\infty}^{\infty} a^n u(n)z^{-n} \tag{4.23}$$

$$= \sum_{n=0}^{\infty} a^n z^{-n} \tag{4.24}$$

$$= \sum_{n=0}^{\infty} (z^{-1}a)^n \tag{4.25}$$

$$= \frac{1}{1 - az^{-1}}, \quad \left|z^{-1}a\right| < 1 \tag{4.26}$$

$$= \frac{1}{1 - az^{-1}}, \quad |z| > |a| \tag{4.27}$$

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

4.6 Let

$$H\left(e^{\jmath\omega}\right) = H\left(z = e^{\jmath\omega}\right). \tag{4.28}$$

Plot $\left|H\left(e^{\jmath\omega}\right)\right|$. Is it periodic? If so, find the period. $H(e^{\jmath\omega})$ is known as the *Discret Time Fourier Transform* (DTFT) of $h(n)$.

Solution: $H(e^{jw})$ is given by

$$H(e^{jw}) = \frac{1 + (e^{jw})^{-2}}{1 + \frac{1}{2}(e^{jw})^{-1}} \tag{4.29}$$

$$= 2\frac{1 + \cos(-2\omega) + j\sin(-2\omega)}{2 + \cos(-\omega) + j\sin(-\omega)} \tag{4.30}$$

$$= 2\frac{1 + \cos(2\omega) - j\sin(2\omega)}{2 + \cos(\omega) - j\sin(\omega)} \tag{4.31}$$

$$= 2\frac{2\cos^2(\omega) - 2j\sin(\omega)\cos(\omega)}{2 + \cos(\omega) - j\sin(\omega)} \tag{4.32}$$

$$= 4\cos(\omega)\frac{\cos(\omega) - j\sin(\omega)}{2 + \cos(\omega) - j\sin(\omega)} \tag{4.33}$$

$$= 4|\cos(\omega)|\frac{e^{jw}}{2 + e^{jw}} \tag{4.34}$$

So,

$$|H(e^{jw})| = 4|\cos(\omega)|\frac{|e^{jw}|}{|2 + e^{jw}|} \tag{4.35}$$

$$= \frac{|4\cos(\omega)|}{5 + 4\cos(\omega)} \tag{4.36}$$

$\left|H\left(e^{\jmath\omega}\right)\right|$ is periodic with period π.(The LCM of the period of $|\cos(\omega)|$ and $5 + 4\cos(\omega)$ is 2π) The graph of $\left|H\left(e^{\jmath\omega}\right)\right|$ is symmetric with respect to y-axis. It is continuous over ω. The following code plots Fig. 4.6.

```
wget https://github.com/
    karthik6281/Signal-
    Processing/blob/main/
    Assignment1/codes/4_6.
    py
```

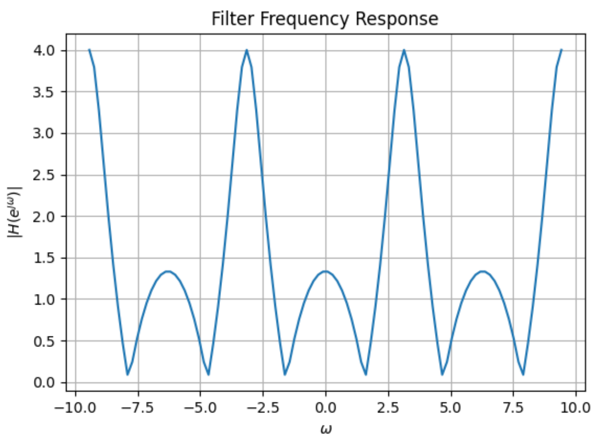

Fig. 4.6: $|H(e^{j\omega})|$

4.7 Express $h(n)$ in terms of $H(e^{j\omega})$.
Solution:

$$\int_{-\pi}^{\pi} H(e^{j\omega})e^{j\omega k}d\omega = \sum_{-\infty}^{\infty} h(n)\int_{-\pi}^{\pi} e^{j\omega n}e^{-j\omega k}d\omega \tag{4.37}$$

$$= \sum_{-\infty}^{\infty} h(n)\int_{-\pi}^{\pi}(\cos(n-k) + i\sin(n-k))d\omega \tag{4.38}$$

$$\int_{-\pi}^{\pi}(\cos(n-k) + i\sin(n-k))d\omega = \begin{cases} 2\pi & n = k \\ 0 & n \neq k \end{cases} \tag{4.39}$$

$$\therefore h(n) = \frac{\int_{-\pi}^{\pi} H(e^{j\omega})e^{j\omega n}d\omega}{2\pi} \tag{4.40}$$

5 Impulse Response

5.1 Using long division, find

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

for H(z) in (4.12).
Solution: $H(z)$ is given by

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

$$\begin{array}{r|l} & 2z^{-1} - 4 \\ \hline z^{-1} + 2 & 2z^{-2} + 2 \\ & 2z^{-2} + 4z^{-1} \\ \hline & -4z^{-1} + 2 \\ & -4z^{-1} - 8 \\ \hline & 10 \end{array}$$

So,

$$H(z) = 2z^{-1} - 4 + \frac{10}{z^{-1} + 2} \tag{5.3}$$

$$= 2z^{-1} - 4 + \frac{5}{\frac{1}{2}z^{-1} + 1} \tag{5.4}$$

$$= 2z^{-1} - 4 + 5\sum_{n=0}^{\infty}\left(-\frac{z^{-1}}{2}\right)^n \tag{5.5}$$

$$= 1 - \frac{1}{2}z^{-1} + 5\sum_{n=2}^{\infty}\left(-\frac{1}{2}\right)^n z^{-n} \tag{5.6}$$

So,h(n) will be given by

$$h(n) = \begin{cases} 5 \times \left(-\frac{1}{2}\right)^n & n \geq 2 \\ \left(-\frac{1}{2}\right)^n & 2 > n \geq 0 \\ 0 & n < 0 \end{cases} \tag{5.7}$$

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 (4.12),

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

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

using (4.22) and (4.6).

5.3 Sketch $h(n)$. Is it bounded? Justify theoretically.
Solution: The following code plots Fig. 5.3.

```
wget https://github.com/
    karthik6281/Signal-
    Processing/blob/main/
    Assignment1/codes/5_3.
    py
```

on simplfying we get h(n) as

$$\begin{cases} 5\times\left(-\frac{1}{2}\right)^n & n \geq 2 \\ \left(-\frac{1}{2}\right)^n & 2 > n \geq 0 \\ 0 & n < 0 \end{cases} \tag{5.11}$$

$$\because 5\times\left(-\frac{1}{2}\right)^n \to 0 \quad \text{for} \quad n \to \infty \tag{5.12}$$

So, we can conclude that h(n) is bounded.

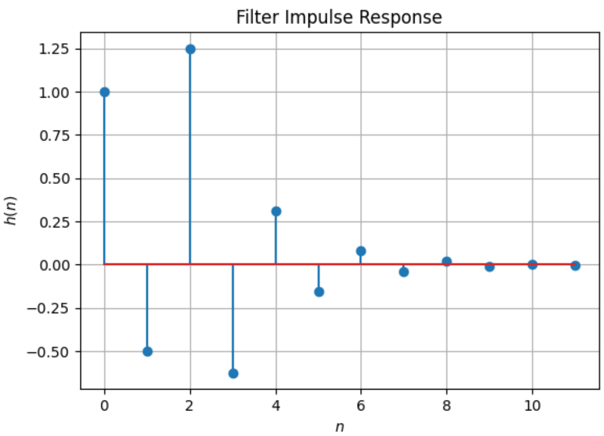

Fig. 5.3: $h(n)$ wrt n

5.4 Convergent? Justify using the ratio test.
Solution: A sequence $\{x_n\}$ is convergent if

$$\lim_{n\to\infty}\left|\frac{x_{n+1}}{x_n}\right| < 1 \tag{5.13}$$

This is known as Ratio test.
In this case the limit will become,

$$\lim_{n\to\infty}\left|\frac{h\left(n+1\right)}{h\left(n\right)}\right| = \lim_{n\to\infty}\left|\frac{5\left(\frac{-1}{2}\right)^{n+1}}{5\left(\frac{-1}{2}\right)^{n}}\right| \tag{5.14}$$

$$= \frac{1}{2} < 1 \tag{5.15}$$

$\therefore h\left(n\right)$ 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: Taking $h(n)$ as defined in (5.7) Then

$$\sum_{n=-\infty}^{\infty} h(n) = +\sum_{n=-\infty}^{-1} 0 + \sum_{n=0}^{1}\left(-\frac{1}{2}\right)^n + \sum_{n=2}^{\infty} 5\times\left(-\frac{1}{2}\right)^n \tag{5.17}$$

$$= \frac{4}{3} \tag{5.18}$$

Since the sum is finite so the system is stable for impulsive response

5.6 Verify the above result using a python code.
Solution: The above result is verified using the below python code

```
wget https://github.com/karthik6281/Signal−
    Processing/tree/main/Assignment1/codes/5
    _6.py
```

5.7 Compute and sketch $h(n)$ using

$$h(n) + \frac{1}{2}h(n-1) = \delta(n) + \delta(n-2), \tag{5.19}$$

This is the definition of $h(n)$.
Solution: The following code plots Fig. 5.7. Note that this is the same as Fig. 3.1.

```
wget https://github.com/karthik/Signal−
    Processing/tree/main/Assignment1/codes/5
    _7.py
```

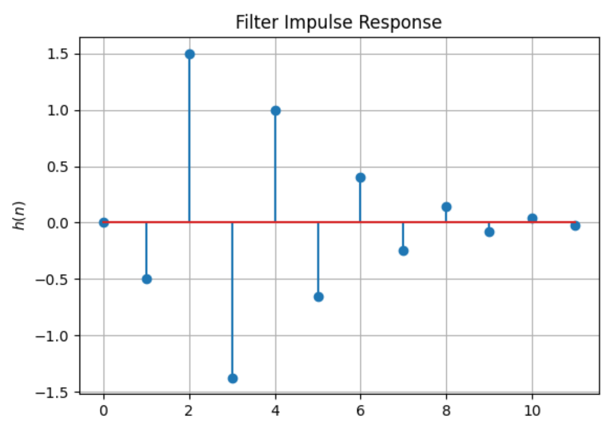

Fig. 5.7: $h(n)$ from the definition

5.8 Compute

$$y(n) = x(n) * h(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) \tag{5.20}$$

Comment. The operation in (5.20) is known as *convolution*.

Solution: The following code plots Fig. 5.8. Note that this is the same as $y(n)$ in Fig. 3.1.

```
wget https://github.com/karthik/Signal-Processing/tree/main/Assignment1/codes/5_8.py
```

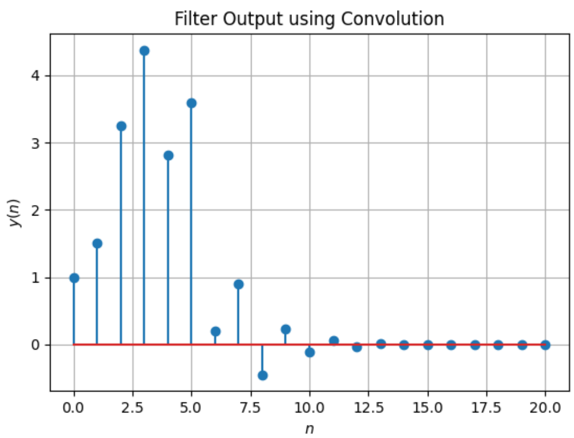

Fig. 5.8: $y(n)$ from the definition of convolution

5.9 Express the above convolution using a Toeplitz matrix.
Solution:

```
wget https://github.com/karthik6281/Signal-Processing/tree/main/Assignment1/codes/5_9.py
```

From (5.20),we express $y(n)$ as

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)\, h(n-k) \tag{5.21}$$

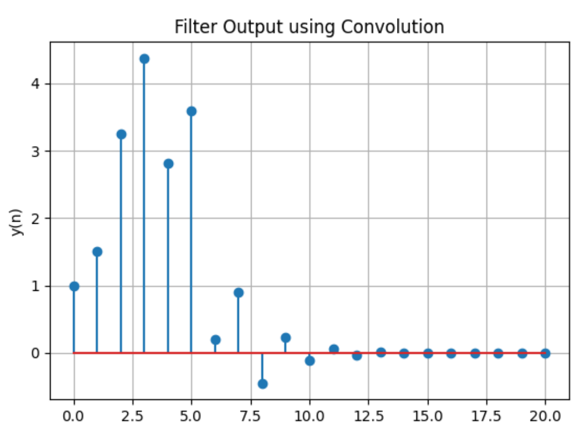

Fig. 5.9: Convolution of $x(n)$ and $h(n)$ using toeplitz matrix

To understand how we can use a Toeplitz matrix, we will see what we are doing in (5.20)

$$y(0) = x(0)\, h(0) \tag{5.22}$$
$$y(1) = x(0)\, h(1) + x(1)\, h(0) \tag{5.23}$$
$$y(2) = x(0)\, h(2) + x(1)\, h(1) + x(2)\, h(0) \tag{5.24}$$

.

.

The same thing can be written as,

$$y(0) = \begin{pmatrix} h(0) & 0 & 0 & . & . & .0 \end{pmatrix} \begin{pmatrix} x(0) \\ x(1) \\ x(2) \\ . \\ . \\ x(5) \end{pmatrix} \tag{5.25}$$

$$y(1) = \begin{pmatrix} h(1) & h(0) & 0 & 0 & . & . & .0 \end{pmatrix} \begin{pmatrix} x(0) \\ x(1) \\ x(2) \\ . \\ . \\ x(5) \end{pmatrix} \tag{5.26}$$

$$y(2) = \begin{pmatrix} h(2) & h(1) & h(0) & 0 & . & .0 \end{pmatrix} \begin{pmatrix} x(0) \\ x(1) \\ x(2) \\ . \\ . \\ x(5) \end{pmatrix} \tag{5.27}$$

.

.

Using Toeplitz matrix of $h(n)$ we can simplify it as,

$$y(n) = \begin{pmatrix} h(0) & 0 & 0 & . & . & .0 \\ h(1) & h(0) & 0 & . & . & .0 \\ h(2) & h(1) & h(0) & . & . & .0 \\ & & .. & & & \\ & & .. & & & \\ 0 & 0 & 0 & . & . & .h(m-1) \end{pmatrix} \begin{pmatrix} x(0) \\ x(1) \\ x(2) \\ . \\ . \\ x(5) \end{pmatrix} \tag{5.28}$$

Now from (3.1) we will take n

$$x(n) = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 2 \\ 1 \end{pmatrix} \tag{5.29}$$

And from (5.7) we will take some values of n,

$$h(n) = \begin{pmatrix} 1 \\ -0.5 \\ 1.25 \\ . \\ . \end{pmatrix} \tag{5.30}$$

Now using (5.28),

$$y(n) = x(n) * h(n) \tag{5.31}$$

$$= \begin{pmatrix} 1 & 0 & 0 & . & . & .0 \\ -0.5 & 1 & 0 & . & . & .0 \\ 1.25 & -0.5 & 1 & . & . & .0 \\ & & .. & & & \\ & & .. & & & \\ 0 & 0 & 0 & . & . & . \end{pmatrix} \begin{pmatrix} x(0) \\ x(1) \\ x(2) \\ . \\ . \\ x(5) \end{pmatrix} \tag{5.32}$$

$$= \begin{pmatrix} 1 \\ 1.5 \\ 3.25 \\ . \\ . \\ . \end{pmatrix} \tag{5.33}$$

5.10 Show that

$$y(n) = \sum_{k=-\infty}^{\infty} x(n-k)h(k) \tag{5.34}$$

Solution: From (5.20)

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

Replacing n-k with a,we get

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

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

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

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 python code can be obtained from

```
wget https://github.com/
    karthik6281/Signal-
    Processing/tree/main/
    Assignment1/codes/6_1.
    py
```

Execute the following commands

```
python3 6_1.py
```

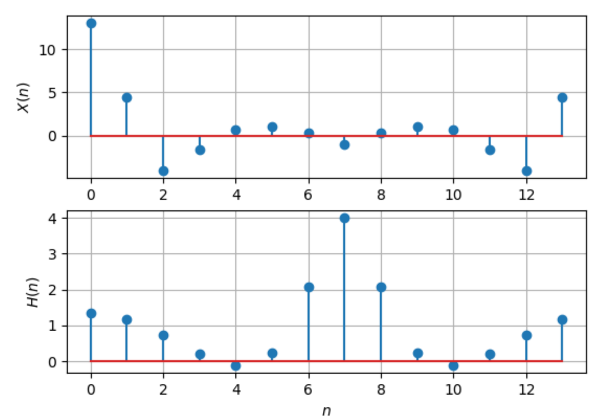

Fig. 6.1: Plots of the real parts of the discrete Fourier transforms of $x(n)$ and $h(n)$

6.2 Compute

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

Solution: The python code can be obtained from

```
wget https://github.com/
    karthik6281/Signal-
    Processing/tree/main/
    Assignment1/codes/6_2.
    py
```

Execute the following commands

```
python3 6_2.py
```

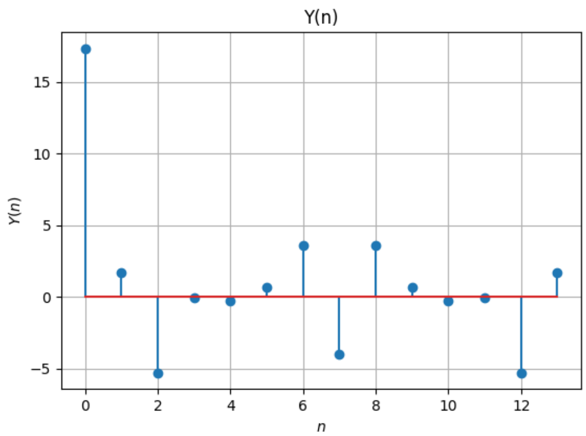

Fig. 6.2: Plot of $Y(k)$

6.3 Compute

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

Solution: The python code can be obtained from

```
wget https://github.com/karthik6281/Signal-Processing/tree/main/Assignment1/codes/6_3.py
```

Execute the following commands

```
python3 6_3.py
```

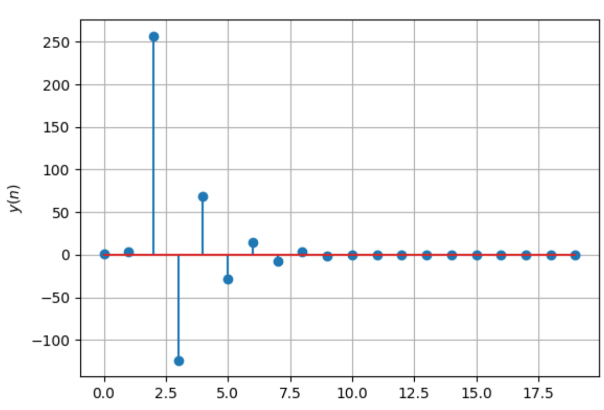

Fig. 6.3: Plot using difference equation of $Y(k)$

6.4 Repeat the previous exercise by computing $X(k), H(k)$ and $y(n)$ through FFT and IFFT.

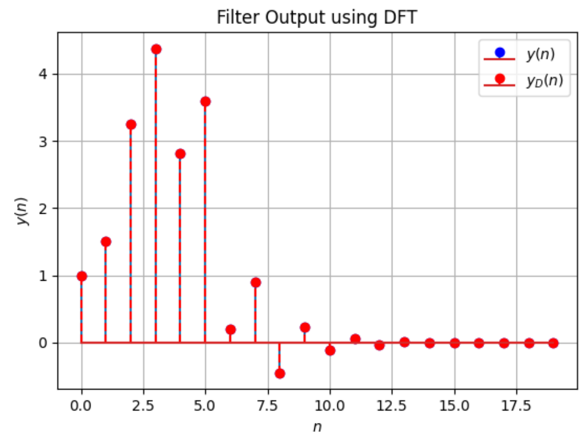

Fig. 6.3: Plot of the inverse discrete Fourier transform of $Y(k)$

Solution: The python code can be obtained from

```
wget https://github.com/karthik6281/Signal-Processing/tree/main/Assignment1/codes/6_4.py
```

Execute the following commands

```
python3 6_4.py
```

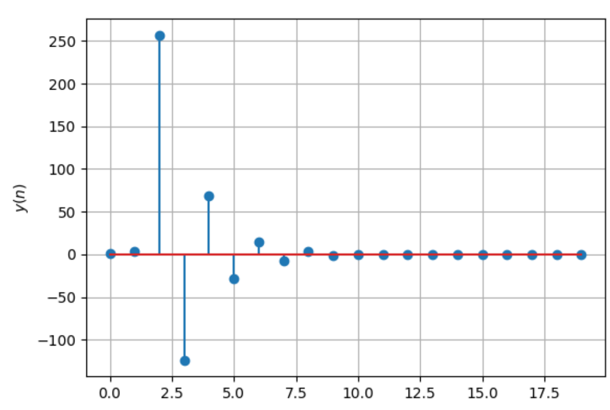

Fig. 6.4: Plot using difference equation of $Y(k)$

6.5 Wherever possible, express all the above equations as matrix equations.
Solution: We use the DFT Matrix, where

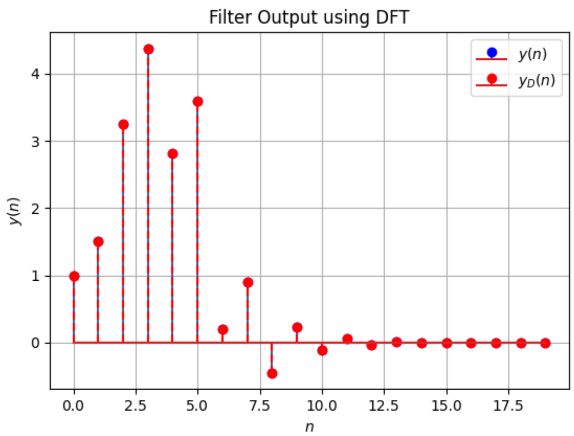

Fig. 6.4: Plot of the inverse discrete Fourier transform of $Y(k)$

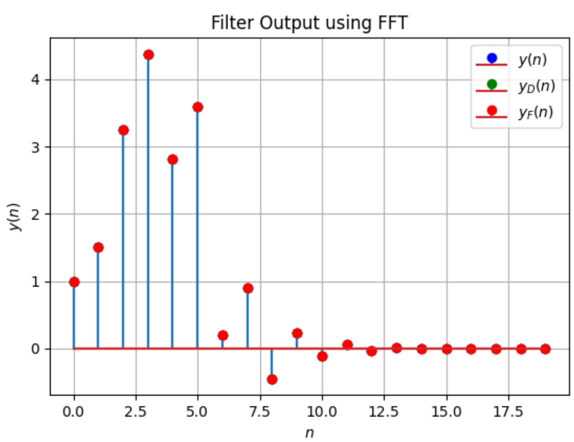

Fig. 6.4: Plot of $y(n)$ by fast Fourier transform

$\omega = e^{-\frac{j2k\pi}{N}}$, which is given by

$$\mathbf{W} = \begin{pmatrix} \omega^0 & \omega^0 & \dots & \omega^0 \\ \omega^0 & \omega^1 & \dots & \omega^{N-1} \\ \vdots & \vdots & \ddots & \vdots \\ \omega^0 & \omega^{N-1} & \dots & \omega^{(N-1)(N-1)} \end{pmatrix} \tag{6.4}$$

i.e. $W_{jk} = \omega^{jk}$, $0 \le j, k < N$. Hence, we can write any DFT equation as

$$\mathbf{X} = \mathbf{W}\mathbf{x} = \mathbf{x}\mathbf{W} \tag{6.5}$$

where

$$\mathbf{x} = \begin{pmatrix} x(0) \\ x(1) \\ \vdots \\ x(n-1) \end{pmatrix} \tag{6.6}$$

The inverse Fourier Transform is given by

$$\mathbf{x} = \mathcal{F}^{-1}(\mathbf{X}) = \mathbf{W}^{-1}\mathbf{X} = \frac{1}{N}\mathbf{W}^{\mathbf{H}}\mathbf{X} = \frac{1}{N}\mathbf{X}\mathbf{W}^{\mathbf{H}} \tag{6.7}$$

$$\implies \mathbf{W}^{-1} = \frac{1}{N}\mathbf{W}^{\mathbf{H}} \tag{6.8}$$

where H denotes hermitian operator. We can compute 6.2 using

$$\mathbf{Y} = \mathbf{H} \cdot \mathbf{X} = (\mathbf{W}\mathbf{h}) \cdot (\mathbf{W}\mathbf{x}) \tag{6.9}$$

6.6 Verify the above equations by generating the DFT matrix in Python.
Solution: The python code can be obtained from

```
wget https://github.com/karthik6281/
    Signal−Processing/tree/main/
    Assignment1/codes/6_6.py
```

Execute the following commands

```
python3 6_6.py
```