

**INSTITUTO
FEDERAL**

Santa Catarina

Câmpus
São José

Transformada de Fourier Discreta

Processamento de Sinais Digitais

Gabriel Luiz Espindola Pedro

18 de Março de 2024

Sumário

1. Questão 1	3
1.1. Reposta	3
1.2. Simulação	3
2. Questão 2	5
2.1. Reposta	5
2.1.1. Item a	5
2.1.2. Item b	5
2.1.3. Item c	5
2.1.3.1. Simulação	6
2.1.4. Item d	7
2.1.4.1. Simulação	8
3. Questão 3	9
3.1. Resposta	9
3.1.1. Simulação	10
4. Questão 4	11
5. Questão 5	13
5.1. Resposta	13
5.2. Simulação	14
6. Questão 6	15
6.1. Reposta	15
7. Questão 7	17
7.1. Respostas	17
7.1.1. Item a	17
7.1.2. Item b	17
8. Questão 8	18
8.1. Respostas	18
8.1.1. Item a	18
8.1.2. Item b	19
8.1.3. Item c	19
9. Questão 9	20
10. Questão 10	20
10.1. Função de convolução por sobreposição e armazenamento	20
10.2. Função de convolução por sobreposição e soma	20

1. Questão 1

As duas sequências de oito pontos $x_1[n]$ e $x_2[n]$ mostradas na figura a seguir tem DFT's $X_1[k]$ e $X_2[k]$, respectivamente. Determine a relação entre $X_1[k]$ e $X_2[k]$.

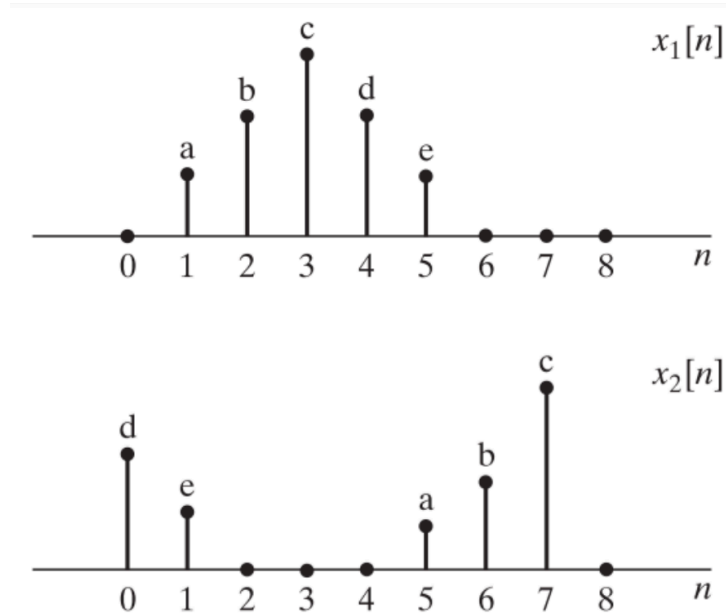


Figura 1: Definições de $x_1[n]$ e $x_2[n]$

1.1. Reposta

Ao compararmos os dois gráficos percebemos que a sequência $x_2[n]$ é a sequência $x_1[n]$ deslocada de 4 posições para a direita, analisamos também um janelamento de 8 pontos $[0, 7]$.

Obtendo a expressão de $x_1[n]$ e $x_2[n]$ em função dos impulsos unitários $\delta[n]$:

$$\begin{aligned} x_1[n] &= \delta[n-1] + 2\delta[n-2] + 3\delta[n-3] + 2\delta[n-4] + \delta[n-5] \\ x_2[n] &= 2\delta[n] + \delta[n-1] + \delta[n-5] + 2\delta[n-6] + 3\delta[n-7] \end{aligned} \quad (1)$$

Portanto podemos reescrever $X_2[k]$ em função de $X_1[k]$:

$$X_2[k] = X_1[k]e^{-j\frac{2\pi}{8}4k} \quad (2)$$

1.2. Simulação

Utilizando a linguagem de programação python, podemos simular a relação entre $X_1[k]$ e $X_2[k]$.

```
1 n = np.arange(0, 8) # Gera um vetor de 0 a 7
2
3 x1_n = [0, 1, 2, 3, 2, 1, 0, 0] # Definição de x1[n]
4 x2_n = [2, 1, 0, 0, 0, 1, 2, 3] # Definição de x2[n]
5
6 X1_k = fft(x1_n) * (
7     np.e ** (-1j * 2 * np.pi / 8 * 4 * n)
8 ) # Calcula a FFT de x1[n] e realiza deslocamento de 4
```

```

9
10 x2_hat_n = ifft(X1_k) # Calcula a IFFT

```

Onde $\hat{x}_2[n]$ é a sequência $x_2[n]$ reconstruída a partir de $X_1[k]$.

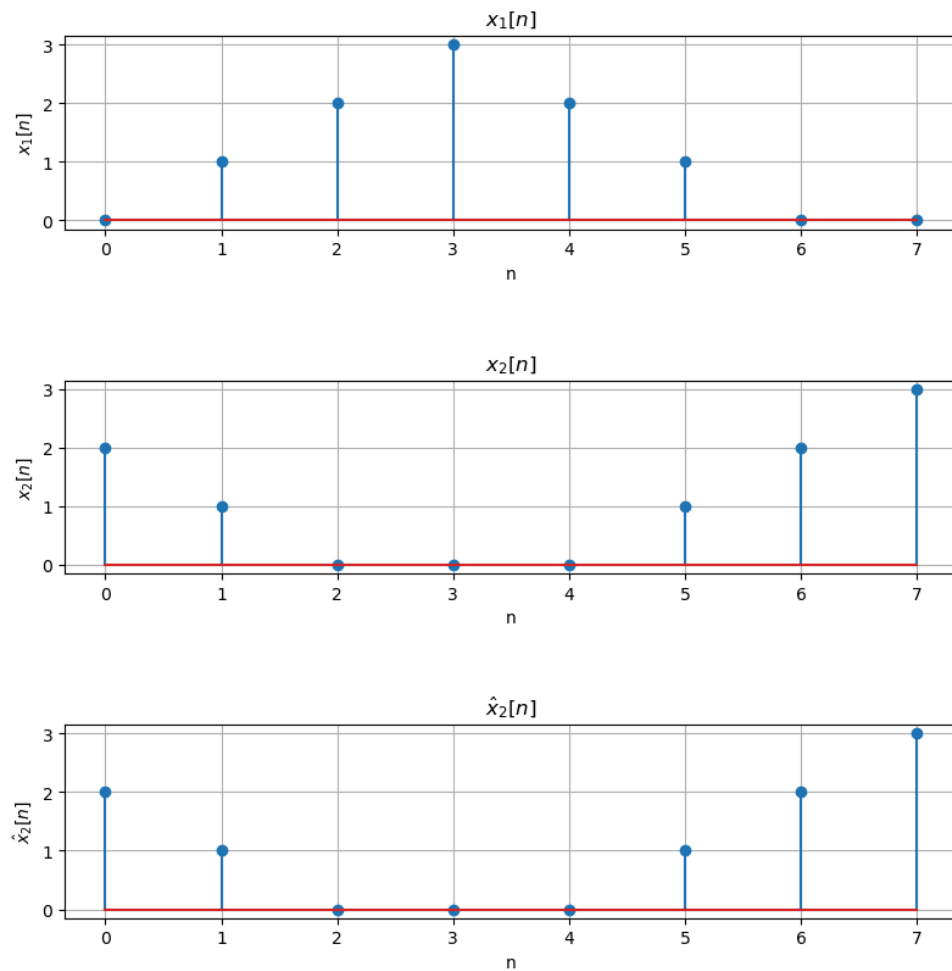


Figura 2: Plot de $x_1[n]$, $x_2[n]$ e $\hat{x}_2[n]$

Verificamos que como esperado $\hat{x}_2[n]$ é igual a $x_2[n]$.

2. Questão 2

Suponha que temos duas seqüências de quatro pontos $x[n]$ e $h[n]$, da seguinte forma:

$$\begin{aligned}x[n] &= \cos\left(\frac{\pi n}{2}\right) & n = 0, 1, 2, 3 \\h[n] &= 2^n & n = 0, 1, 2, 3\end{aligned}\tag{3}$$

Portanto temos:

- a) Calcule a *DFT* de quatro pontos $X[k]$.
- b) Calcule a *DFT* de quatro pontos $H[k]$.
- c) Calcule $y[n] = x[n] \otimes_4 h[n]$ (realizando a convolução circular diretamente).
- d) Calcule $y[n]$ do item anterior multiplicando as *DFT's* de $x[n]$ e $h[n]$ e realizando uma *DFT* inversa.

2.1. Reposta

A partir do enunciado podemos obter as seqüências $x[n]$ e $h[n]$:

$$\begin{aligned}x[n] &= \delta[n] - \delta[n - 2] \\h[n] &= \delta[n] + 2\delta[n - 1] + 4\delta[n - 2] + 8\delta[n - 3]\end{aligned}\tag{4}$$

2.1.1. Item a

Para calcular a *DFT* de quatro pontos $X[k]$ utilizamos a seguinte expressão:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}nk}\tag{5}$$

Substituindo os valores de $x[n]$ e $N = 4$:

$$\begin{aligned}X[k] &= 1e^{-j\frac{2\pi}{4}0k} + 0e^{-j\frac{2\pi}{4}1k} - 1e^{-j\frac{2\pi}{4}2k} + 0e^{-j\frac{2\pi}{4}3k} \\&= 1 - e^{-j\frac{2\pi}{4}2k}\end{aligned}\tag{6}$$

2.1.2. Item b

Realizando o mesmo procedimento para $H[k]$:

$$\begin{aligned}H[k] &= 1e^{-j\frac{2\pi}{4}0k} + 2e^{-j\frac{2\pi}{4}1k} + 4e^{-j\frac{2\pi}{4}2k} + 8e^{-j\frac{2\pi}{4}3k} \\&= 1 + 2e^{-j\frac{2\pi}{4}k} + 4e^{-j\frac{2\pi}{4}2k} + 8e^{-j\frac{2\pi}{4}3k}\end{aligned}\tag{7}$$

2.1.3. Item c

Para calcular $y[n] = x[n] \otimes_4 h[n]$ realizamos a convolução circular diretamente:

$$y[n] = \sum_{m=0}^{N-1} x[m] h[(n - m)_{\text{mod } 4}]\tag{8}$$

Para $N = 4$ então obtemos:

$$\begin{aligned}
y[n] &= x[0]h[n_{\text{mod } 4}] + x[1]h[(n-1)_{\text{mod } 4}] + \\
&\quad x[2]h[(n-2)_{\text{mod } 4}] + x[3]h[(n-3)_{\text{mod } 4}] \\
&= h[n_{\text{mod } 4}] - h[(n-2)_{\text{mod } 4}]
\end{aligned} \tag{9}$$

Portanto para os valores de 0 a 3 temos:

$$\begin{aligned}
y[0] &= h[0] - h[-2_{\text{mod } 4}] & y[2] &= h[2] - h[0_{\text{mod } 4}] \\
&= h[0] - h[2] & &= h[2] - h[0] \\
&= 1 - 4 & &= 4 - 1 \\
&= -3 & &= 3 \\
& & (10) & & (11) \\
y[1] &= h[1] - h[-1_{\text{mod } 4}] & y[3] &= h[3] - h[1_{\text{mod } 4}] \\
&= h[1] - h[3] & &= h[3] - h[1] \\
&= 2 - 8 & &= 8 - 2 \\
&= -6 & &= 6
\end{aligned}$$

Portanto $y[n]$ pode ser representado por:

$$y[n] = -3\delta[n] - 6\delta[n-1] + 3\delta[n-2] + 6\delta[n-3] \tag{12}$$

2.1.3.1. Simulação

Utilizado a biblioteca `scipy` do `python`, podemos simular a convolução circular de $x[n]$ e $h[n]$.

```

1  N = 4
2  n = np.arange(0, N)
3
4  x_n = cos(pi * n / 2)
5  h_n = 2**n
6
7  y_n = ndimage.convolve(x_n, h_n, mode="wrap", origin=-int(N / 2))

```

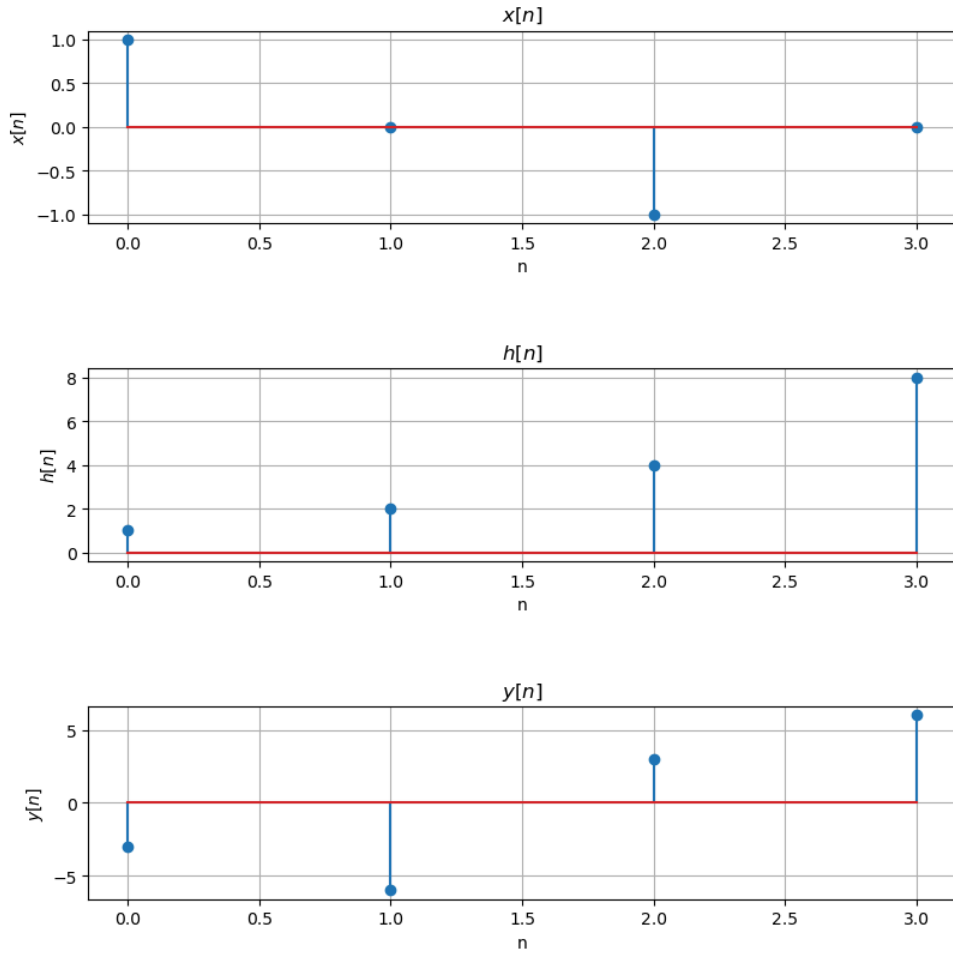


Figura 3: Plot de $x[n]$, $h[n]$ e $y[n]$

2.1.4. Item d

Com as *DFT*'s calculadas no item a e b, podemos calcular $y[n]$ multiplicando as *DFT*'s de $x[n]$ e $h[n]$ e realizando uma *DFT* inversa.

$$\begin{aligned}
 Y[k] &= X[k]H[k] \\
 &= (1 - e^{-j\frac{2\pi}{4}2k})H[k] \\
 &= H[k] - H[k]e^{-j\frac{2\pi}{4}2k} \\
 &\Downarrow \\
 y[n] &= h[n] - h[(n-2)_{\text{mod } 4}] \\
 &= \delta[n] + 2\delta[n-1] + 4\delta[n-2] + 8\delta[n-3] - \\
 &\quad \delta[(n-2)_{\text{mod } 4}] - 2\delta[(n-3)_{\text{mod } 4}] - 4\delta[(n-4)_{\text{mod } 4}] - 8\delta[(n-5)_{\text{mod } 4}] \\
 &= \delta[n] + 2\delta[n-1] + 4\delta[n-2] + 8\delta[n-3] - \\
 &\quad \delta[n-2] - 2\delta[n-3] - 4\delta[n] - 8\delta[n-1] \\
 &= -3\delta[n] - 6\delta[n-1] + 3\delta[n-2] + 6\delta[n-3]
 \end{aligned} \tag{13}$$

2.1.4.1. Simulação

Utilizando a biblioteca numpy do python, podemos calcular a DFT inversa de $X[k]H[k]$.

```
1 N = 4
2 n = np.arange(0, N)
3
4 x_n = cos(pi * n / 2)
5 h_n = 2**n
6
7 X_k = fft(x_n)
8 H_k = fft(h_n)
9
10 Y_k = X_k * H_k
11
12 y_n = ifft(Y_k)
```

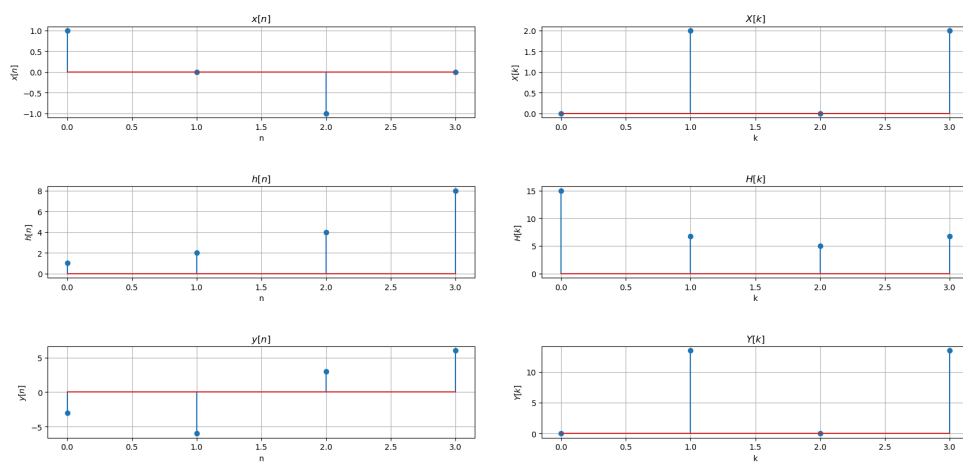


Figura 4: Plot de $x[n]$, $h[n]$ e $y[n]$ e suas respectivas DFT 's

3. Questão 3

Dois sinais de comprimento finito, $x_1[n]$ e $x_2[n]$, são esboçados na figura a seguir. Suponha que $x_1[n]$ e $x_2[n]$ sejam nulos fora da região mostrada na figura. Seja $x_3[n]$ a convolução circular de oito pontos de $x_1[n]$ e $x_2[n]$. Determine $x_3[2]$.

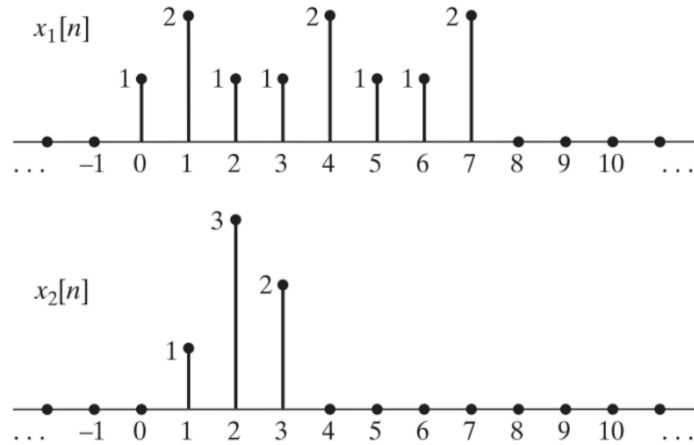


Figura 5: Definições de $x_1[n]$ e $x_2[n]$

3.1. Resposta

Analisando a figura, podemos obter as sequências $x_1[n]$ e $x_2[n]$:

$$x_1[n] = \delta[n] + 2\delta[n-1] + \delta[n-2] + \delta[n-3] + 2\delta[n-4] + \delta[n-5] + \delta[n-6] + 2\delta[n-7] \quad (14)$$

$$x_2[n] = \delta[n-1] + 3\delta[n-2] + 2\delta[n-3]$$

Utilizando a formula da convolução circular sendo $N = 8$:

$$x_3[n] = \sum_{m=0}^{N-1} x_1[m]x_2[(n-m)_{\text{mod } N}] \quad (15)$$

Podemos então obter $x_3[2]$:

$$\begin{aligned} x_3[2] &= x_1[0]x_2[(2-0)_{\text{mod } 8}] + x_1[1]x_2[(2-1)_{\text{mod } 8}] + \\ &\quad x_1[2]x_2[(2-2)_{\text{mod } 8}] + x_1[3]x_2[(2-3)_{\text{mod } 8}] + \\ &\quad x_1[4]x_2[(2-4)_{\text{mod } 8}] + x_1[5]x_2[(2-5)_{\text{mod } 8}] + \\ &\quad x_1[6]x_2[(2-6)_{\text{mod } 8}] + x_1[7]x_2[(2-7)_{\text{mod } 8}] \\ &= x_1[0]x_2[2] + x_1[1]x_2[1] + x_1[2]x_2[0] + x_1[3]x_2[7] + \\ &\quad x_1[4]x_2[6] + x_1[5]x_2[5] + x_1[6]x_2[4] + x_1[7]x_2[3] \\ &= 1 \cdot 3 + 2 \cdot 1 + 1 \cdot 0 + 1 \cdot 0 + 2 \cdot 0 + 1 \cdot 0 + 1 \cdot 0 + 2 \cdot 2 \\ &= 3 + 2 + 0 + 0 + 0 + 0 + 0 + 4 \\ &= 9 \end{aligned} \quad (16)$$

Portanto $x_3[2] = 9$.

3.1.1. Simulação

Utilizando python simulamos a operação

```
1 N = 8
2 n = np.arange(0, N)
3
4 x1_n = [1, 2, 1, 1, 2, 1, 1, 2]
5 x2_n = [0, 1, 3, 2, 0, 0, 0, 0]
6
7 x3_n = ndimage.convolve(x1_n, x2_n, mode="wrap", origin=-int(N / 2))
```

Obtemos o seguinte resultado

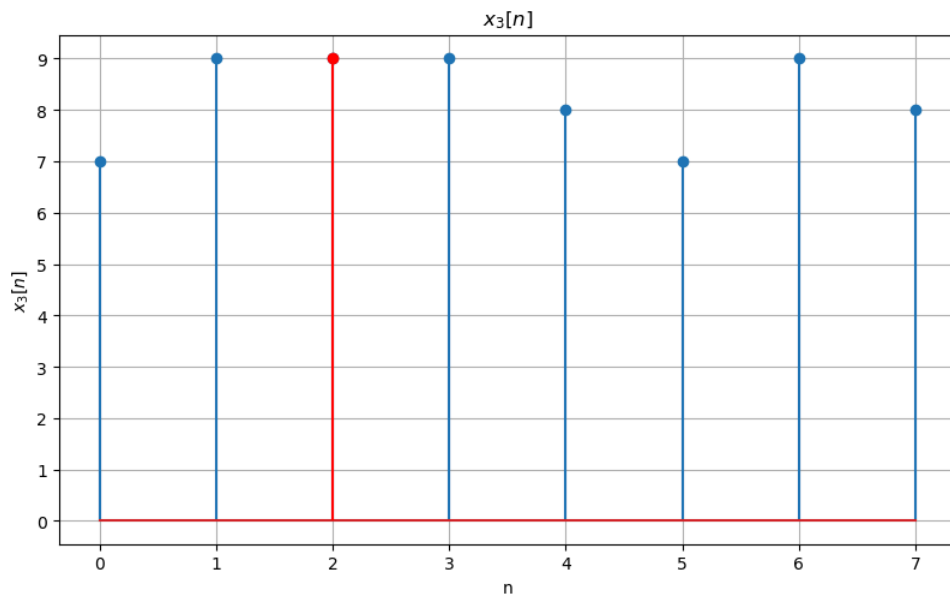
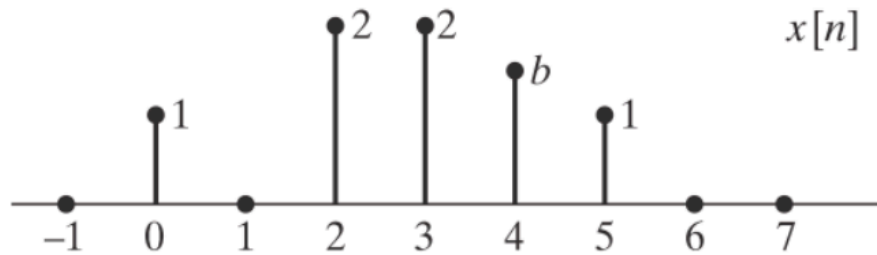


Figura 6: Plot de $x_3[n]$ evidenciando o valor em $x_3[2]$

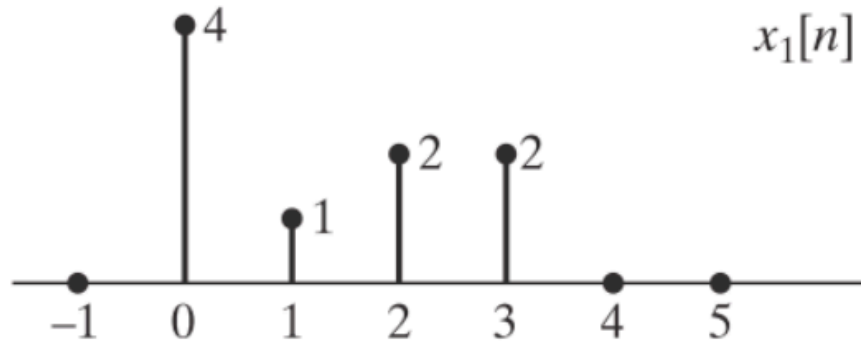
4. Questão 4

Na figura a seguir é mostrada uma sequência de tempo discreto com seis pontos $x[n] = 0$ fora do intervalo mostrado. O valor de $x[4]$ não é conhecido e é representado como b . Observe que a amostra mostrada como b na figura não está necessariamente na escala. Sejam $X(e^{j\omega})$ a *TFTD* de $x[n]$ e $X_1[k]$ as amostras de $X(e^{j\omega})$ em a cada $\pi/2$, isto é,

$$X_1[k] = X(e^{j\omega}) \Big|_{\omega=k\pi/2} \quad 0 \leq k \leq 3$$



A sequência com quatro pontos $x_1[n]$ que resulta da inversa com quatro pontos de $X_1[k]$ é mostrada a seguir. Com base nessa figura, você pode determinar b de modo único? Caso afirmativo, dê esse valor de b .



Definimos $x_1[n]$ a partir da sequência mostrada na figura como:

$$x_1[n] = 4\delta[n] + \delta[n-1] + 2\delta[n-2] + 2\delta[n-3] \quad (17)$$

Com essa informação podemos determinar a *DFT* de 4 pontos da sequência:

$$X_1[k] = 4 + e^{-j\frac{2\pi}{4}k} + 2e^{-j\frac{2\pi}{4}2k} + 2e^{-j\frac{2\pi}{4}3k} \quad (18)$$

Observando a sequência $x[n]$ dada na primeira imagem podemos defini-la em termos de função:

$$x[n] = \delta[n] + 2\delta[n-2] + 2\delta[n-3] + b\delta[n-4] + \delta[n-5] \quad (19)$$

Portanto a DFT de 4 pontos de $x[n]$ é:

$$X_1[k] = 1 + 2e^{-j\frac{2\pi}{4}2k} + 2e^{-j\frac{2\pi}{4}3k} + b + e^{-j\frac{2\pi}{4}k} \quad (20)$$

Comparando as duas expressões de $X_1[k]$ podemos obter o valor de b :

$$\begin{aligned} 4 + e^{-j\frac{2\pi}{4}k} + 2e^{-j\frac{2\pi}{4}2k} + 2e^{-j\frac{2\pi}{4}3k} \\ = \\ 1 + 2e^{-j\frac{2\pi}{4}2k} + 2e^{-j\frac{2\pi}{4}3k} + b + e^{-j\frac{2\pi}{4}k} \end{aligned} \quad (21)$$

Anulando os termos comuns obtemos:

$$4 = 1 + b \therefore \boxed{b = 3} \quad (22)$$

5. Questão 5

Na figura a seguir são mostradas duas seqüências de comprimento finito $x_1[n]$ e $x_2[n]$. Qual é o menor N tal que a convolução circular de N pontos de $x_1[n]$ e $x_2[n]$ seja igual à convolução linear dessas seqüência?

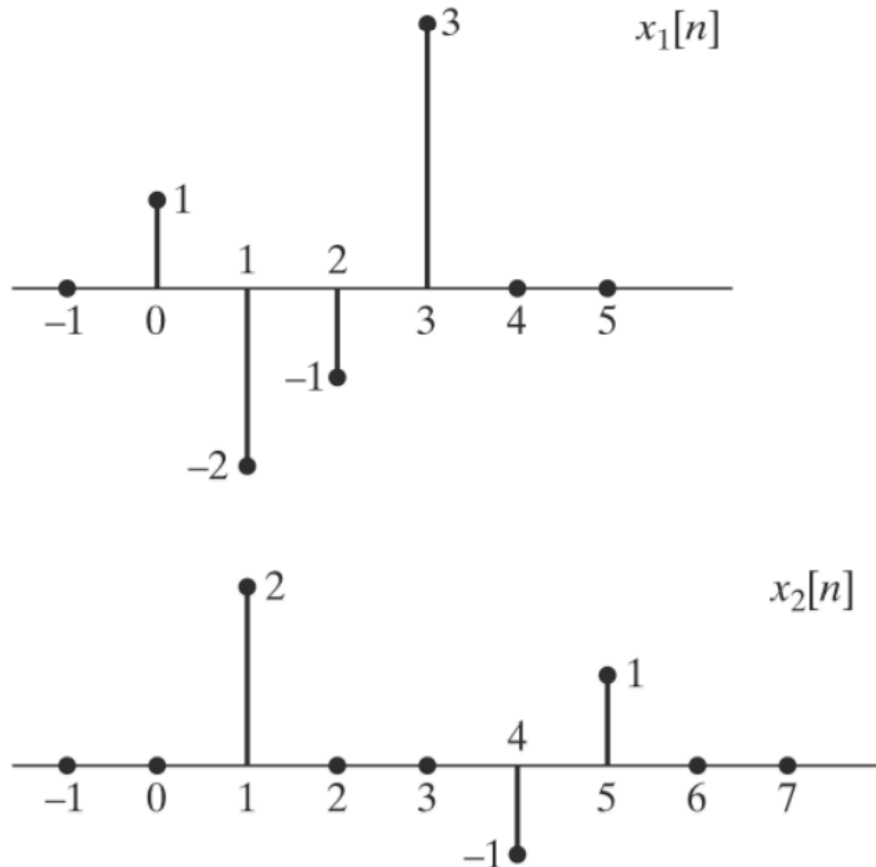


Figura 9: Definições de $x_1[n]$ e $x_2[n]$

5.1. Resposta

A partir da figura definimos as seqüências $x_1[n]$ e $x_2[n]$:

$$\begin{aligned} x_1[n] &= \delta[n] - 2\delta[n-1] - \delta[n-2] + 3\delta[n-3] \\ x_2[n] &= 2\delta[n-1] - \delta[n-4] + \delta[n-5] \end{aligned} \quad (23)$$

Verificamos pelas imagens também que as respectivas janelas que contem valores são 4 para $x_1[n]$ e 6 para $x_2[n]$.

Portanto utilizando a fórmula do tamanho da convolução linear:

$$N = N_1 + N_2 - 1 \quad (24)$$

Obtemos que o menor N tal que a convolução circular de N pontos de $x_1[n]$ e $x_2[n]$ seja igual à convolução linear dessas seqüência é 9.

5.2. Simulação

Para validar os cálculos realizados simulamos ambas as convoluções e as comparamos. Será realizado utilizando python.

```
1 N = 9
2 n = np.arange(0, N)
3
4 x1_n = [1, -2, -1, 3, 0, 0, 0, 0, 0]
5 x2_n = [0, 2, 0, 0, -1, 1, 0, 0, 0]
6
7 circ_conv = ndimage.convolve(x1_n, x2_n, mode="wrap", origin=-int(N / 2))
8 linear_conv = signal.convolve(x1_n, x2_n)
```

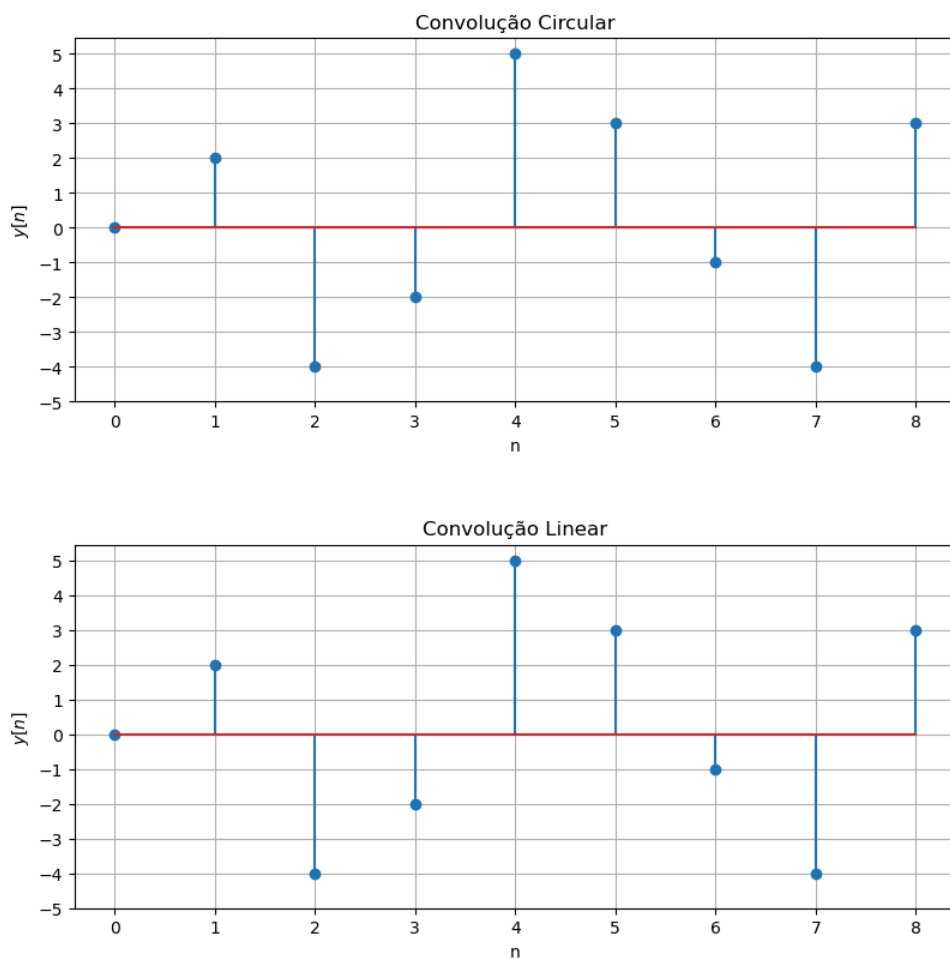
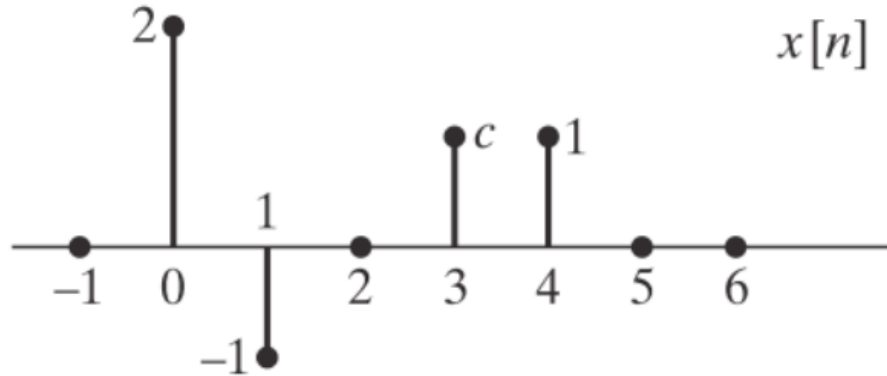


Figura 10: Plot das convoluções circular e linear de $x_1[n]$ e $x_2[n]$

Como observamos acima, a convolução circular de 9 pontos de $x_1[n]$ e $x_2[n]$ é igual à convolução linear dessas sequências.

6. Questão 6

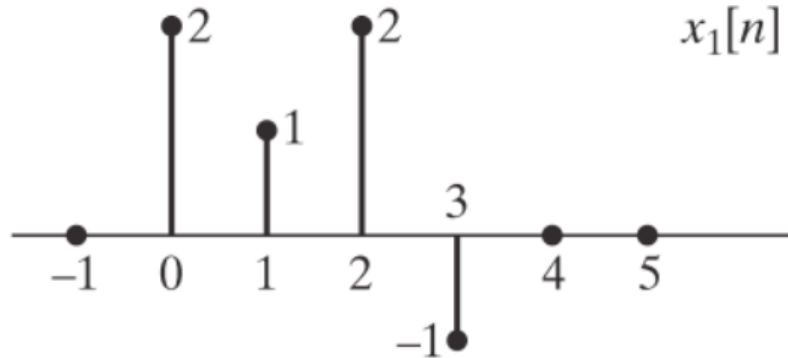
Na figura a seguir é mostrada uma sequência $x[n]$ para a qual o valor de $x[3]$ é uma constante desconhecida c .



O valor da amostra com amplitude c não está necessariamente representada na escala. Considere:

$$X_1[k] = X[k]e^{j\frac{2\pi}{5}2k} \quad (25)$$

Sendo $X[k]$ a *DFT* de cinco pontos de $x[n]$. A sequência $x_1[n]$ representada na figura a seguir é a *DFT* inversa de $X_1[k]$. Qual o valor de c ?



6.1. Reposta

Obtendo as sequências $x[n]$ e $x_1[n]$ a partir das figuras:

$$x[n] = 2\delta[n] - \delta[n-1] + c\delta[n-3] + \delta[n-4] \quad (26)$$

$$x_1[n] = 2\delta[n] + \delta[n-1] + 2\delta[n-2] - \delta[n-3]$$

Sabendo que a multiplicação por $e^{j\frac{2\pi}{5}2k}$ em $X[k]$ resulta em um deslocamento de 2 posições para a direita, podemos definir a relação entre $x_1[n]$ e $x[n]$:

$$x_1[n] = x[(n-2)_{\text{mod}(5)}] \quad (27)$$

Portanto reescrevendo $x \left[(n-2)_{\text{mod}(5)} \right]$:

$$\begin{aligned}
 x \left[(n-2)_{\text{mod}(5)} \right] &= 2\delta[n-2] - \delta[n-3] + c\delta \left[(n-5)_{\text{mod}(5)} \right] + \delta \left[(n-6)_{\text{mod}(5)} \right] \\
 &= 2\delta[n-2] - \delta[n-3] + c\delta[n] + \delta[n-1] \\
 &= c\delta[n] + \delta[n-1] + 2\delta[n-2] - \delta[n-3]
 \end{aligned} \tag{28}$$

Realizando a comparação com $x_1[n]$ obtemos:

$$\begin{aligned}
 x_1[n] &= 2\delta[n] + \delta[n-1] + 2\delta[n-2] - \delta[n-3] \\
 &= \\
 x \left[(n-2)_{\text{mod}(5)} \right] &= c\delta[n] + \delta[n-1] + 2\delta[n-2] - \delta[n-3]
 \end{aligned} \tag{29}$$

Cancelando os termos comuns obtemos:

$$c = 2 \tag{30}$$

7. Questão 7

Suponha que tenhamos uma sequência de 1025 pontos de dados (1 a mais que $N = 2^{10}$). Em vez de descartar o valor final, vamos preencher a sequência com zeros até que seu comprimento seja de $N = 2^{11}$, de modo que possamos usar um algoritmo *FFT* de raiz 2.

- a) Quantas multiplicações complexas são necessárias para se computar a *DFT* usando um algoritmo de *FFT* raiz 2?
- b) Quantas multiplicações complexas seriam necessárias para se computar diretamente a *DFT* de 1025 pontos?

7.1. Respostas

7.1.1. Item a

Utilizando a fórmula para o cálculo do número de multiplicações complexas necessárias para se computar a *DFT* usando um algoritmo de *FFT* raiz 2:

$$\frac{N}{2} \log_2 N \quad (31)$$

Sendo $N = 2^{11}$:

$$\frac{2^{11}}{2} \log_2 2^{11} = \boxed{11.264} \quad (32)$$

Portanto são necessárias 11264 multiplicações complexas.

7.1.2. Item b

Para calcular diretamente a *DFT* de 1025 pontos, utilizamos a fórmula:

$$N^2 \quad (33)$$

Sendo $N = 1025$:

$$1025^2 = \boxed{1050625} \quad (34)$$

Portanto são necessárias 1050625 multiplicações complexas.

8. Questão 8

Considere a sequência de comprimento finito real $x[n]$ mostrada na figura a seguir.

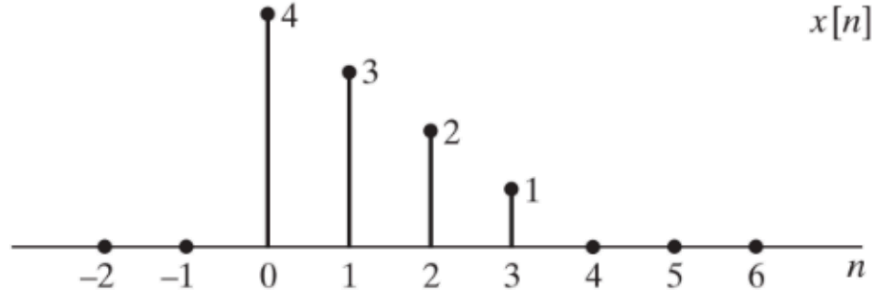


Figura 13: Definições de $x[n]$

a) Esboce a sequência de comprimento finito $y[n]$ cuja *DFT* de seis pontos seja

$$Y[k] = W_6^{5k} X[k] \quad (35)$$

Sendo $X[k]$ a *DFT* de seis pontos de $x[n]$.

b) Esboce a sequência de comprimento finito $w[n]$ cuja *DFT* de seis pontos seja

$$W[k] = \Im\{X[k]\} \quad (36)$$

c) Esboce a sequência de comprimento finito $q[n]$ cuja *DFT* de três pontos seja

$$Q[k] = X[2k + 1] \quad k = 0, 1, 2 \quad (37)$$

8.1. Respostas

8.1.1. Item a

A partir da figura podemos obter a sequência $x[n]$:

$$x[n] = 4\delta[n] + 3\delta[n - 1] + 2\delta[n - 2] + \delta[n - 3] \quad (38)$$

Sabendo que $W_6^{5k} = e^{-j\frac{2\pi}{6}5k}$ e que a multiplicação por $e^{-j\frac{2\pi}{6}5k}$ em $X[k]$ resulta em um deslocamento de 5 posições para a direita, podemos facilmente relacionar $y[n]$ e $x[n]$:

$$y[n] = x[(n - 5)_{\text{mod } 6}] \quad (39)$$

Portanto reescrevendo $x[(n - 5)_{\text{mod } 6}]$ obtemos $y[n]$:

$$\begin{aligned} y[n] &= x[(n - 5)_{\text{mod } 6}] = 4\delta[n - 5] + 3\delta[(n - 6)_{\text{mod } 6}] + \\ &\quad 2\delta[(n - 7)_{\text{mod } 6}] + \delta[(n - 8)_{\text{mod } 6}] \\ &= 4\delta[n - 5] + 3\delta[n] + 2\delta[n - 1] + \delta[n - 2] \\ &= 3\delta[n] + 2\delta[n - 1] + \delta[n - 2] + 4\delta[n - 5] \end{aligned} \quad (40)$$

8.1.2. Item b

Calculamos a *DFT* de seis pontos de $x[n]$:

$$X[k] = 4 + 3W_6^k + 2W_6^{2k} + W_6^{3k} \quad (41)$$

Sabendo que:

$$W_6 = e^{-j\frac{2\pi}{6}} \quad e^{-j\theta} \quad \Im\{e^{-j\theta}\} = -\sin(\theta) \quad (42)$$

Podemos obter a sequência $W[k]$:

$$\begin{aligned} W[k] &= \Im\{X[k]\} \\ &= -3 \sin\left[\frac{2\pi}{6}k\right] - 2 \sin\left[\frac{2\pi}{6}2k\right] - \sin\left[\frac{2\pi}{6}3k\right] \\ &= -3 \sin\left[\left(\frac{\pi}{3}\right)k\right] - 2 \sin\left[\left(\frac{2\pi}{3}\right)k\right] - \sin[\pi k] \end{aligned} \quad (43)$$

8.1.3. Item c

Para obter a sequência $q[n]$ a partir de $x[n]$ utilizamos a expressão:

$$Q[k] = X[2k + 1] \quad k = 0, 1, 2 \quad (44)$$

E sendo $X[k]$ a *DFT* de seis pontos de $x[n]$:

$$X[k] = 4 + 3W_6^k + 2W_6^{2k} + W_6^{3k} \quad (45)$$

Portanto obtemos:

$$\begin{aligned} Q[0] &= X[1] \\ &= 3W_6^1 + 2W_6^2 + W_6^3 \\ &= 3W_6 + 2W_6^2 + W_6^3 \\ Q[1] &= X[3] \\ &= 3W_6^3 + 2W_6^6 + W_6^9 \\ &= 3W_6^3 + 2W_6^0 + W_6^3 \\ Q[2] &= X[5] \\ &= 3W_6^5 + 2W_6^{10} + W_6^{15} \\ &= 3W_6^5 + 2W_6^4 + W_6^3 \end{aligned} \quad (46)$$

Portanto $q[n]$ pode ser representado por:

$$q[n] = 3\delta[n - 1] + 2\delta[n - 2] + \delta[n - 3] \quad (47)$$

9. Questão 9

Realize a simulação das questões utilizando software.

10. Questão 10

Comente os códigos feitos no MATLAB dos dois métodos de convolução fornecido pela professora. Faça testes usando essas funções fornecidas e compare com os resultados das funções cconv e conv.

10.1. Função de convolução por sobreposição e armazenamento

```
1 function y=sobreposicao_armazena(x,h,N)
2
3 % Define o número de blocos necessários para realizar a convolução
4 L = length(x)
5 K = length(h)
6 B = ceil((L + K - 1)/(N - K + 1))
7
8 % Adiciona zeros para garantir que haja espaço suficiente para sobreposição
9 % e soma durante a convolução.
10 x=[zeros(1,K-1) x zeros(1,B*(N-K+1))]
11 hm = [h zeros(1,N-K)]
12
13 % Este loop divide o sinal de entrada x em blocos de tamanho N, com uma
14 % sobreposição de K-1 amostras entre os blocos.
15 for i = 1:B
16     % Armazena os blocos de tamanho N
17     X(i,:) = [x(1+(i-1)*(N-(K-1)):(i*N-(i-1)*(K-1))];
18 end
19
20 % Realiza a convolução circular do primeiro bloco X com a resposta ao
21 % impulso estendida hm e, em seguida, selecionam apenas a parte relevante do
22 % resultado, removendo as amostras extras causadas pela convolução circular.
23 y = cconv(X(1,:),hm,N)
24 y = y(K:N)
25
26 % Este loop realiza o mesmo processo para os blocos restantes e concatena
27 % os resultados ao longo do tempo.
28 for i = 2:B
29     y_aux = cconv(X(i,:),hm,N)
30     y = [y y_aux(K:N)]
31 end
```

10.2. Função de convolução por sobreposição e soma

```
1 function [yconv,yfft]=sobreposicao_soma(x,h,N)
2
3 % Determina o número de blocos necessários para dividir o sinal de entrada
4 % em blocos de tamanho N
5 t_x = length(x);
6 t_h = length(h);
7 blocos = t_x/N;
8
9 % Adiciona zeros para acomodar a resposta ao impulso h.
10 for i = 1:blocos
```

```

10     X(i,:) = [x(1+(i-1)*N:i*N) zeros(1,t_h-1)];
11 end
12 hm = [h zeros(1,N-1)];
13
14 % Calcula a convolução por dois métodos diferentes: convolução circular
15 % e FFT.
16 for i = 1:blocos
17     Y(i,:) = [zeros(1,(i-1)*N) cconv(X(i,:),hm,N+t_h-1) zeros(1,t_x-(i)*N)];
18     YY(i,:) = [zeros(1,(i-1)*N) ifft(fft(X(i,:)).*fft(hm)) zeros(1,t_x-
19     (i)*N)];
20 end
21
22 % Este loops somam os resultados de convolução de todos os blocos para obter
23 % o sinal de saída final tanto para a abordagem de sobreposição e soma quanto
24 % para a FFT.
25 yconv = zeros(1,t_x+t_h-1);
26 yfft = zeros(1,t_x+t_h-1);
27
28 for i = 1:blocos
29     yconv = yconv+Y(i,:);
30     yfft = yfft+YY(i,:);
31 end

```