Digital Signal Processing

Aditya Gangula EP20BTECH11001*

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 **??** 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 **??** is the audio file Sound_With_ReducedNoise.wav. Play the file in the spectrogram in Problem **??**. 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 $x(n)$.
Solution:

```
import numpy as np
import matplotlib.pyplot as plt
```

*

```
x=np.array([1.0,2.0,3.0,4.0,2.0,1.0])

plt.stem(range(0,6),x)
plt.xlabel('$n$')
plt.ylabel('$x(n)$')
plt.grid()
plt.show()
```

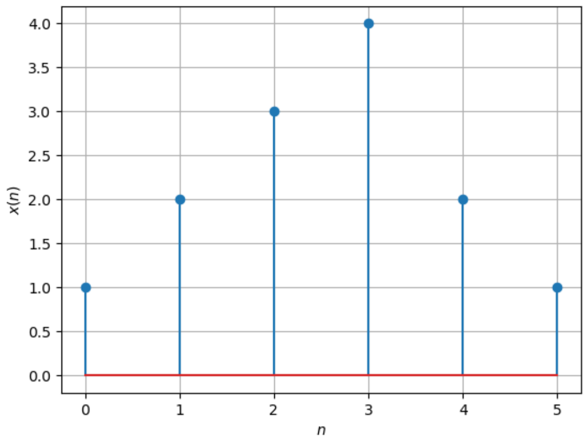

Fig. 3.1

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. ??.

```
import numpy as np
import matplotlib.pyplot as plt



x=np.array([1.0,2.0,3.0,4.0,2.0,1.0])
k = 20
y = np.zeros(20)


y[0] = x[0]
y[1] = −0.5∗y[0]+x[1]

for n in range(2,k−1):
        if n < 6:
                y[n] = −0.5∗y[n−1]+x[n]+x
                    [n−2]
        elif n > 5 and n < 8:
                y[n] = −0.5∗y[n−1]+x[n−2]
        else:
                y[n] = −0.5∗y[n−1]
print(y)

plt.stem(range(0,k),y)
plt.xlabel('$n$')
plt.ylabel('$y(n)$')
plt.grid()
plt.show()
```

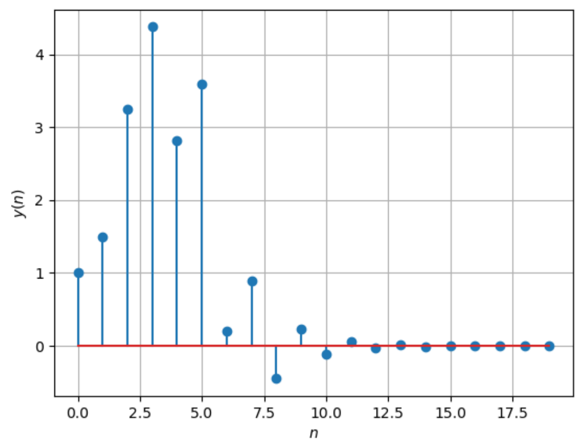

Fig. 3.2

3.3 Repeat the above exercise using a C code.
Solution:

```
#include<stdio.h>

int main()
{
    float x[6] = {1.0,2.0,3.0,4.0,2.0,1.0};
    int k =20;
    float y[20] = {0};

    y[0] = x[0];
    y[1] = −0.5∗y[0]+ x[1];

    for (int i=2; i<k; i++)
    {
        if (i<6)
        {
            y[i] = −0.5∗y[i−1]+x[i]+x[i−2];
        }
```

```
        else if (i>5 && i<8)
        {
            y[i] = −0.5∗y[i−1]+x[i−2];
        }
        else
        {
            y[i] = −0.5∗y[i−1];
        }
    }
    for (int i =0; i<20; i++)
    {
        printf("%f,␣",y[i]);
    }

    FILE ∗x_val;
    FILE ∗y_val;
    x_val = fopen("3.3x.txt","w");
    y_val = fopen("3.3y.txt","w");
    for(int i=0; i<6; i++)
    {
        fprintf(x_val,"%f␣", x[i]);
    }
    for(int i=0; i<20; i++)
    {
        fprintf(y_val,"%f␣", y[i]);
    }
    fclose(x_val);
    fclose(y_val);

}
```

```
import numpy as np
import matplotlib.pyplot as plt

x=np.loadtxt('3.3x.txt')
y=np.loadtxt('3.3y.txt')
k=20
#subplots
plt.subplot(2, 1, 1)
plt.stem(range(0,6),x)
plt.title('Digital␣Filter␣Input−Output')
plt.ylabel('$x(n)$')
plt.grid()


plt.subplot(2, 1, 2)
plt.stem(range(0,k),y)
plt.xlabel('$n$')
plt.ylabel('$y(n)$')
plt.grid()
plt.show()
```

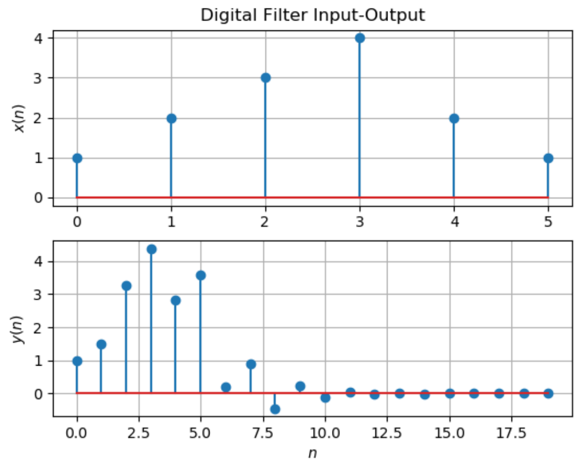

Fig. 3.3

4 Z-transform

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

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

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 (**??**),

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

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

resulting in (**??**). 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:

$$x(n) = \{1, 2, 3, 4, 2, 1\} \quad (4.7)$$

$$X(z) = 1 + 2/z + 3/z^2 + 4/z^3 + 2/z^4 + 1/z^5 \quad (4.8)$$

4.3 Find

$$H(z) = \frac{Y(z)}{X(z)} \tag{4.9}$$

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

Solution: Applying (**??**) in (**??**),

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

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

4.4 Find the Z transform of

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

and show that the Z-transform of

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

is

$$U(z) = \frac{1}{1 - z^{-1}}, \quad |z| > 1 \tag{4.14}$$

Solution: It is easy to show that

$$\delta(n) \overset{\mathcal{Z}}{\rightleftharpoons} 1 \tag{4.15}$$

and from (**??**),

$$U(z) = \sum_{n=0}^{\infty} z^{-n} \tag{4.16}$$

$$= \frac{1}{1 - z^{-1}}, \quad |z| > 1 \tag{4.17}$$

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

4.5 Show that

$$a^n u(n) \overset{\mathcal{Z}}{\rightleftharpoons} \frac{1}{1 - az^{-1}} \quad |z| > |a| \tag{4.18}$$

Solution:

$$\mathcal{Z}\{a^n u(n)\} = \sum_{n=-\infty}^{\infty} x(n-1)z^{-n} \tag{4.19}$$

$$= \sum_{n=0}^{\infty} a^n z^{-n} = \sum_{n=0}^{\infty} (az^{-1})^n \tag{4.20}$$

Since

$$|z| > |a| \tag{4.21}$$

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

$$\mathcal{Z}\{a^n u(n)\} = \frac{1}{1 - az^{-1}} \tag{4.22}$$

4.6 Let

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

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

Solution: The following code plots Fig. **??**.

```
import numpy as np
import matplotlib.pyplot as plt
#if using termux
import subprocess
import shlex
#end if


#DTFT
def H(z):
        num = np.polyval([1,0,1],z**(-1))
        den = np.polyval([0.5,1],z**(-1))
        H = num/den
        return H



#Input and Output
omega = np.linspace(-3*np.pi,3*np.pi,100)

#subplots
plt.plot(omega, abs(H(np.exp(1j*omega))))
plt.title('Filter Frequency Response')
plt.xlabel('$\omega$')
plt.ylabel('$|H(e^{\jmath\omega})| $')
plt.grid()# minor

plt.show()
```

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

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

Solution:

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

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

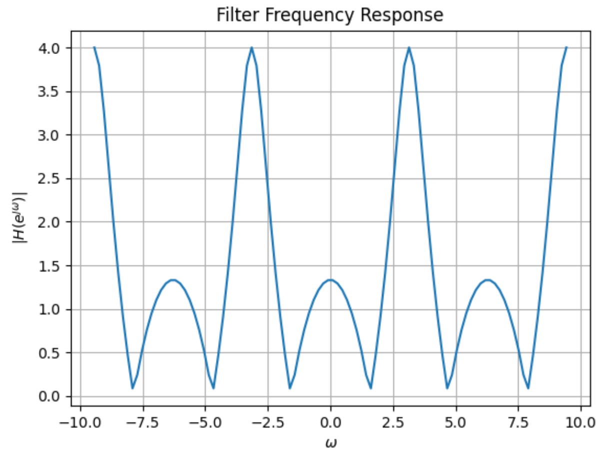

Fig. 4.6: $|H\left(e^{\jmath\omega}\right)|$

$$= \sum_{k=-\infty}^{\infty} h(k)\delta(n-k)$$

$$= h(n)$$

5 Impulse Response

5.1 Using long division, find

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

for H(z) in (**??**).
Solution:

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

$$\begin{array}{r|l} & \quad\ 2X - 4 \\ \hline \frac{1}{2}X + 1) & X^2 \qquad +1 \\ & -X^2 - 2X \\ \hline & \quad -2X + 1 \\ & \quad\ 2X + 4 \\ \hline & \qquad\quad 5 \end{array}$$

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

Using the inverse Z-transform

$$h(n) = 2\delta(n-1) - 4\delta(n) + 5\left(-\frac{1}{2}\right)^n u(n)$$

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

$$h(n) \overset{\mathcal{Z}}{\rightleftharpoons} H(z) \tag{5.2}$$

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 (**??**).
Solution: From (**??**),

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

$$\implies h(n) = \left(-\frac{1}{2}\right)^n u(n) + \left(-\frac{1}{2}\right)^{n-2} u(n-2) \tag{5.4}$$

using (**??**) and (**??**).

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

```
import numpy as np
import matplotlib.pyplot as plt



n = np.arange(10)
fn=(-1/2)**n
hn1=np.pad(fn, (0,2), 'constant',
    constant_values=(0))
hn2=np.pad(fn, (2,0), 'constant',
    constant_values=(0))
plt.stem(np.arange(12), hn1+hn2)
plt.title('Filter Impulse Response')
plt.xlabel('$n$')
plt.ylabel('$h(n)$')
plt.grid()# minor

plt.show()
```

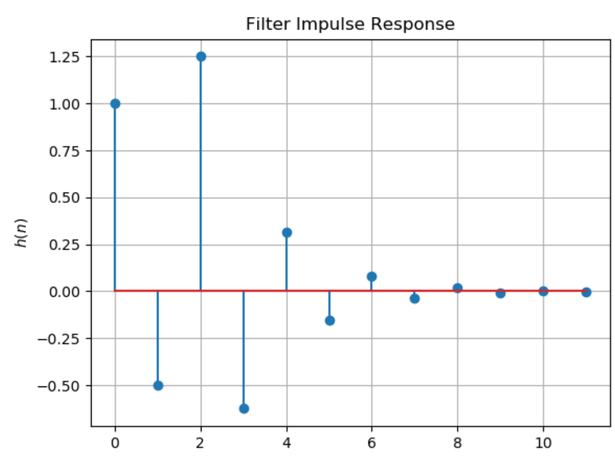

Fig. 5.3: $h(n)$ as the inverse of $H(z)$

From the figure it clear that h(n) is bounded,

also
$$\left|\left(-\frac{1}{2}\right)^{n-2} u(n-2)\right| \le 1$$
$$h(n) \le 2$$

5.4 Convergent? Justify using the ratio test.
Solution:
$$\lim_{n\to\infty}\left|\frac{h(n+1)}{h(n)}\right|$$
$$= \lim_{n\to\infty}\left|\frac{\left(-\frac{1}{2}\right)^{n+1} u(n+1) + \left(-\frac{1}{2}\right)^{n-1} u(n-1)}{\left(-\frac{1}{2}\right)^{n} u(n) + \left(-\frac{1}{2}\right)^{n-2} u(n-2)}\right|$$
$$= \left|\frac{-\frac{1}{2}-2}{1+4}\right|$$
$$= \frac{1}{2} < 1$$
h(n) is convergent.

5.5 The system with $h(n)$ is defined to be stable if
$$\sum_{n=-\infty}^{\infty} h(n) < \infty \tag{5.5}$$
Is the system defined by (**??**) stable for the impulse response in (**??**)?
Solution:
$$\sum_{n=-\infty}^{\infty} h(n) = \sum_{n=-\infty}^{\infty}\left(-\frac{1}{2}\right)^{n} u(n) + \left(-\frac{1}{2}\right)^{n-2} u(n-2)$$
$$= \sum_{n=0}^{\infty}\left(-\frac{1}{2}\right)^{n} + \sum_{n=2}^{\infty}\left(-\frac{1}{2}\right)^{n-2}$$
$$= \frac{1}{1+\frac{1}{2}} + \frac{1}{1+\frac{1}{2}}$$
$$= \frac{4}{3} < \infty$$

5.6 Verify the above result using a python code.
Solution:

```
import numpy as np

def u(n):
    if n >= 0:
        return 1
    else:
        return 0

def h(n):
    return u(n) * (−0.5)**n + u(n−2) *
        (−0.5)**(n−2)

sum = 0
for i in range(100001):
    sum = sum + h(i)
print(sum)
```

5.7 Compute and sketch $h(n)$ using
$$h(n) + \frac{1}{2}h(n-1) = \delta(n) + \delta(n-2), \tag{5.6}$$
This is the definition of $h(n)$.
Solution: The following code plots Fig. **??**. Note that this is the same as Fig. **??**.

```
import numpy as np
import matplotlib.pyplot as plt



k = 12
h = np.zeros(k)
h[0] = 1
h[1] = −0.5*h[0]
h[2] = −0.5*h[1] + 1

for n in range(3,k−1):
                h[n] = −0.5*h[n−1]

#subplots
plt.stem(range(0,k),h)
plt.title('Impulse␣Response␣Definition')
plt.xlabel('$n$')
plt.ylabel('$h(n)$')
plt.grid()# minor
plt.show()
```

5.8 Compute
$$y(n) = x(n) * h(n) = \sum_{n=-\infty}^{\infty} x(k)h(n-k) \tag{5.7}$$
Comment. The operation in (**??**) is known as *convolution*.
Solution: The following code plots Fig. **??**. Note that this is the same as $y(n)$ in Fig. **??**.

```
import numpy as np
import matplotlib.pyplot as plt
#If using termux
import subprocess
import shlex
#end if
```

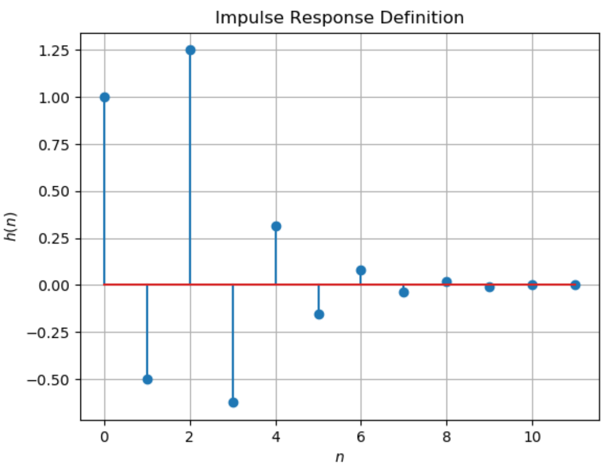

Fig. 5.7: $h(n)$ from the definition

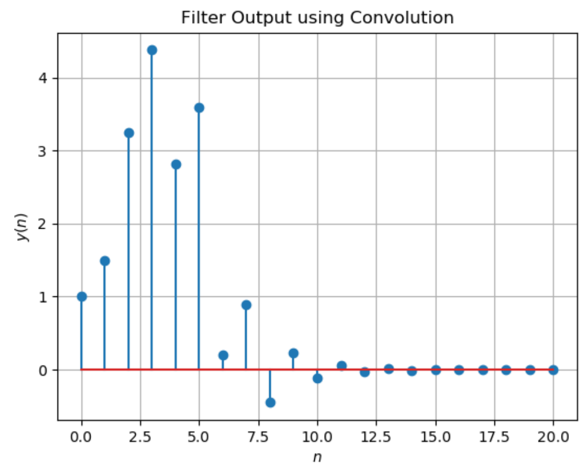

Fig. 5.8: $y(n)$ from the definition of convolution

```
n = np.arange(14)
fn=(-1/2)**n
hn1=np.pad(fn, (0,2), 'constant',
     constant_values=(0))
hn2=np.pad(fn, (2,0), 'constant',
     constant_values=(0))
h = hn1+hn2

nh=len(h)
x=np.array([1.0,2.0,3.0,4.0,2.0,1.0])
nx = len(x)

y = np.zeros(nx+nh-1)

for k in range(0,nx+nh-1):
          for n in range(0,nx):
                    if k-n >= 0 and k-n < nh:
                              y[k]+=x[n]*h[k-n]

print(y)
#plots
plt.stem(range(0,nx+nh-1),y)
plt.title('Filter Output using Convolution')
plt.xlabel('$n$')
plt.ylabel('$y(n)$')
plt.grid()# minor
plt.show()
```

5.9 Express the above convolution using a Teoplitz matrix.

Solution: Let x(n) and h(n) be represented by the following matrices.

$$x = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 2 \\ 1 \end{pmatrix} \qquad h = \begin{pmatrix} 1 \\ -0.5 \\ 1.25 \\ -0.625 \\ 0.315 \\ 0.15625 \end{pmatrix} \tag{5.8}$$

$$T = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ -0.5 & 1 & 0 & 0 & 0 \\ 1.25 & -0.5 & 1 & 0 & 0 \\ -0.625 & 1.25 & -0.5 & 1 & 0 \\ 0.315 & -0.625 & 1.25 & -0.5 & 1 \\ 0.156 & 0.315 & -0.625 & 1.25 & -0.5 \\ 0 & 0.156 & 0.315 & -0.625 & 1.25 \\ 0 & 0 & 0.156 & 0.315 & -0.625 \\ 0 & 0 & 0 & 0.156 & 0.315 \\ 0 & 0 & 0 & 0 & 0.156 \end{pmatrix}$$

We have

$$y = x * h = Tx$$

$$y = \begin{pmatrix} 1 \\ 1.5 \\ 3.25 \\ 4.38 \\ 2.81 \\ 3.59 \\ 0.12 \\ 0.78 \\ -0.62 \\ 0 \\ -0.16 \end{pmatrix}$$

5.10 Show that

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

Solution: We know that,

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

replacing n-k with k

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

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

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 $h(n)$.

6.2 Compute

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

6.3 Compute

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

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

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

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

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

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

6.6 Verify the above equations by generating the DFT matrix in python.

7 Exercises

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

7.1 The command

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

in Problem **??** 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.

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

7.3 What is the sampling frequency of the input signal?
Solution: Sampling frequency(fs)=44.1kHZ.

7.4 What is type, order and cutoff-frequency of the above butterworth filter
Solution: The given butterworth filter is low pass with order=2 and cutoff-frequency=4kHz.

7.5 Modifying the code with different input parameters and to get the best possible output.