EE3025 Assignment-1

Kuntal Kokate - EE18BTECH11028

Download all python codes from

https://github.com/Kkuntal990/EE3025−DSP/tree/main/assignment_1/code

and latex-tikz codes from

https://github.com/Kkuntal990/EE3025−DSP/blob/main/assignment_1/ee18btech11028.tex

1 Problem

1.1. Defining x(n) and h(n),

$$x(n) = \left\{\underset{\uparrow}{1}, 2, 3, 4, 2, 1\right\} \tag{1.1.1}$$

$$h(n) = \left(-\frac{1}{2}\right)^n u(n) + \left(-\frac{1}{2}\right)^{n-2} u(n-2) \tag{1.1.2}$$

1.2. Compute X(k), H(k) and y(n) using FFT and IFFT

2 Solution

2.1. input signal x(n)

$$x(n) = \left\{\underset{\uparrow}{1}, 2, 3, 4, 2, 1\right\} \tag{2.1.1}$$

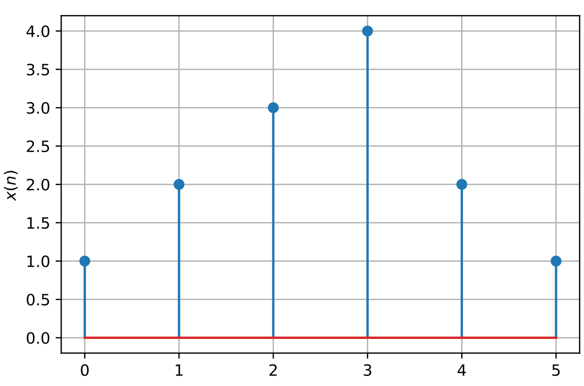

Fig. 2.1: input signal : x(n)

2.2. Impulse Response of the System is

$$h(n) = \left(-\frac{1}{2}\right)^n u(n) + \left(-\frac{1}{2}\right)^{n-2} u(n-2) \tag{2.2.1}$$

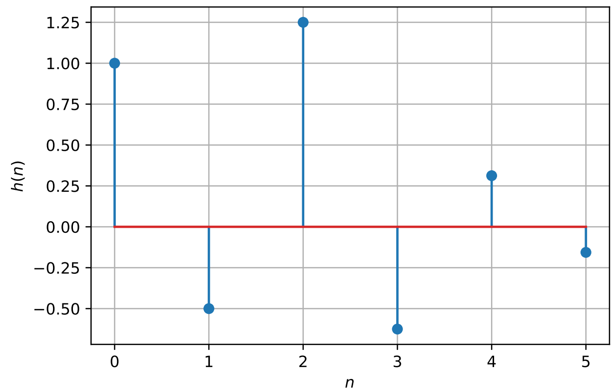

Fig. 2.2: impulse response : h(n)

2.3. FFT of the input signal x(n) is

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}, \quad k = 0, 1 \ldots N-1 \tag{2.3.1}$$

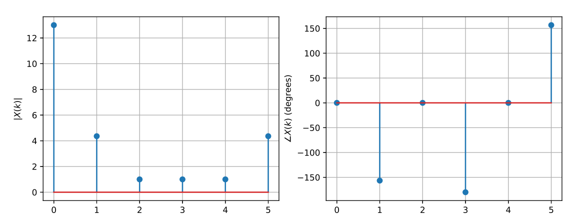

Fig. 2.3: FFT of x(n) : X(k)

2.4. FFT of the impulse response h(n) is

$$H(k) = \sum_{n=0}^{N-1} h(n) e^{-j2\pi kn/N}, \quad k = 0, 1, \ldots, N-1 \tag{2.4.1}$$

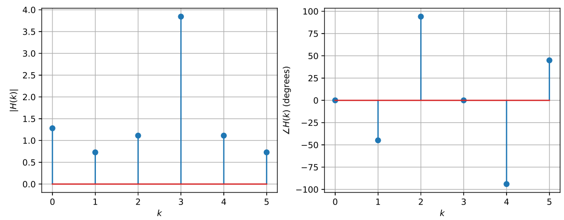

Fig. 2.4: FFT of h(n) : H(k)

2.5. FFT of output Signal y(n) can be computed as

$$Y(k) = X(k)H(k) \tag{2.5.1}$$

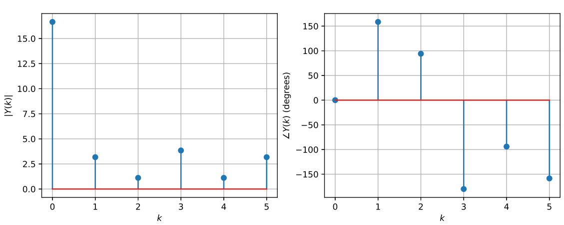

Fig. 2.5: $Y(k) = H(k)X(k)$

2.6. y(n) can be computed by taking IFFT of Y(k)

$$y(n) = \frac{1}{N}\sum_{n=0}^{N-1} Y(k)e^{j2\pi nk/N}, \quad k = 0, 1, \ldots, N-1 \tag{2.6.1}$$

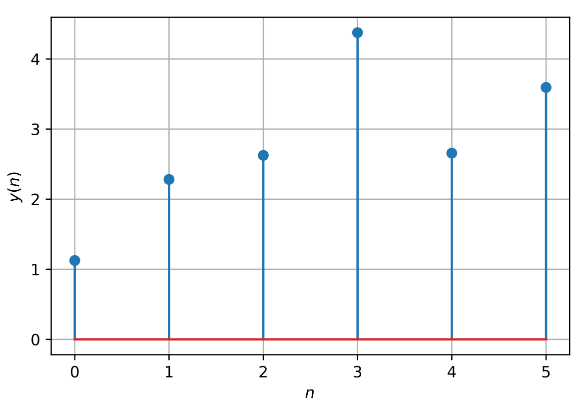

Fig. 2.6: IFFT of Y(k) : y(n)

3 Problem

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

4 Solution

4.1. FFT of signal X(n)

$$X(k) \triangleq W_N^{nk}x(n), \quad k = 0, 1, \ldots, N-1 \tag{4.1.1}$$

where $W_N^{nk} = e^{-j2\pi kn/N}$ which can be ex− pressed as:

$$\begin{bmatrix} X(0)\\ X(1)\\ X(2)\\ X(3)\\ X(4)\\ X(5) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1\\ 1 & W_N^1 & W_N^2 & W_N^3 & W_N^4 & W_N^5\\ 1 & W_N^2 & W_N^4 & W_N^6 & W_N^8 & W_N^{10}\\ 1 & W_N^3 & W_N^6 & W_N^9 & W_N^{12} & W_N^{15}\\ 1 & W_N^4 & W_N^8 & W_N^{12} & W_N^{16} & W_N^{20}\\ 1 & W_N^5 & W_N^{10} & W_N^{15} & W_N^{20} & W_N^{25} \end{bmatrix} \begin{bmatrix} x(0)\\ x(1)\\ x(2)\\ x(3)\\ x(4)\\ x(5) \end{bmatrix} \tag{4.1.2}$$

$$\implies \begin{bmatrix} X(0)\\ X(1)\\ X(2)\\ X(3)\\ X(4)\\ X(5) \end{bmatrix} = \begin{bmatrix} 1+2+3+4+2+1\\ 1+(2)e^{-j\pi/3}+\ldots+(1)e^{-j5\pi/3}\\ 1+(2)e^{-2j\pi/3}+\ldots+(1)(e^{-2j5\pi/3}\\ 1+(2)e^{-3j\pi/3}+\ldots+(1)e^{-3j5\pi/3}\\ 1+(2)e^{-4j\pi/3}+\ldots+(1)e^{-4j5\pi/3}\\ 1+(2)e^{-5j\pi/3}+\ldots+(1)e^{-5j5\pi/3} \end{bmatrix} \tag{4.1.3}$$

On solving,

$$\implies \begin{bmatrix} X(0)\\ X(1)\\ X(2)\\ X(3)\\ X(4)\\ X(5) \end{bmatrix} = \begin{bmatrix} 13+0j\\ -4-1.732j\\ 1+0j\\ -1+0j\\ 1+0j\\ -4+1.732j \end{bmatrix} \tag{4.1.4}$$

4.2. Similarly,

$$\begin{bmatrix} H(0)\\ H(1)\\ H(2)\\ H(3)\\ H(4)\\ H(5) \end{bmatrix} = \begin{bmatrix} h(0)+h(1)+h(2)+h(3)+h(4)+h(5)\\ h(0)+h(1)e^{-j\pi/3}+\ldots+h(5)e^{-j5\pi/3}\\ h(0)+h(1)e^{-2j\pi/3}+\ldots+h(5)e^{-2j5\pi/3}\\ h(0)+h(1)e^{-3j\pi/3}+\ldots+h(5)e^{-3j5\pi/3}\\ h(0)+h(1)e^{-4j\pi/3}+\ldots+h(5)e^{-4j5\pi/3}\\ h(0)+h(1)e^{-5j\pi/3}+\ldots+h(5)e^{-5j5\pi/3} \end{bmatrix} \tag{4.2.1}$$

On solving,

$$\implies \begin{bmatrix} H(0) \\ H(1) \\ H(2) \\ H(3) \\ H(4) \\ H(5) \end{bmatrix} = \begin{bmatrix} 1.28125 + 0j, \\ 0.51625 - 0.5141875j, \\ -0.078125 + 1.1095625j, \\ 3.84375 + 0j, \\ -0.071825 - 1.1095625j. \\ 0.515625 + 0.5141875j \end{bmatrix} \tag{4.2.2}$$

4.3. Compute $Y(k)$ using Eq (2.5.1)

$$\begin{bmatrix} Y(0) \\ Y(1) \\ Y(2) \\ Y(3) \\ Y(4) \\ Y(5) \end{bmatrix} = \begin{bmatrix} X(0) \cdot H(0) \\ X(1) \cdot H(1) \\ X(2) \cdot H(2) \\ X(3) \cdot H(3) \\ X(4) \cdot H(4) \\ X(5) \cdot H(5) \end{bmatrix} \tag{4.3.1}$$

Solving,

$$\begin{bmatrix} Y(0) \\ Y(1) \\ Y(2) \\ Y(3) \\ Y(4) \\ Y(5) \end{bmatrix} = \begin{bmatrix} 16.6562 + 0j \\ -2.95312 + 1.16372j \\ -0.07812 + 1.10959j \\ -3.84375 - 9.27556j \\ -0.07812 - 1.10959j \\ -2.95312 - 1.16372j \end{bmatrix} \tag{4.3.2}$$

4.4.

$$y(n) \triangleq (W_N^{nk})^* Y(k), \quad n = 0, 1, \dots, N-1 \tag{4.4.1}$$

where $(W_N^{nk})^*$ is conjugate of W_N^{nk} from (4.1.1).

$$\begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ y(4) \\ y(5) \end{bmatrix} = \begin{bmatrix} 1.125 + 0j \\ 2.28125071 + 0j \\ 2.6250019 - 1.11022302 \times 10^{-16} j \\ 4.37499667 - 1.47104551 \times 10^{-15} j \\ 2.6562481 + 6.10622664 \times 10^{-16} j \\ 3.59375262 - 1.60982339 \times 10^{-15} j \end{bmatrix} \tag{4.4.2}$$

4.5. Properties :

a) symmetry property :

$$W_N^{k+N/2} = -W_N^k$$

b) Periodicity property :

$$W_N^{k+N} = W_N^k$$

c)

$$W_N^2 = W_{N/2}$$

4.6. Using properties to derive FFT from DFT :

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}, \quad k = 0, 1, \dots, N-1 \tag{4.6.1}$$

$$= \sum_{n=even} x(n) W_N^{kn} + \sum_{n=odd} x(n) W_N^{kn} \tag{4.6.2}$$

$$= \sum_{m=0}^{2} x(2m) W_N^{2mk} + \sum_{m=0}^{2} x(2m+1) W_N^{(2m+1)k} \tag{4.6.3}$$

using property c, we get,

$$X(k) = \sum_{m=0}^{2} x(2m) W_{N/2}^{mk} + W_N^k \sum_{m=0}^{2} x(2m+1) W_{N/2}^{mk} \tag{4.6.4}$$

$$= X_1(k) + W_N^k X_2(k) \tag{4.6.5}$$

4.7.
- X_1(k) and X_2(k) are 3 point DFTs of x(2m) and x(2m+1) , m=0,1,2.
- X_1(k) and X_2(k) are periodic, Hence X_1(k+3) = X_1(k) and X_2(k+3) = X_1(k).
- By performing this step once we can see that number of operations have been reduced from N^2 to $\frac{N^2}{2}$.

4.8. Using the above properties recursively we have implemented radix-2 Fast-Fourier transform algorithm.

Algorithm	t($N = 128$)	t($N = 2048$)
DTFT	33.2 ms	7.36 s
FFT	1.54 ms	27.6 ms

- $t(N)$ corresponds to average time of execution for sample size of N.
- We can observe that as we increase N, the difference in execution times is drastically increasing.
- From our implementation of radix-2 FFT we can see that complexity is reduced from $O(n^2)$ to $O(n \log n)$