Digital Signal Processing

Gautam Singh

Contents

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

1 Software Installation

Run the following commands (commands may change depending on Linux distro)

```
$ sudo apt update && sudo apt upgrade
$ sudo apt install libffi-dev libsndfile1 python3-
    scipy python3-pip python3-numpy python3-
    matplotlib
$ pip3 install cffi pysoundfile
```

2 Digital Filter

2.1 Download the sound file using

```
$ wget https://raw.githubusercontent.com/
    goats-9/ee3900-assignments/main/
    Assignment_01/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: Download the source code using

```
$ wget https://raw.githubusercontent.com/
    goats-9/ee3900-assignments/main/
    Assignment_01/codes/2_3.py
```

and execute it using

```
$ python3 2_3.py
```

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

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

Sketch $x(n)$.

3.2 Let

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

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

```
$ wget https://raw.githubusercontent.com/
    goats-9/ee3900-assignments/main/codes
    /3_2.py
```

and execute it using

```
$ python3 3_2.py
```

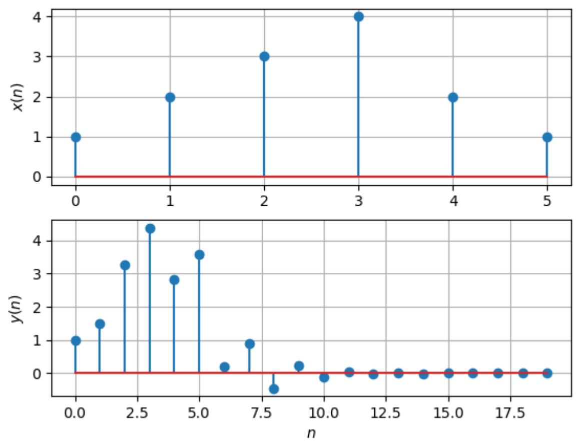

Fig. 3.2: Plot of $x(n)$ and $y(n)$

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),

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

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

$$= z^{-k} \sum_{n=-\infty}^{\infty} x(n) z^{-n} \tag{4.6}$$

$$= z^{-k} X(n) \tag{4.7}$$

Putting $k = 1$ gives (4.2).

4.2 Find

$$H(z) = \frac{Y(z)}{X(z)} \tag{4.8}$$

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

Solution: Applying (4.7) in (3.2),

$$Y(z) + \frac{1}{2} z^{-1} Y(z) = X(z) + z^{-2} X(z) \tag{4.9}$$

$$\implies \frac{Y(z)}{X(z)} = \frac{1 + z^{-2}}{1 + \frac{1}{2} z^{-1}} \tag{4.10}$$

4.3 Find the Z transform of

$$\delta(n) = \begin{cases} 1 & n = 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.11}$$

and show that the Z-transform of

$$u(n) = \begin{cases} 1 & n \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.12}$$

is

$$U(z) = \frac{1}{1 - z^{-1}}, \quad |z| > 1 \tag{4.13}$$

Solution: We see using (4.11) that

$$\mathcal{Z}\{\delta(n)\} = \delta(0) = 1 \tag{4.14}$$

and from (4.12),

$$U(z) = \sum_{n=0}^{\infty} z^{-n} \tag{4.15}$$

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

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

4.4 Show that

$$a^n u(n) \overset{\mathcal{Z}}{\rightleftharpoons} \frac{1}{1 - a z^{-1}} \quad |z| > |a| \tag{4.17}$$

Solution:

$$a^n u(n) \overset{\mathcal{Z}}{\rightleftharpoons} \sum_{n=0}^{\infty} \left(a z^{-1}\right)^n \tag{4.18}$$

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

4.5 Let

$$H\left(e^{\jmath\omega}\right) = H\left(z = e^{\jmath\omega}\right). \tag{4.20}$$

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

Solution: The following code plots Fig. (4.5).

```
$ wget https://raw.githubusercontent.com/
    goats-9/ee3900-assignments/main/
    Assignment_01/codes/4_5.py
```

The figure can be generated using

```
$ python3 4_5.py
```

We observe that $\left|H\left(e^{j\omega}\right)\right|$ is periodic with fundamental period 2π.

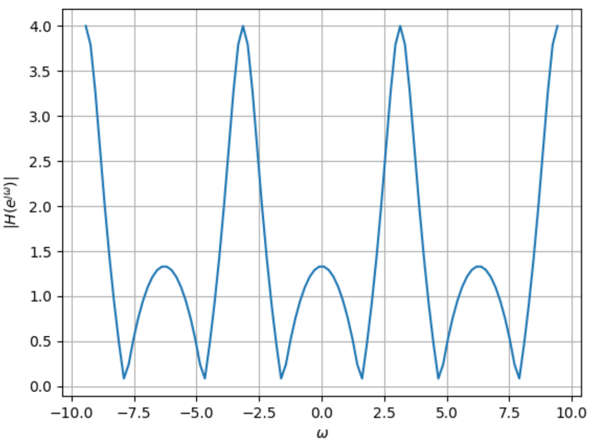

Fig. 4.5: Plot of $\left|H\left(e^{j\omega}\right)\right|$ against ω

5 Impulse Response

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

$$h(n) \overset{\mathcal{Z}}{\rightleftharpoons} H(z) \tag{5.1}$$

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 (4.10),

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

$$\implies h(n) = \left(-\frac{1}{2}\right)^n u(n) + \left(-\frac{1}{2}\right)^{n-2} u(n-2) \tag{5.3}$$

using (4.17) and (4.7).

5.2 Sketch $h(n)$. Is it bounded? Convergent?
Solution: The following code plots Fig. (5.2).

```
$ wget https://raw.githubusercontent.com/
    goats-9/ee3900-assignments/main/
    Assignment_01/codes/5_2.py
```

and execute it using

```
$ python3 5_2.py
```

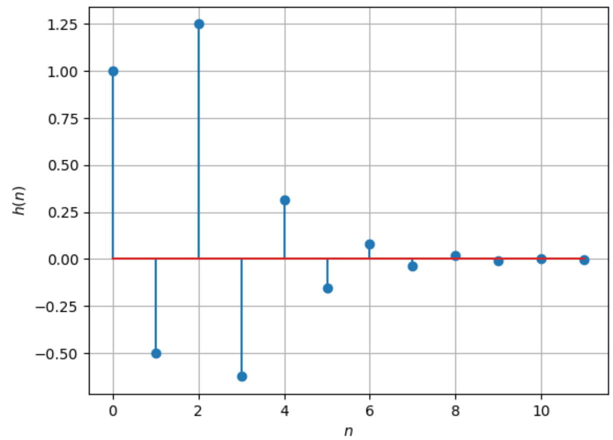

Fig. 5.2: $h(n)$ as the inverse of $H(z)$

$h(n)$ is bounded and convergent.

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

$$\sum_{n=-\infty}^{\infty} h(n) < \infty \tag{5.4}$$

Is the system defined by (3.2) stable for the impulse response in (5.1)?
Solution: Note that

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

$$= 2\left(\frac{1}{1+\frac{1}{2}}\right) = \frac{4}{3} \tag{5.6}$$

Thus, the given system is stable.

5.4 Compute and sketch $h(n)$ using

$$h(n) + \frac{1}{2}h(n-1) = \delta(n) + \delta(n-2), \tag{5.7}$$

This is the definition of $h(n)$.
Solution: The following code plots Fig. (5.4). Note that this is the same as Fig. (5.2).

```
$ wget https://raw.githubusercontent.com/
    goats-9/ee3900-assignments/main/
    Assignment_01/codes/5_4.py
```

and executed using

```
$ python3 5_4.py
```

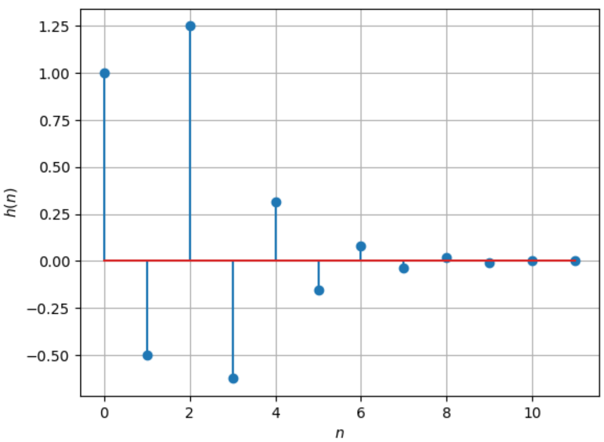

Fig. 5.4: $h(n)$ as the inverse of $H(z)$

5.5 Compute

$$y(n) = x(n) * h(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) \quad (5.8)$$

Comment. The operation in (5.8) is known as *convolution*.
Solution: The following code plots Fig. (5.5). Note that this is the same as $y(n)$ in Fig. (3.2).

```
$ wget https://raw.githubusercontent.com/
    goats-9/ee3900-assignments/main/
    Assignment_01/codes/5_5.py
```

and executed using

```
$ python3 5_5.py
```

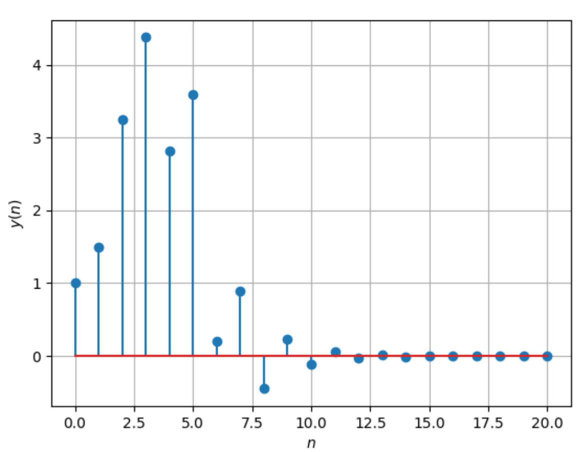

Fig. 5.5: $y(n)$ from the definition

5.6 Show that

$$y(n) = \sum_{k=-\infty}^{\infty} x(n-k)h(k) \quad (5.9)$$

Solution: From (5.8), we substitute $k := n-k$ to get

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)\, h(n-k) \quad (5.10)$$
$$= \sum_{n-k=-\infty}^{\infty} x(n-k)\, h(k) \quad (5.11)$$
$$= \sum_{k=-\infty}^{\infty} x(n-k)\, h(k) \quad (5.12)$$

6 DFT and FFT

6.1 Compute

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

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

6.2 Compute

$$Y(k) = X(k)H(k) \quad (6.2)$$

6.3 Compute

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

Solution: The following code plots Fig. (6.3) and computes $X(k)$ and $Y(k)$. Note that this is the same as $y(n)$ in Fig. (3.2).

```
$ wget https://raw.githubusercontent.com/
    goats-9/ee3900-assignments/main/
    Assignment_01/codes/6_3.py
```

and executed using

```
$ python3 6_3.py
```

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

```
$ wget https://raw.githubusercontent.com/
    goats-9/ee3900-assignments/main/
    Assignment_01/codes/6_4.py
```

and execute it using

```
$ python3 6_4.py
```

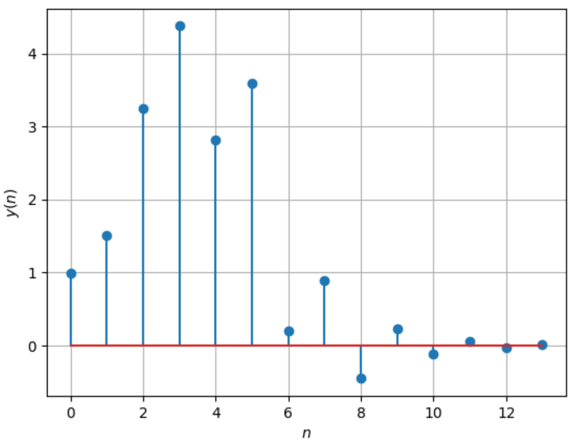

Fig. 6.3: $y(n)$ from the DFT

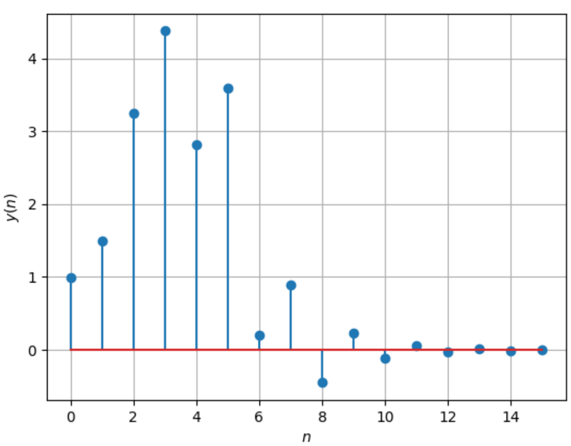

Fig. 6.4: $y(n)$ using FFT and IFFT

Observe that Fig. (6.4) is the same as $y(n)$ in Fig. (3.2).

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

Solution: We use the DFT Matrix, where $\omega = e^{-\frac{j2k\pi}{N}}$, which is given by

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

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

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

Using (6.3), 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 rewrite (6.2) using the element-wise multiplication operator as

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

7 Exercises

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

7.1 The command

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

in Problem 2.3 is executed through the following difference equation

$$\sum_{m=0}^{M} a(m)\, y(n-m) = \sum_{k=0}^{N} b(k)\, x(n-k) \tag{7.1}$$

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: The implementation is at

```
$ wget https://raw.githubusercontent.com/
    goats-9/ee3900-assignments/main/
    Assignment_01/codes/7_1.py
```

and can be run using

```
$ python3 7_1.py
```

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

Solution: The filter frequency response is plotted at

```
$ wget https://raw.githubusercontent.com/
    goats-9/ee3900-assignments/main/
    Assignment_01/codes/7_2_1.py
```

To plot the impulse response function $h(n)$, we rewrite $H(z)$ using partial fractions to take the inverse z-transform.

$$H(z) = \sum_i \frac{r_i}{1 - p_i z^{-1}} + \sum_j k_j z^{-j} \tag{7.2}$$

The impulse response function is plotted at

```
$ wget https://raw.githubusercontent.com/
    goats−9/ee3900−assignments/main/
    Assignment_01/codes/7_2_2.py
```

We see that $h(n)$ is bounded and convergent. Also, since 1 is not a pole of the transfer function, the system is stable. The codes can be run all at once by typing

```
$ python3 7_2_*.py
```

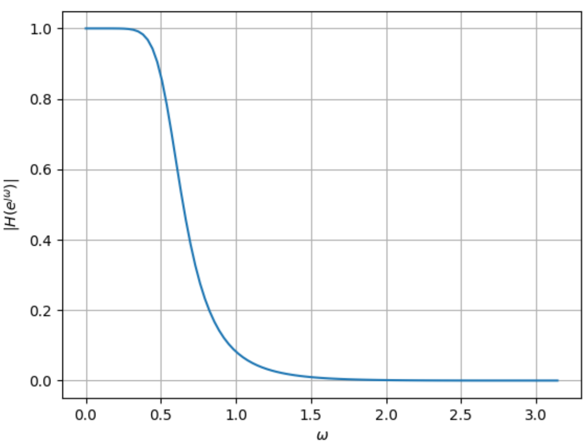

Fig. 7.2: Filter frequency response

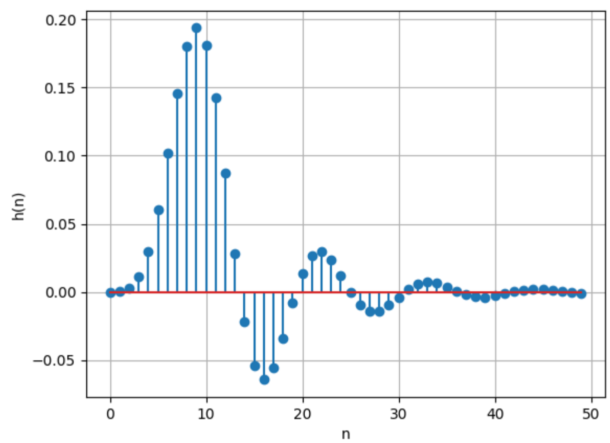

Fig. 7.2: Plot of $h(n)$

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

7.4 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.

7.5 Modifying the code with different input parameters and to get the best possible output.
Solution: A better filtering was found on setting the order of the filter to be 7.