2.10 A Transformada de Fourier discreta (TFD)

EPUSP - PTC 3424, maio de 2017. Profa. Maria D. Miranda *

- A transformada de Fourier de tempo discreto (TFTD) é uma função contínua em ω, portanto, não é computacionalmente eficiente. Não serve para o processamento em tempo discreto de sinal em um computador digital.

- A série de Fourier de tempo discreto (SFD) relaciona sequências periódicas com comprimento infinito. Porém, nem todo sinal é periódico.

- É conveniente mapear uma sequência de comprimento finito no domínio do tempo a uma sequência de mesmo comprimento no domínio da frequência e vice-versa.

- A Transformada de Fourier Discreta (TFD) é uma transformada que relaciona uma sequência finita no domínio do tempo com uma outra sequência finita no domínio da frequência.

Definição da TFD

A TFD da sequência $x(n)$ de comprimento N é definida por

$$\boxed{X_N(k) \triangleq \sum_{n=0}^{N-1} x(n) e^{-j\frac{2\pi}{N}kn}}$$

para $0 \leq k \leq N-1$ e $X_N(k) = 0$ para os demais valores de k.

A transformação inversa da TFD é

$$\boxed{x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X_N(k) e^{j\frac{2\pi}{N}kn}}$$

para $0 \leq n \leq N-1$ e $x(n) = 0$ para os demais valores de n.

Notação compacta: É usual a notação $W_N = e^{-j\frac{2\pi}{N}}$. Assim, $e^{-j\frac{2\pi}{N}kn} = W_N^{kn}$

Partes do texto e figuras contaram com a colaboração do doutorando Flávio Renê M. Pavan

Exemplo
Considere

$$x(n) = \delta(n) + \delta(n-1) + \delta(n-2) + \delta(n-3) + \delta(n-4).$$

Aplicando a definição para $N = 5$ temos

$$X_5(k) = \sum_{n=0}^{4} e^{-j\frac{2\pi}{5}kn}$$

Após manipulações algébricas resulta $X(k) = 5$ para $k = 0$ e $X(k) = 0$ para $k = 1, 2, 3$ e 4. Como interpretar esse resultado?

Na Figura 1 está ilustrado a TFTD$\{x(n)\}$ e a TFD para diferentes valores de N. Note que $x(n)$ tem uma duração de $L = 5$ amostras, porém a sua TFD está sendo calculada com 10, 15 e 50 amostras. Como $x(n)$ de comprimento $L < N$, as amostras faltantes até completar N são tomadas iguais a 0. Os N pontos da TFD são N amostras de um período da TFTD $\omega \in [0, 2\pi)$. Assim, quanto maior o número de pontos usados para calcular da TFD de $x(n)$ mais a TFD se aproxima da TFTD de $x(n)$.

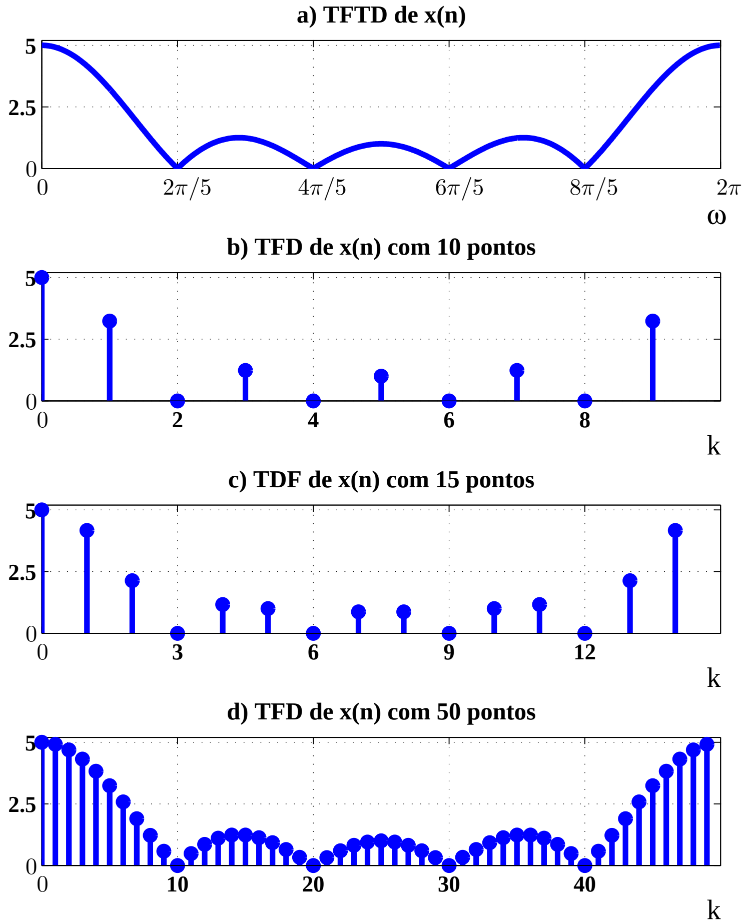


Figura 1: Representações em frequência para $x(n) = \delta(n) + \delta(n-1) + \delta(n-2) + \delta(n-3) + \delta(n-4)$: a) TFTD$\{x(n)\}$; b) TFD$_{10}\{x(n)\}$; c) TFD$_{15}\{x(n)\}$ e d) TFD$_{50}\{x(n)\}$.

A TFD pode ser obtida de forma eficiente usando os algoritmos de fft (*fast Fourier transform*). No MatLab esse cálculo é feito com a função fft(sinal, N) em que sinal representa a sequência de tempo discreto e N o número de pontos em que se deseja obter a TFD. No caso do Exemplo da Figura 1, esse foi o comando usado para obter as DFTs dos gráficos (b), (c) e (d). Particularmente usou-se os seguintes comandos no MatLab:

```
x=[ 1 1 1 1 1];
X=fft(x,N);
stem([0:N-1], abs(X))
```

para $N = 10$, $N = 15$ e $N = 50$.

Cálculo computacional da Transformada de Fourier discreta (TFD)

Por conveniência vamos reescrever a TFD de N pontos como

$$X_N(k) = \sum_{n=0}^{N-1} x(n) W_N^{k,n}$$

sendo

$$W_N^{k,n} \triangleq e^{-\frac{2\pi}{N}kn}$$

No caso de uma TFD de $N = 8$ temos as seguintes operações

$$\underbrace{\begin{bmatrix} X_8(0) \\ X_8(1) \\ X_8(2) \\ X_8(3) \\ X_8(4) \\ X_8(5) \\ X_8(6) \\ X_8(7) \end{bmatrix}}_{\mathbf{X}_8(k)} = \underbrace{\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & W_8^{1,1} & W_8^{1,2} & W_8^{1,3} & W_8^{1,4} & W_8^{1,5} & W_8^{1,6} & W_8^{1,7} \\ 1 & W_8^{2,1} & W_8^{2,2} & W_8^{2,3} & W_8^{2,4} & W_8^{2,5} & W_8^{2,6} & W_8^{2,7} \\ 1 & W_8^{3,1} & W_8^{3,2} & W_8^{3,3} & W_8^{3,4} & W_8^{3,5} & W_8^{3,6} & W_8^{3,7} \\ 1 & W_8^{4,1} & W_8^{4,2} & W_8^{4,3} & W_8^{4,4} & W_8^{4,5} & W_8^{4,6} & W_8^{4,7} \\ 1 & W_8^{5,1} & W_8^{5,2} & W_8^{5,3} & W_8^{5,4} & W_8^{5,5} & W_8^{5,6} & W_8^{5,7} \\ 1 & W_8^{6,1} & W_8^{6,2} & W_8^{6,3} & W_8^{6,4} & W_8^{6,5} & W_8^{6,6} & W_8^{6,7} \\ 1 & W_8^{7,1} & W_8^{7,2} & W_8^{7,3} & W_8^{7,4} & W_8^{7,5} & W_8^{7,6} & W_8^{7,7} \end{bmatrix}}_{\mathbf{W}_{8\times 8}} \underbrace{\begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \\ x(7) \end{bmatrix}}_{\mathbf{x}_8(k)} \tag{1}$$

Genericamente de forma compacta para um número arbitrário N de pontos a TFD pode ser convenientemente representada como

$$\mathbf{X}_N(k) = \mathbf{W}_{N\times N}\,\mathbf{x}_N(n)$$

sendo $\mathbf{W}_{N\times N}$ uma matriz com N linha e N colunas. Nota-se que $\mathbf{W}_{N\times N}$ é uma matriz cujo inverso é apenas o seu complexo conjugado transposto $\mathbf{W}_{N\times N}^H$. Para um número arbitrário de N pontos a TFD inversa pode ser representada como

$$\mathbf{x}_N(n) = \frac{1}{N}\,\mathbf{W}_{N\times N}^H\,\mathbf{X}_N(k).$$

Nota-se que para cada N pontos de TDF são necessárias $\begin{cases} (N-1)^2 & \textbf{multiplicações complexas} \\ N(N-1) & \textbf{somas complexas} \end{cases}$

Em algumas aplicações práticas esse número de pontos é proibitivo. Esse problema foi parcialmente resolvido em 1965 por Cooley (IBM) e Tukey (Princeton) (1965). Eles observaram que a A TFD de N pontos pode ser calculada como 2 TFD's de $N/2$ pontos. Especificamente, se considerarmos as amostras de $x(n)$ de índice par e de índice ímpar separadamente, ou seja,

$$\begin{aligned} a(n) =& \quad x(2n) \\ b(n) =& \quad x(2n+1) \end{aligned}$$

temos

$$X_N(k) = \sum_{n=0}^{\frac{N}{2}-1} a(n) W_N^{k,2n} + \sum_{n=0}^{\frac{N}{2}-1} b(n) W_N^{k,2n+1}$$

Observando que

$$\begin{aligned} W_N^{k,2n} &= W_{N/2}^{k,n} \\ W_N^{k,2n+1} &= W_N^k \, W_{N/2}^{k,n} \end{aligned}$$

Com isso podemos escrever

$$X_N(k) = \sum_{n=0}^{\frac{N}{2}-1} a(n) W_{N/2}^{k,n} + W_N^k \sum_{n=0}^{\frac{N}{2}-1} b(n) W_{N/2}^{k,n}$$

Definindo

$$\begin{aligned} G_{N/2}(k) &= \sum_{n=0}^{\frac{N}{2}-1} a(n) W_{N/2}^{k,n} \\ H_{N/2}(k) &= \sum_{n=0}^{\frac{N}{2}-1} b(n) W_{N/2}^{k,n} \end{aligned}$$

temos

$$X_N(k) = G_{N/2}(k) + W_N^k H_{N/2}(k)$$

que representa uma TFD de N pontos calculada como duas TFD's de $N/2$ pontos.

No caso do exemplo teríamos uma TFD com 8 pontos calculada como a soma de duas TFD's de quatro pontos

$$X_8(k) = G_4(k) + W_8^k H_4(k)$$

e cada TFD de quadro pontos calculada como a soma de duas TFD's de dois pontos

$$X_4(k) = G_2(k) + W_4^k H_2(k)$$

Algoritmos que calculam a TFD baseada na decomposição de TFD's de sequências menores são chamados de algoritmos rápidos de DFT (fft, *Fast Fourier Transform*) baseados na dizimação no tempo. Detalhes sobre esse e outros tipos de algoritmos de fft podem ser encontrados na terceira edição do livro do Oppenheim, *Discrete-time Signal Processing*, Capítulo 9.

Custo computacional da TFD

O custo computacional para calcular a DFT de N pontos usando diretamente a definição é proporcional a $(N-1)^2$. O custo computacional para o mesmo cálculo com os algoritmos de fft é proporcional $N \log N$. Nas curvas da Figura 2 é possível notar que quanto maior o número de pontos considerado para o cálculo da TFD maior a vantagem do ponto de vista de custo computacional para os algoritmos de fft.

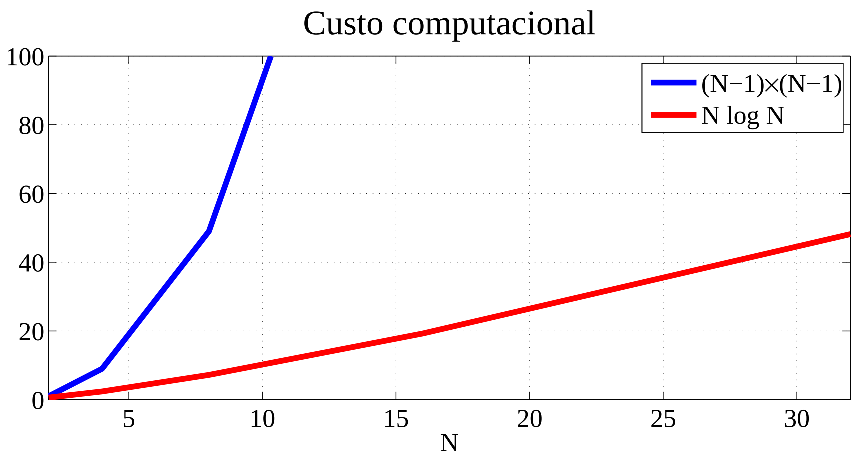


Figura 2: Custo computacional para o cálculo da TDF N pontos usando diretamente a definição (curva azul) e usando algoritmos rápidos (curva vermelha).

Lembre que:

- No MatLab esse cálculo é feito com a função fft(sinal, N) em que sinal representa a sequência de tempo discreto e N o número de pontos em que se deseja obter a TFD. Se o comprimento da sequência sinal é maior que N o MatLab inclui zeros e se for dado um N menor que o comprimento da sequência sinal ela será truncada.
- A TFD com N pontos de $x(n)$, $\text{TFD}_N\{x(n)\} = X_N(k)$ representa N pontos da TFTD de $x(n)$, $\text{TFTD}\{x(n)\} = X(e^{j\omega})$, para $\omega \in [0, 2\pi)$.
- Quanto maior o número de pontos acrescentado para obter $X_N(k)$ mais essa TDF se parece com a TFTD de $x(n)$.
- A quantidade de zeros acrescentadas no cálculo da TFD depende do custo computacional permitido para a aplicação em questão, pois quanto maior o número de pontos, maior o número de multiplicações e somas da TFD.

Relação entre TFD e TFTD

Seja $x(n)$ uma sequência de comprimento N, tal que $x(n) = 0$ para $n < 0$ e $n > N-1$.

- TFTD: $X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} = \sum_{n=0}^{N-1} x(n)e^{-j\omega n}$
- TFD: $X(k) = \sum_{n=0}^{N-1} x(n)e^{-j\frac{2\pi}{N}kn}$, para $0 \leq k \leq N-1$

Comparando as expressões de $X(e^{j\omega})$ e $X(k)$, obtém-se

$$\boxed{X(e^{j\omega})|_{\omega=\frac{2\pi}{N}k} = X(k)} \quad \text{para} \quad 0 \leq k \leq N-1$$

A amostragem da TFTD resulta na superposição periódica no intervalo $[0, N-1]$ da correspondente sequência no domínio do tempo.

Relação entre TFD e SFD

$$\begin{array}{ccc} x(n) & \underset{\text{um período}}{\overset{\text{repetição periódica}}{\rightleftarrows}} & \tilde{x}(n) \\ \text{TFD}_N \updownarrow & & \updownarrow \text{SFD} \\ X(k) & \underset{\text{um período}}{\overset{\text{repetição periódica}}{\rightleftarrows}} & \tilde{X}(k) \end{array}$$

Figura 3: Relação entre TFD e SFD

Exemplo: superposição periódica, $N = 12$

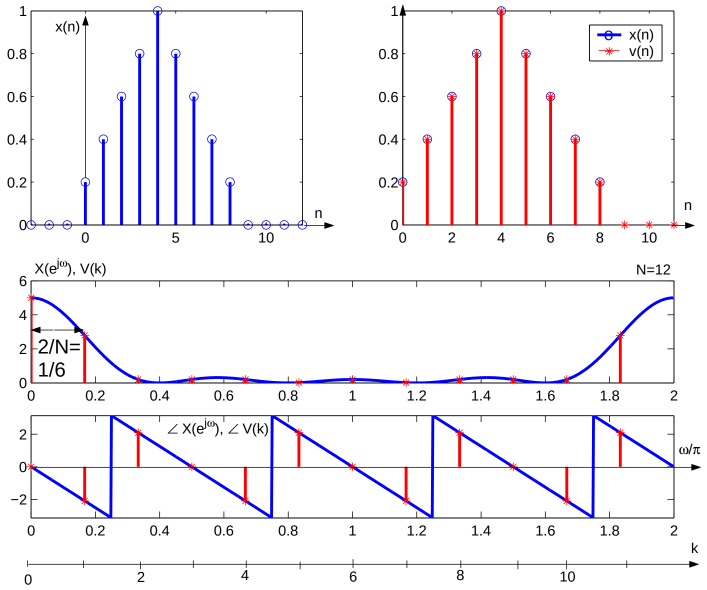


Figura 4: Exemplo: superposição periódica, $N = 12$

Exemplo: superposição periódica, $N = 7$

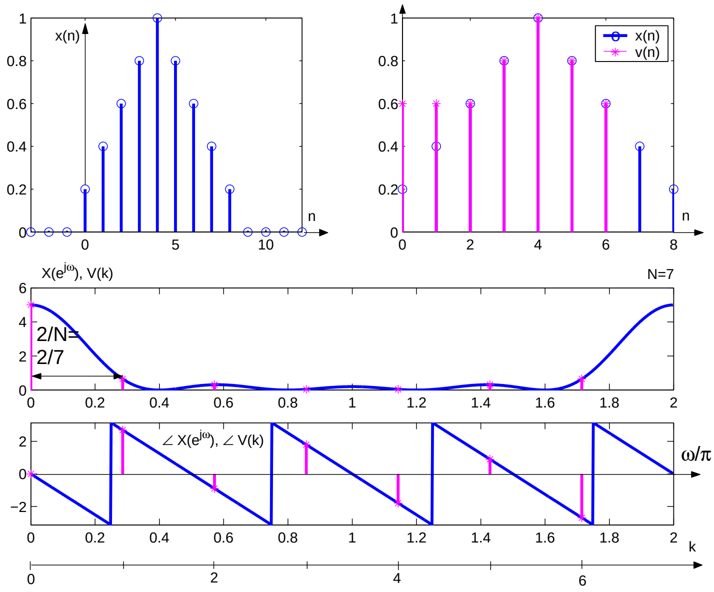


Figura 5: Exemplo: superposição periódica, $N = 7$

Algumas propriedades da TFD

Deslocamento circular em um intervalo N

O deslocamento no tempo de sequências não-nulas para $0 \leq n \leq N-1$ pode ser obtido da relação entre TFD e SFD. Lembre que uma sequência periódica $\tilde{x}(n)$ de período N pode ser convenientemente representada em termos de uma sequência qualquer $x(n)$ com comprimento N conforme ilustrado na Figura 6.

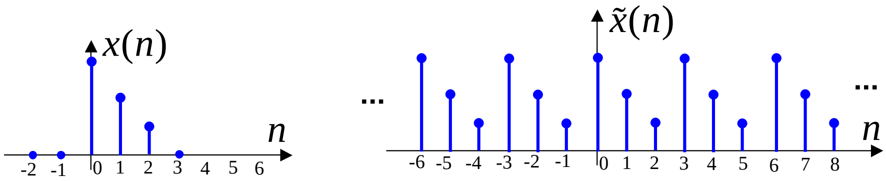
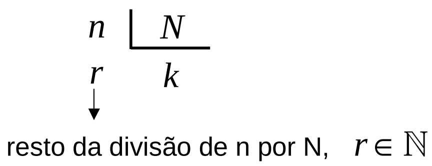


Figura 6: Repetição periódica de $x(n)$ com $N = 3$.

Nesse contexto, a operação de deslocamento no contexto da TFD é facilitado com o uso da operação $n \bmod N$, sendo $n \bmod N = \lfloor n \rfloor_N$ igual ao resto inteiro da divisão de n por N:



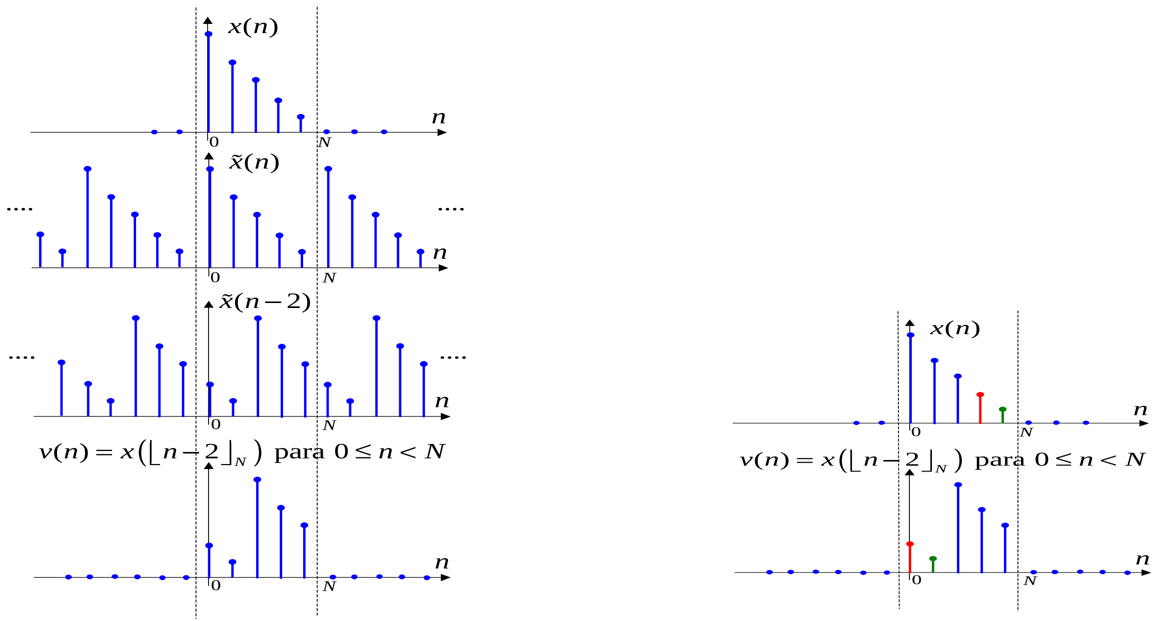


Figura 7: Deslocamento circular de $x(n)$

Seja uma sequência $x(n)$ de comprimento N e sua sequência periódica deslocada de m amostras $\tilde{v}(n) = \tilde{x}(n-m)$. Usando a propriedade do deslocamento no tempo da SFD temos

$$\tilde{v}(n) = \tilde{x}(n-m) \overset{\text{SFD}}{\leftrightarrow} \tilde{V}(k) = e^{-j\frac{2\pi}{N}km}\tilde{X}(k)$$

A correspondente sequência finita é

$$v(n) = x(\lfloor n-m \rfloor_N), \text{ para } 0 \leq n \leq N-1$$

Usando a propriedade do deslocamento no tempo da SFD temos

$$v(n) = x(\lfloor n-m \rfloor_N) \overset{\text{TFD}}{\leftrightarrow} V(k) = e^{-j\frac{2\pi}{N}km}X(k)$$

A convolução circular

Sejam $x_1(n)$ e $x_2(n)$ duas sequências de comprimento N com TFDs $X_1(k)$ e $X_2(k)$. É possível mostrar que

$$\boxed{\sum_{m=0}^{N-1} x_1(m)x_2(\lfloor n-m \rfloor_N) \overset{\text{TFD}}{\leftrightarrow} X_1(k)X_2(k)}$$

Substituição do deslocamento no tempo por um deslocamento circular $\rightarrow$ convolução circular

Exemplo: convolução linear e convolução circular

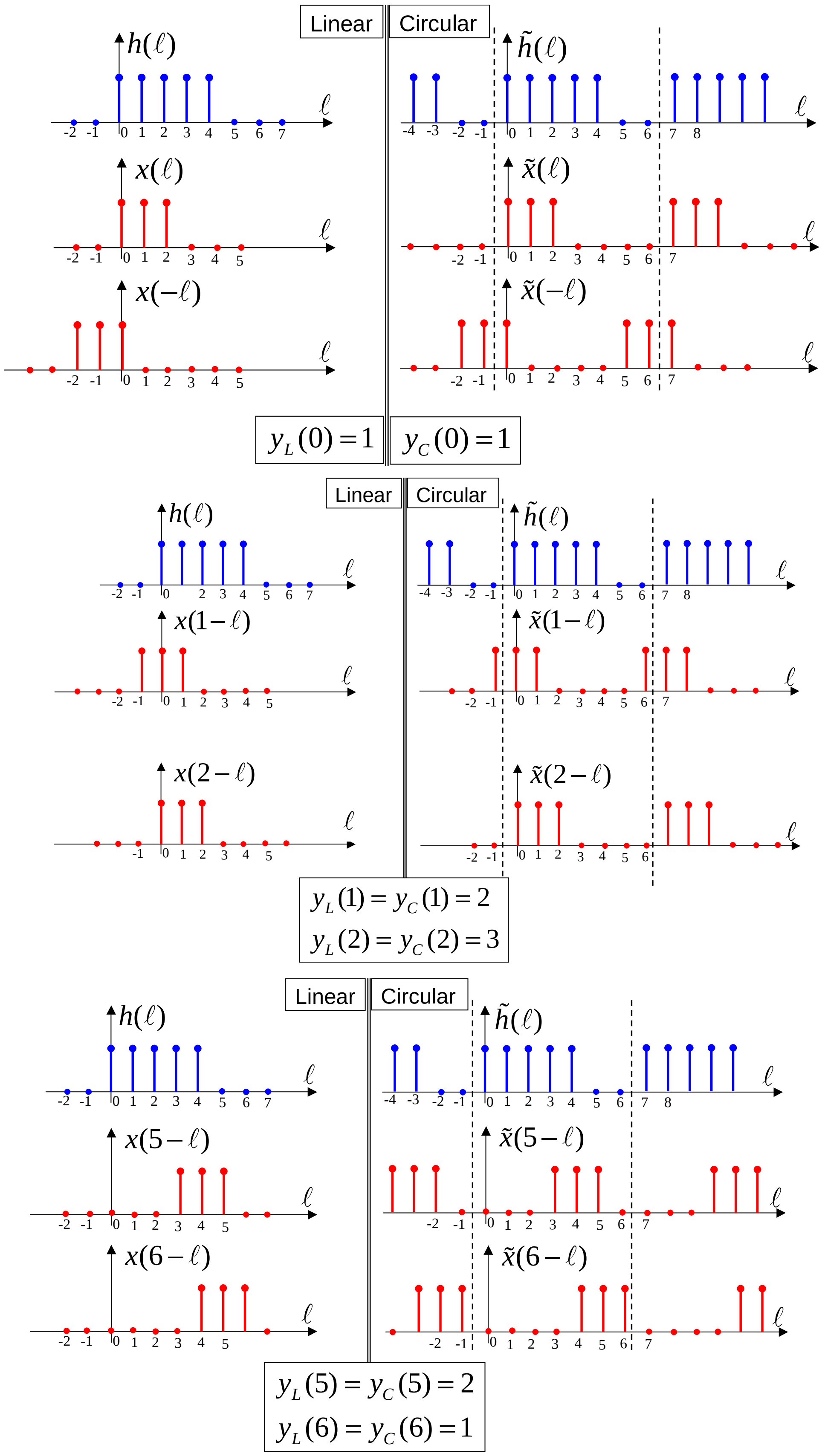


Figura 8: Convolução linear e convolução circular

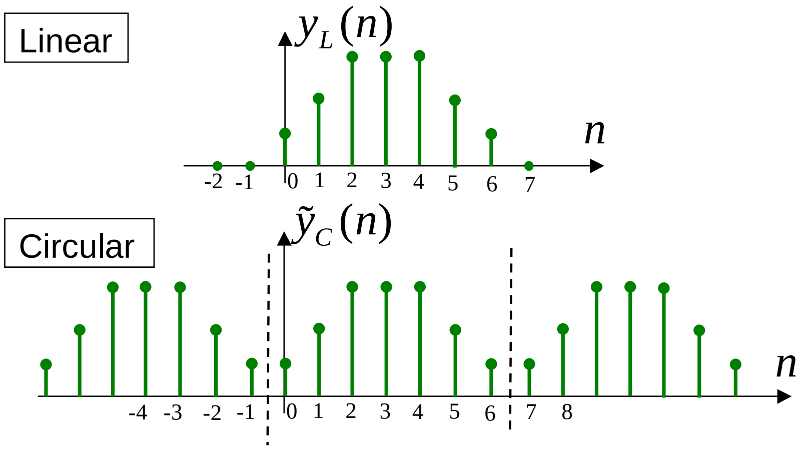


Figura 9: Resumo do resultado das convoluções.

Uma outra forma de calcular a convolução circular

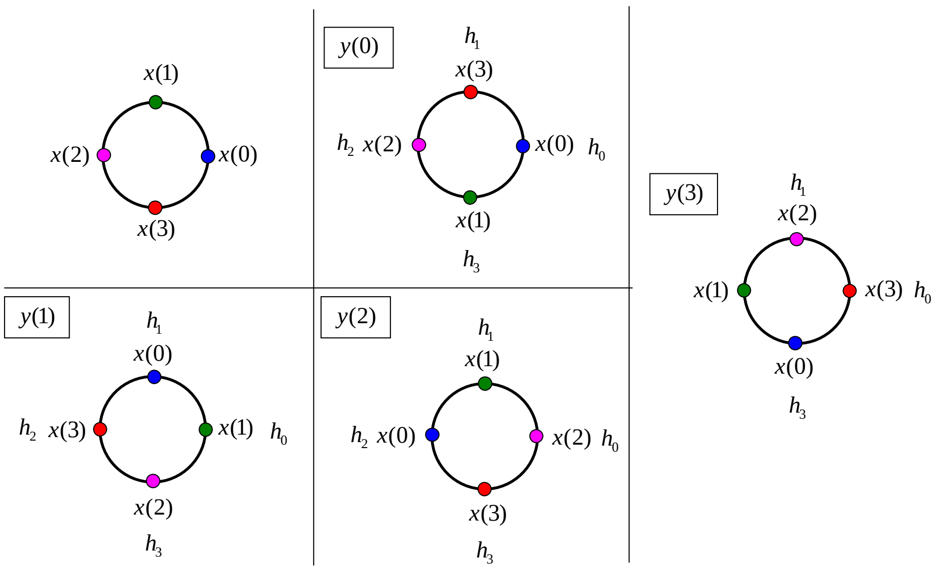


Figura 10: Forma pictórica de obter a convolução circular

Resumo de algumas propriedades da TFD

Propriedade	sequência no tempo	TFD
Deslocamento	$x(\lfloor n-m \rfloor_N)$	$e^{-j\frac{2\pi}{N}km}X(k)$
Modulação	$e^{j\frac{2\pi}{N}\lambda n}\tilde{x}(n)$	$X(\lfloor k-\lambda \rfloor_N)$
Dualidade	$X(n)$ $X(\lfloor -n \rfloor_N)$	$Nx(\lfloor -k \rfloor_N)$ $Nx(k)$
Multiplicação no tempo	$x_1(n)x_2(n)$	$\frac{1}{N}\sum_{\lambda=0}^{N-1} X_1(\lambda)X_2(\lfloor k-\lambda \rfloor_N)$
Convolução circular no tempo	$\sum_{\lambda=0}^{N-1} x_1(\lambda)x_2(\lfloor n-\lambda \rfloor_N)$	$X_1(k)X_2(k)$
Igualdade de Parseval	$\sum_{n=0}^{N-1} \lvert x(n)\rvert^2 = \frac{1}{N}\sum_{k=0}^{N-1} \lvert X(k)\rvert^2$	

2.11 Aplicações com fft

Convolução linear com fft

Sejam $x(n)$ e $h(n)$ duas sequências de comprimento N e M respectivamente. Completando essas sequências com zeros até $N + M - 1$, ou seja,

$$x_e(n) = \begin{cases} x(n), & 0 \leq\ n \leq N-1 \\ 0, & N \leq n \leq N+M-1 \end{cases} \qquad h_e(n) = \begin{cases} h(n), & 0 \leq n \leq M-1 \\ 0, & M \leq n \leq N+M-1 \end{cases}$$

então,

$$y_L(n) = x(n) * h(n) = y_C(n) = x_e(n) \circledcirc h_e(n)$$

em que $y_L(n)$ e $y_C(n)$ são os resultados da convolução linear e da convolução circular respectivamente.

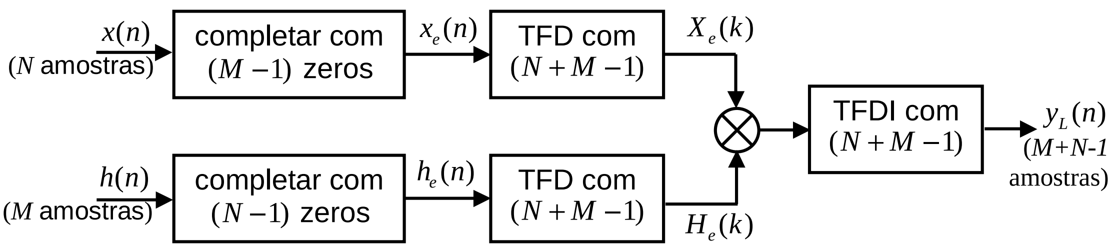


Figura 11: Representação em blocos para realizar a convolução linear com a TFD

Filtragem de sequências longas com a TFD

Há aplicações em que é necessário calcular a convolução linear de uma sequência de comprimento finito com outra de comprimento infinito ou de comprimento muito maior do que o da primeira sequência. Neste caso há dois métodos

- Método da sobreposição e soma
- Método da sobreposição e armazenamento

Método da sobreposição e soma

A sequência $x(n)$ pode ser decomposta em blocos de comprimento N:

$$x(n) = \sum_{m=0}^{\infty} x_m(n - mN)$$

sendo

$$x_m(n) = \begin{cases} x(n+mN), & \text{para } \ 0 \leq n \leq N-1 \\ 0, & \text{caso contrário} \end{cases}$$

A convolução linear de $x(n)$ com $h(n)$ de comprimento M pode ser escrita como

$$\begin{aligned} y(n) &= \sum_{\ell=0}^{M-1} h(\ell) \left(\sum_{m=0}^{\infty} x_m(n - mN - \ell) \right) \\ &= \sum_{m=0}^{\infty} \left(\sum_{\ell=0}^{M-1} h(\ell) x_m(n - mN - \ell) \right) = \sum_{m=0}^{\infty} y_m(n - mN), \end{aligned}$$

em que $y_m(n) = h(n) * x_m(n)$, tendo comprimento $N + M - 1$. Assim, $y_m(n)$ pode ser obtido usando a TFD e a TFDI.

De $y(n) = \sum_{m=0}^{\infty} y_m(n - mN)$ é possível observar que

- $h(n) * x_0(n)$ tem comprimento $N + M - 1$ e é definido para $0 \leq n \leq N + M - 2$
- $h(n) * x_1(n)$ também tem comprimento $N + M - 1$, sendo definido no intervalo $N \leq n \leq 2N + M - 2$

Há uma sobreposição de $M - 1$ amostras entre duas convoluções adjacentes!

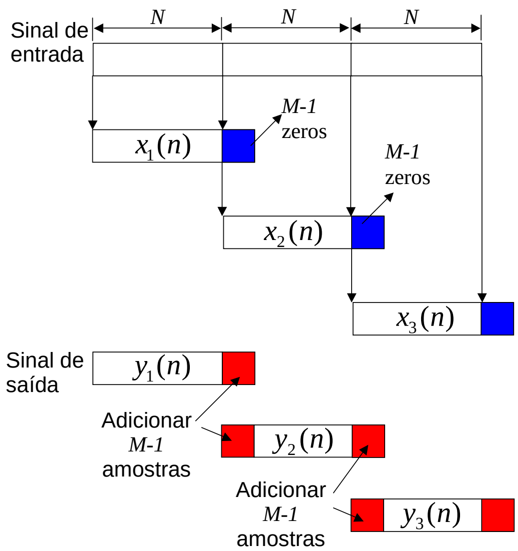


Figura 12: Método da sobreposição e soma.

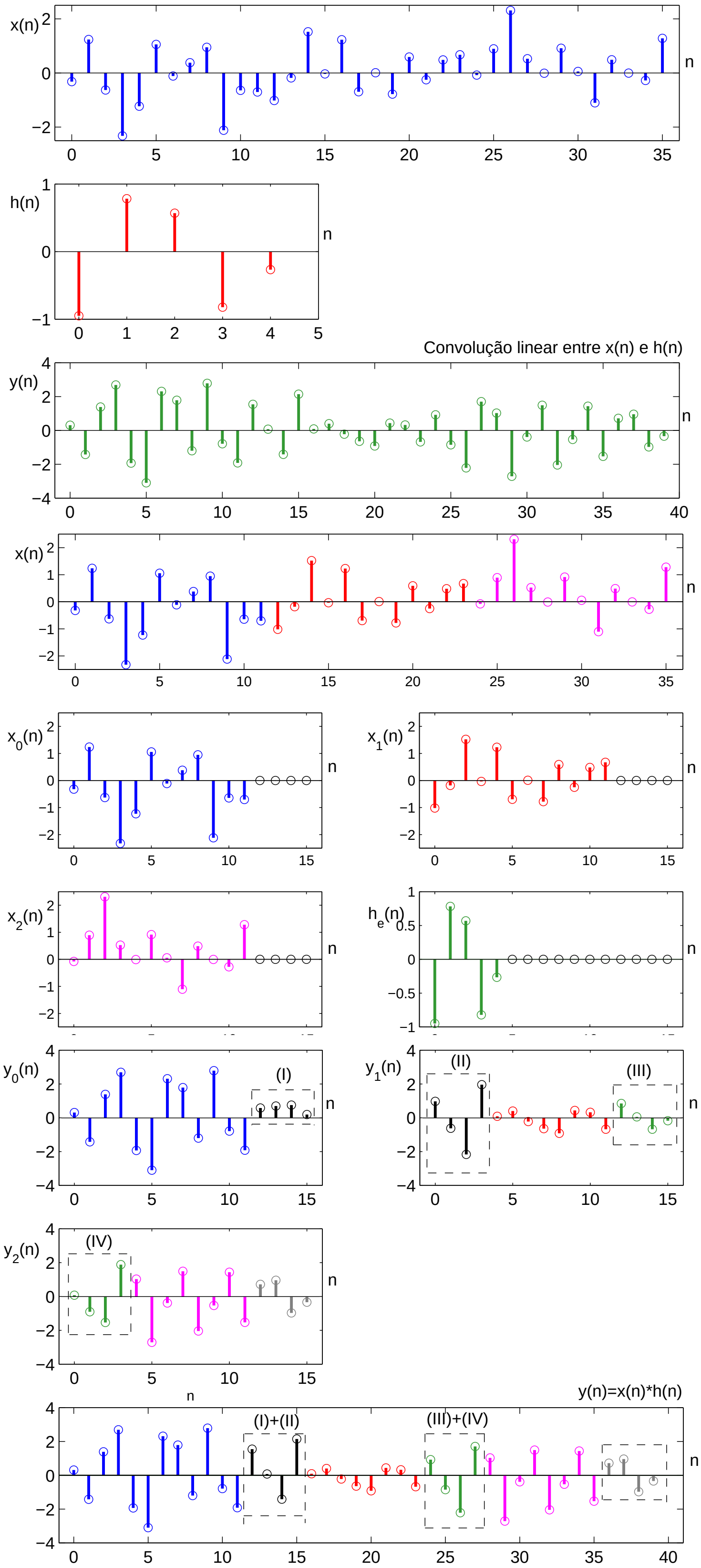


Figura 13: Método da sobreposição e soma.

Método da sobreposição e armazenamento

Sejam $x(n)$ uma sequência de comprimento 4 e $h(n)$ uma sequência de comprimento 3. As seis amostras da convolução linear $y_L(n) = x(n) * h(n)$ são iguais a

$$
\begin{aligned}
y_L(0) &= h(0)x(0) \\
y_L(1) &= h(0)x(1) + h(1)x(0) \\
y_L(2) &= h(0)x(2) + h(1)x(1) + h(2)x(0) \\
y_L(3) &= h(0)x(3) + h(1)x(2) + h(2)x(1) \\
y_L(4) &= h(1)x(3) + h(2)x(2) \\
y_L(5) &= h(2)x(3)
\end{aligned}
$$

Completando $h(n)$ com 1 zero, as 4 amostras da convolução circular $y_C(n) = h_e(n) \circledcirc x(n)$ são

$$
\begin{aligned}
y_C(0) &= h(0)x(0) + h(1)x(3) + h(2)x(2) \neq y_L(0) \\
y_C(1) &= h(0)x(1) + h(1)x(0) + h(2)x(3) \neq y_L(1) \\
y_C(2) &= h(0)x(2) + h(1)x(1) + h(2)x(0) = y_L(2) \\
y_C(3) &= h(0)x(3) + h(1)x(2) + h(2)x(1) = y_L(3)
\end{aligned}
$$

Se $x(n)$ tem comprimento N e $h(n)$ tem comprimento M, com $N > M$, então as $M-1$ amostras da convolução circular de comprimento N não coincidem com as da convolução linear e devem ser descartadas!

Considere $h(n)$ de comprimento M e $x(n)$ longa, decomposta em blocos de sequências de comprimento N:

$$x_m(n) = x(n + m(N - M + 1)),$$

com $0 \leq n \leq N-1, \;\; 0 \leq m \leq \infty$ e as amostras da convolução linear podem ser calculadas como

$$y_L(n + m(N - M + 1)) = y_m(n), \quad M - 1 \leq n \leq N - 1,$$

sendo

$$y_m(n) = h(n) \circledcirc x_m(n), \;\; \text{para} \;\; M - 1 \leq n \leq N - 1$$

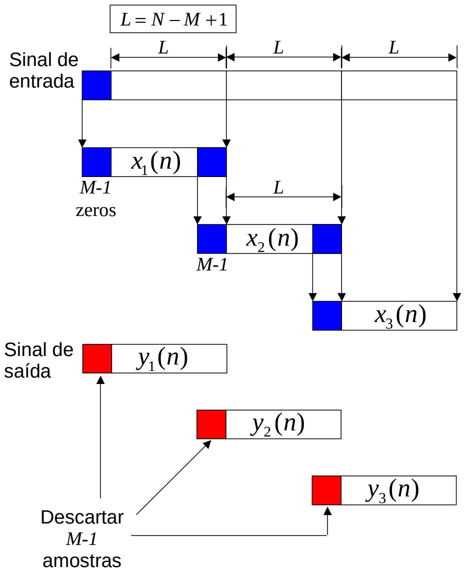


Figura 14: Método da sobreposição e armazenamento.

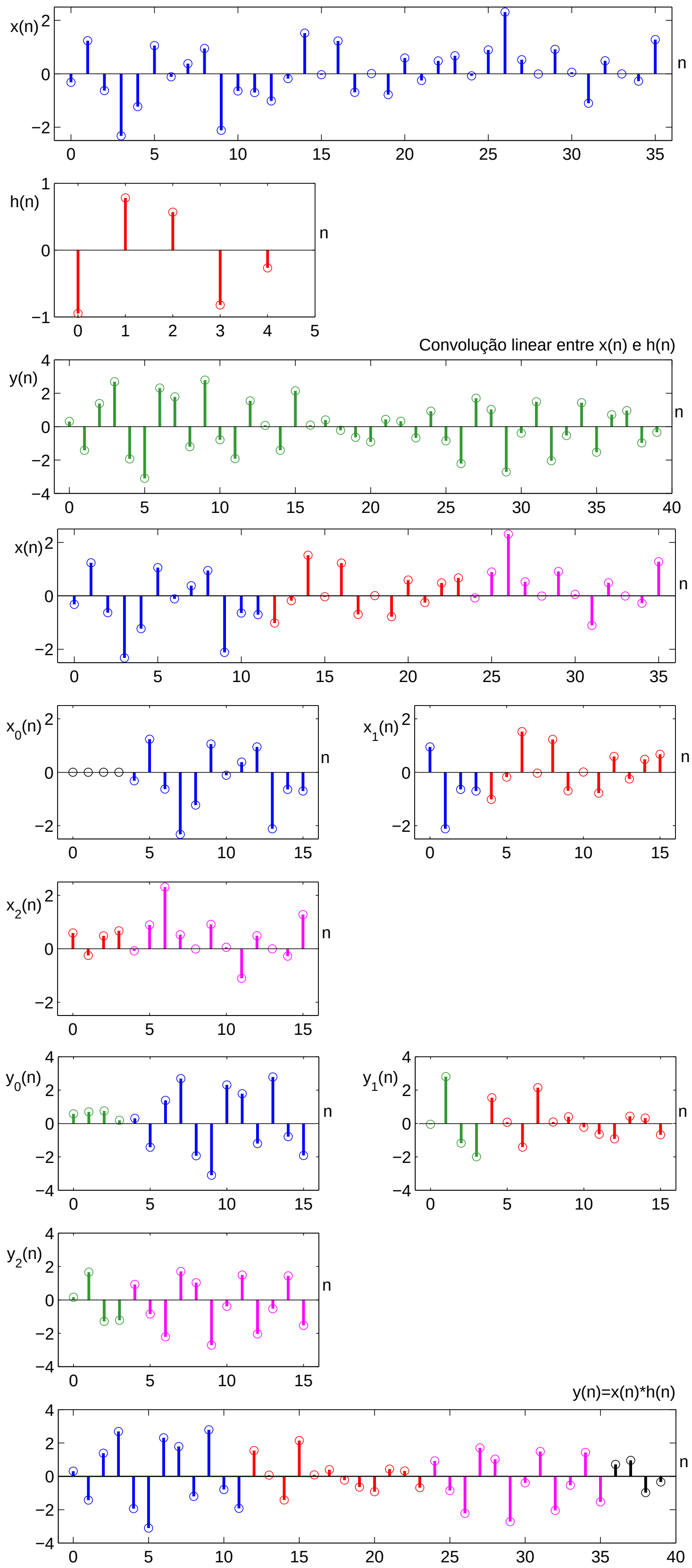


Figura 15: Método da sobreposição e armazenamento.

Análise de sinais periódicos com fft

Análise de sinais periódicos sem ruído

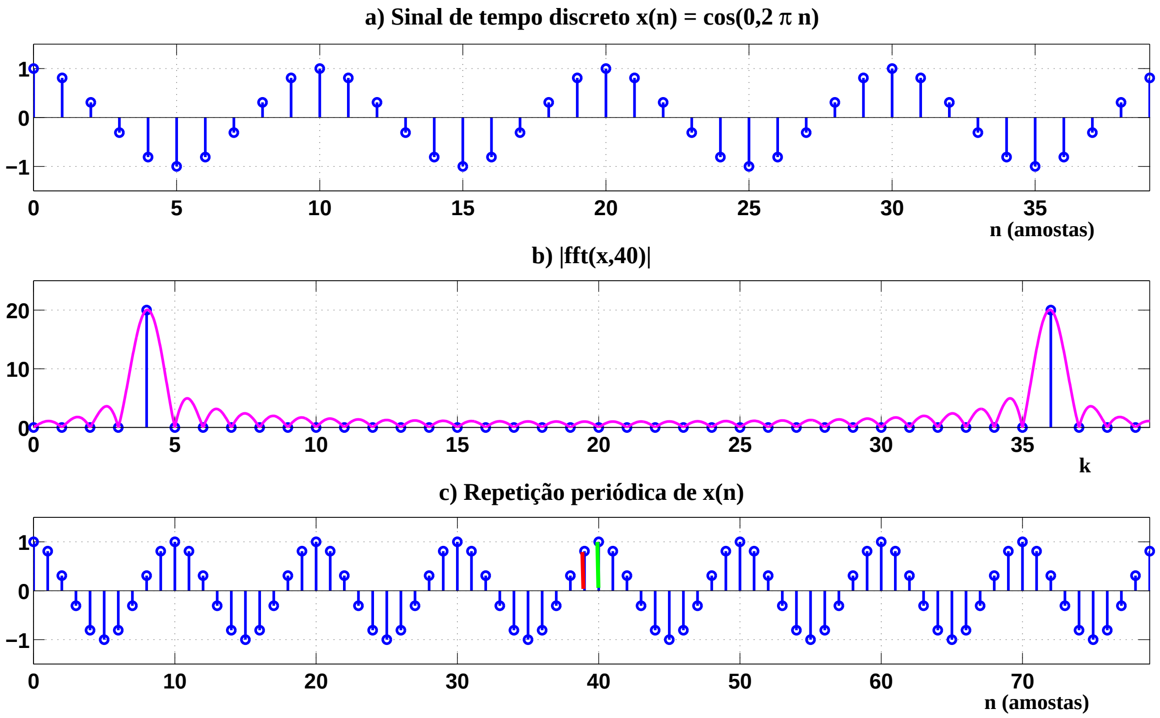


Figura 16: (a) Sinal $\cos(0,2\pi n)$ janelado com uma janela retangular de duração de 40 amostras; (b) módulo da fft de $x(n)$ com $N = 40$ pontos; (c) repetição periódica do sinal janelado.

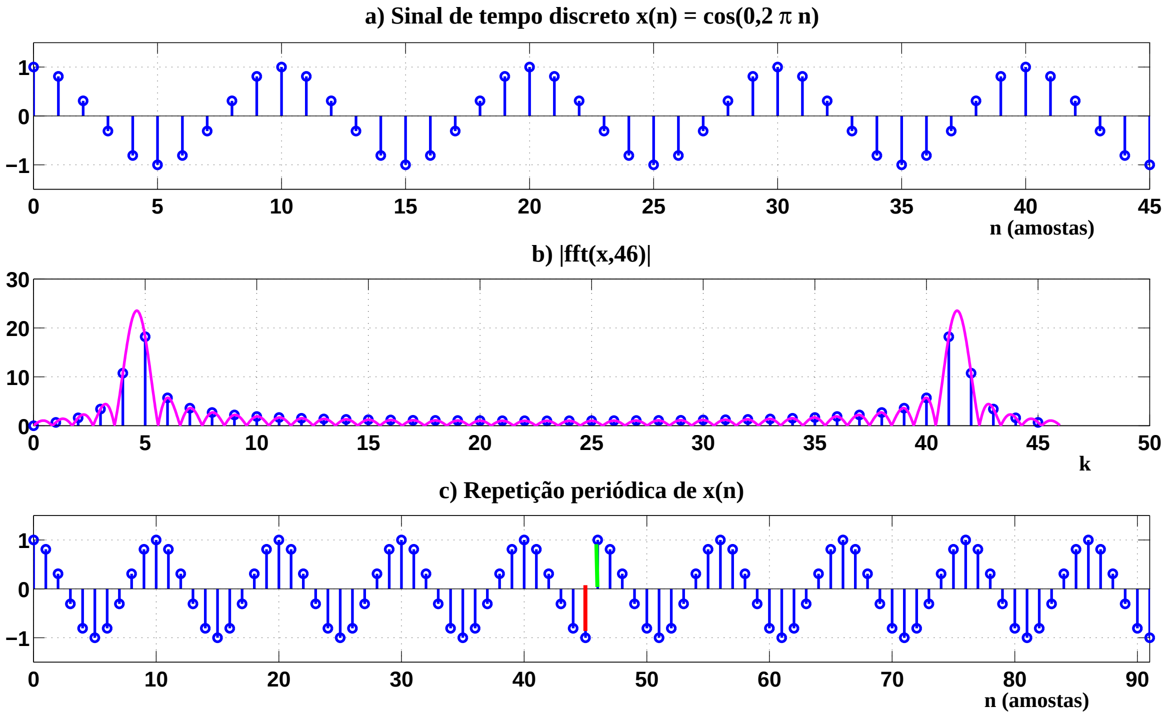


Figura 17: (a) Sinal $\cos(0, 2\pi n)$ janelado com uma janela retangular de duração de 46 amostras; (b) módulo da fft de $x(n)$ com $N = 46$ pontos; (c) repetição periódica do sinal janelado.

Quando a janela usada para truncar o sinal não possui um comprimento igual a um múltiplo inteiro do período do sinal ocorre o vazamento (*leakage*). O vazamento tem duas consequências:

1. efeito cerca (*picket fence effect*): a raia principal não aparece e é cercada por raias próximas a raia principal.

2. vazamento (*leakage effect*): aparecimento de diversas raias de amplitudes relativamente baixas devido as amostras tomadas nos lóbulos laterais da TFTD da janela deslocada.

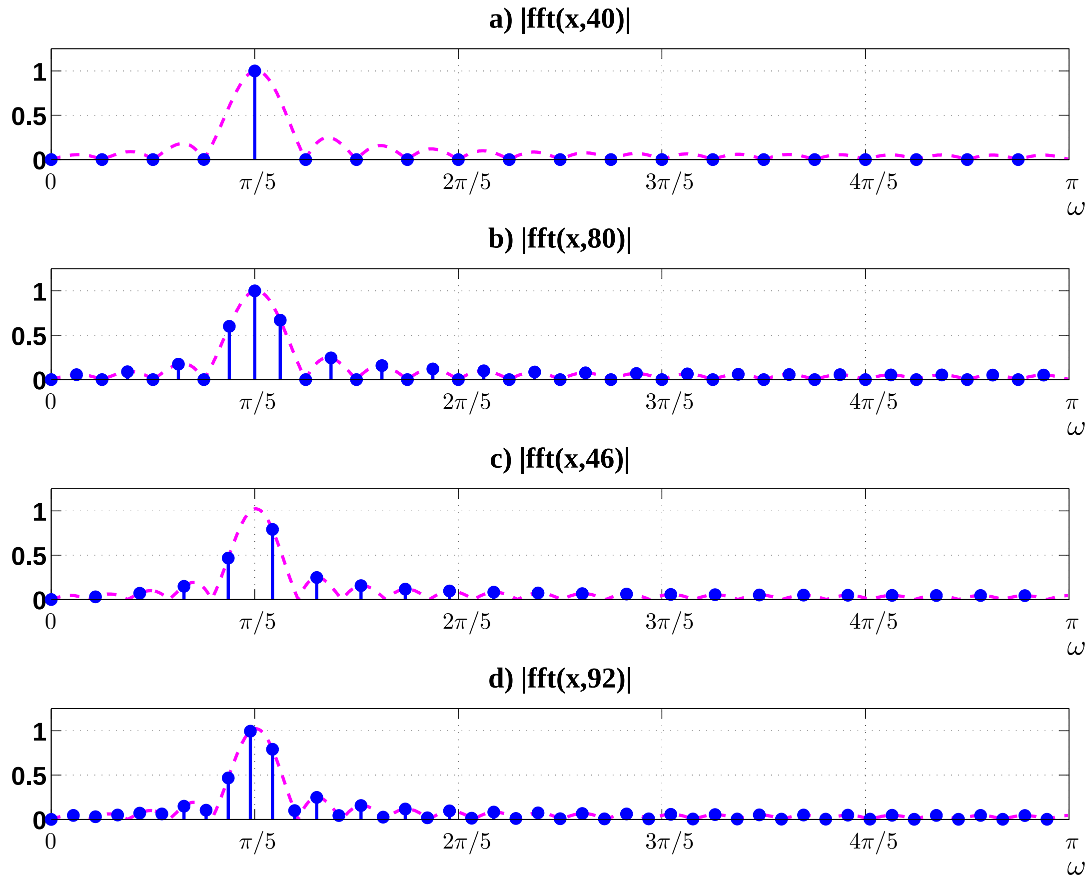


Figura 18: Módulo da fft de $\cos(0,2\pi n)$: (a) Comprimento da janela e número de pontos da fft 40 amostras; (b) Comprimento da janela 40 amostras e número de pontos da fft 80 amostras; (c) Comprimento da janela e número de pontos da fft 46 amostras; (d) Comprimento da janela 46 amostras e número de pontos da fft 92 amostras.

Análise de sinais periódicos com ruído

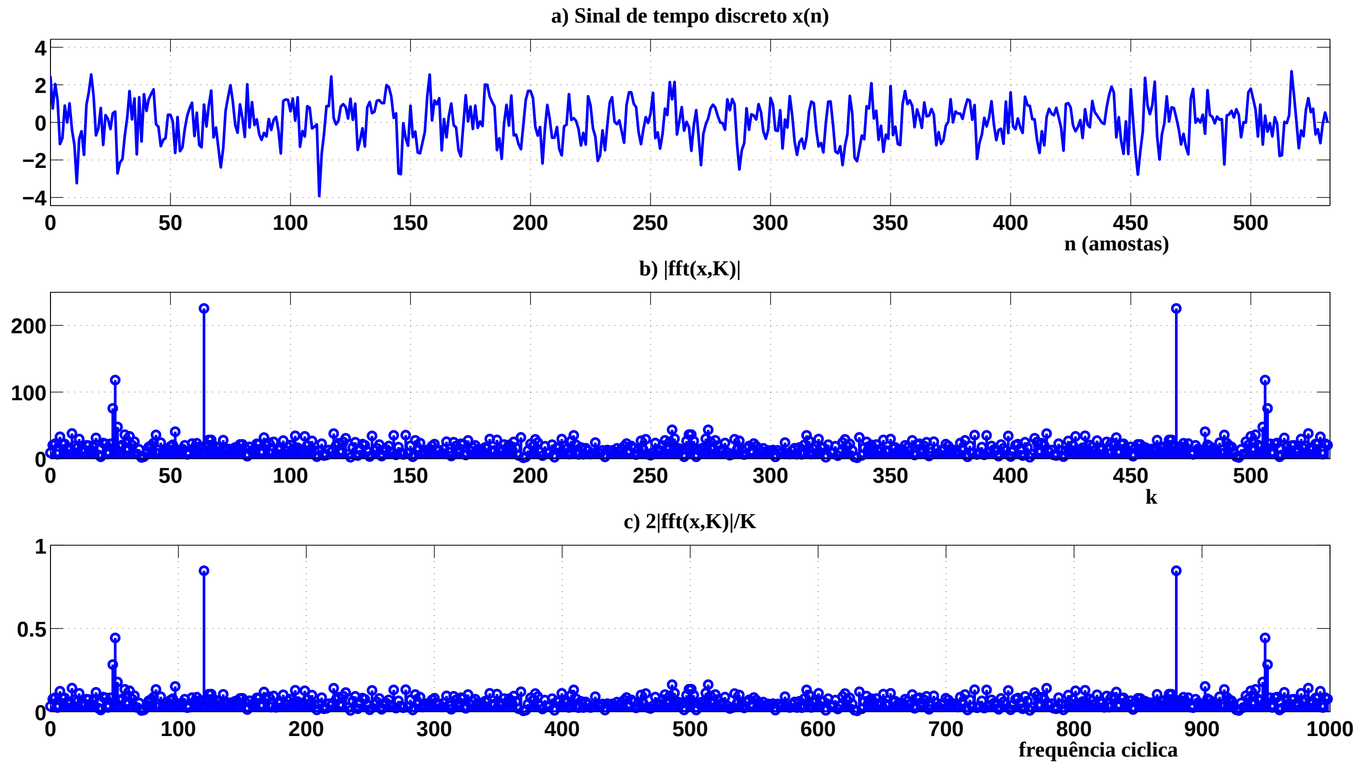


Figura 19: Exemplo de FFt de um sinal periódico com ruído.

```
clear all;
Nx=600; % duração total do sinal
J=533; % Duração da janela
K=J; % no. de pontos da TFD (no. de pontos da fft)
% %%%%%%%%%%%%%%%sinal a ser estimado
A1=0.6;
A2=0.8;
f1=50; %122;
f2=120;
fs=1000;
omega1=2*pi*f1/fs;
omega2=2*pi*f2/fs;
n=0:Nx-1;
x=A1*cos(omega1*n)+A2*cos(omega2*n)+0.8*randn(1,Nx); % sinal
figure(1);
```

```
subplot(311); plot(n(1:J),x(1:J),'Linewidth',2); grid
title('a) Sinal de tempo discreto x(n)','FontSize',16,'FontName','Times New Roman','FontWeight','bold'
xlabel('n (amostas)','FontSize',16,'FontName','Times New Roman','FontWeight','bold')
axis([0 J -max(abs(x))-.5 max(abs(x))+.5])
set(gca,'FontSize',16,'FontWeight','bold')
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
X=fft(x(1:J),K);
Kk=round(length(X));
k=[0:Kk-1];
%
subplot(312); stem(k,abs(X(1:Kk)),'Linewidth',2);
grid
title('b) |fft(x,K)|','FontSize',16,'FontName','Times New Roman','FontWeight','bold')
xlabel('k','FontSize',16,'FontName','Times New Roman','FontWeight','bold')
set(gca,'FontSize',16,'FontWeight','bold')
subplot(313); stem(k*fs/J,2*abs(X(1:Kk)/J),'Linewidth',2);
grid
title('c) 2|fft(x,K)|/K','FontSize',16,'FontName','Times New Roman','FontWeight','bold')
xlabel('frequência cíclica','FontSize',16,'FontName','Times New Roman','FontWeight','bold')
set(gca,'FontSize',16,'FontWeight','bold')
```

Referências

- A. V. Oppenheim, R. W. Shafer, J.R. Buck, *Discrete-time signal processing.* 2a. ou 3a. edição, Prentice Hall.
- S. K. Mitra, *Digital signal processing - a computer based approach.* 3a. edição, McGraw-Hill, 2006.
- J. G. Proakis, D. G. Manolakis, *Digital signal processing : principles, algorithms, and applications*, 4a. edição, Prentice Hall, 2006.
- M. Gerken, P. M. S. Burt, *Processamento Digital de Sinais.* Publicação interna da EPUSP, junho de 2000.

- As Figuras: 4, 5, 3, 7, 8, 9, 10, 11, 12, 13, 14, 15 foram gentilmente cedidas pelo Prof. Magno T. Madeira da Silva em 2008.