UNIVERSIDADE FEDERAL RURAL DO SEMI-ÁRIDO (UFERSA)
CENTRO MULTIDISCIPLINAR DE PAU DOS FERROS (CMPF)
DEPARTAMENTO DE ENGENHARIAS E TECNOLOGIA (DETEC)

PROCESSAMENTO DIGITAL DE SINAIS

Transformada Discreta de Fourier

Prof.: Pedro Thiago Valério de Souza
UFERSA – Campus Pau dos Ferros
pedro.souza@ufersa.edu.br

Introdução

- Importância da transformada de Fourier (DTFT) e da transformada Z:
  - Convolução;
  - Análise de sistemas de tempo discreto.
- Problema: Implementação computacional da transformada de Fourier:

$$X\left(e^{j\omega}\right) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}$$

ω → variável contínua.

Implementação computacional → requer que seja calculado em um conjunto discreto de frequências.

- Solução: Transformada Discreta de Fourier (DFT):

DFT = Amostragem da DTFT: $X\left(e^{j\omega}\right) \xrightarrow{\textit{Amostragem}} X[k]$

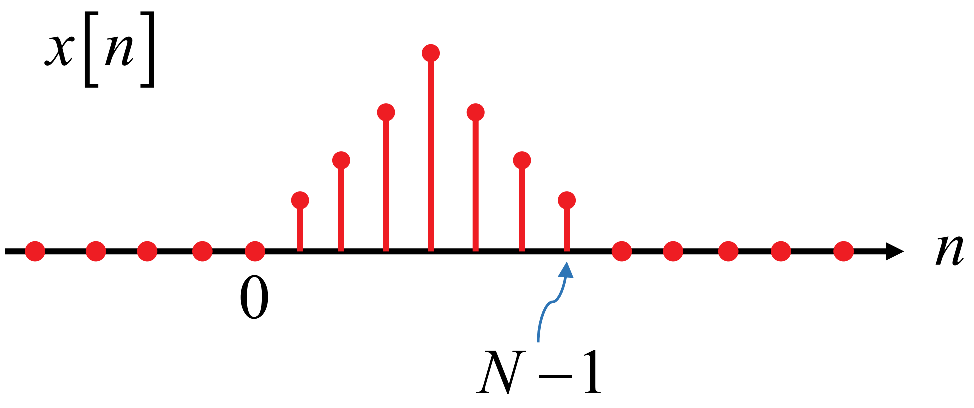
Restrição: sequências de tamanho finito.

Introdução

- Amostragem no tempo → Periódico na frequência;
- Amostragem na frequência → Periódico no tempo;
- Análise de sinais periódicos no tempo → Série Discreta de Fourier (DFS);
- Analogia da DFT com a série discreta de Fourier (DFS).

Introdução

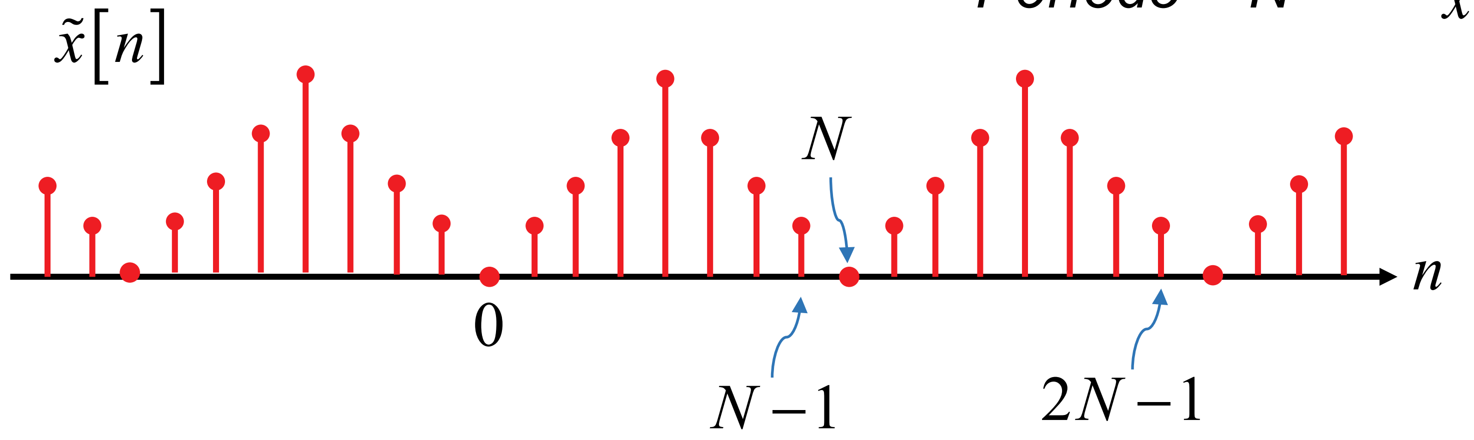
- Considerando $x[n]$ de tamanho finito.



$$x[n] = 0 \quad n > N-1$$
$$n < 0$$

$\tilde{x}[n]$ Periódico → pode ser construído a partir da repetição de *x*[*n*]

Período = N



$$\tilde{x}[n] = x[n] + x[n-N] + x[n-2N] + \cdots$$
$$+ x[n+N] + x[n+2N] + \cdots$$

$$\tilde{x}[n] = \sum_{r=-\infty}^{\infty} x[n+rN]$$

Introdução

$\tilde{x}[n]$ Periódico → pode ser construído a partir da repetição de *x*[*n*]

$$x[n] = x[n \text{ modulo } N] = x\left[((n))_N\right]$$

n modulo *N* → Resto da divisão de *n* por *N*.

Se $n = n_1 + n_2 N$ → n modulo $N = n_1$

Exemplo: $\tilde{x}[n] = x\left[((n))_N\right]$

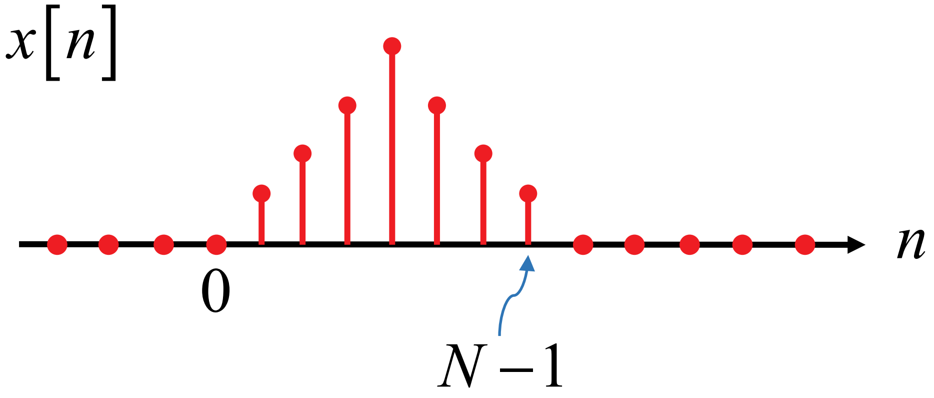
$$\tilde{x}[0] = x\left[((0))_N\right] = x[0]$$

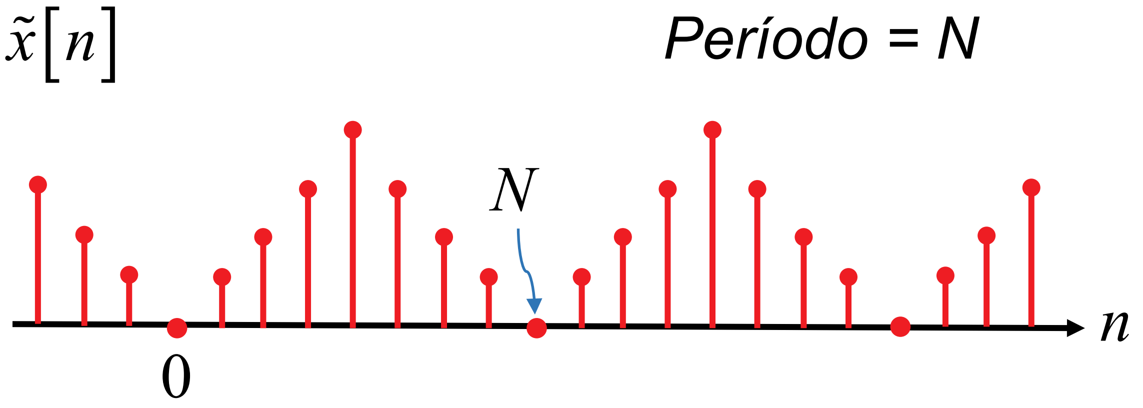
$$\tilde{x}[1] = x\left[((1))_N\right] = x[1]$$

$$\tilde{x}[2] = x\left[((2))_N\right] = x[2]$$

$$\tilde{x}[N] = x\left[((N))_N\right] = x[0]$$

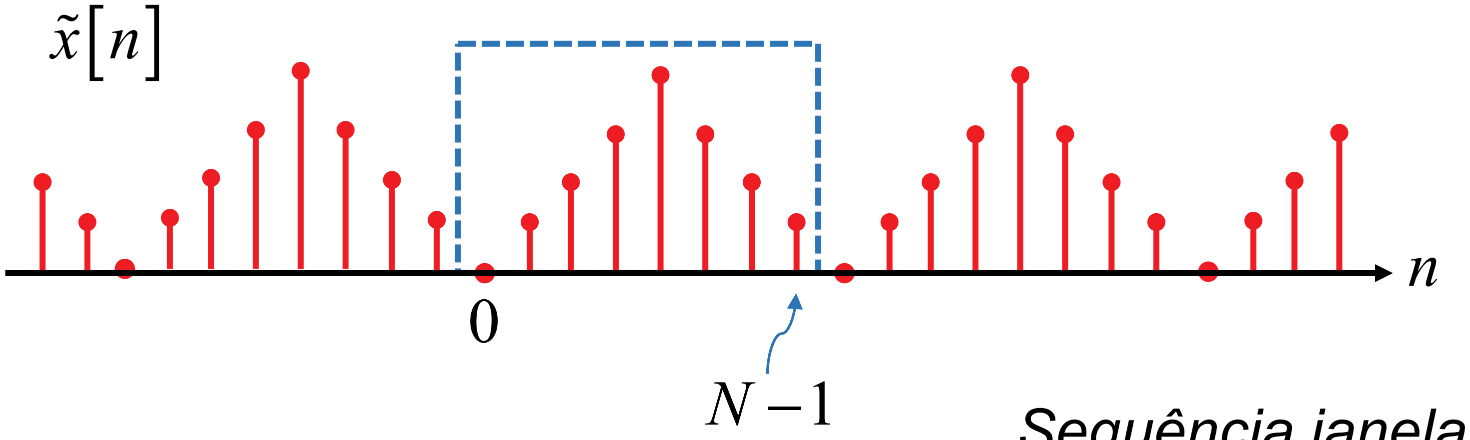
$$\tilde{x}[N+1] = x\left[((N+1))_N\right] = x[1]$$





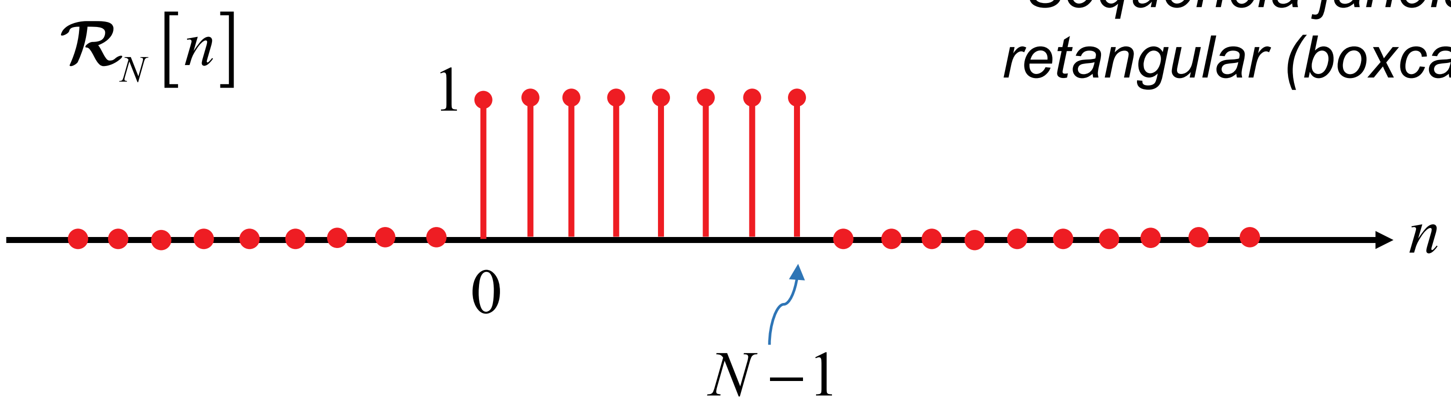
Introdução

- Relação de $\tilde{x}[n]$ com $x[n]$:



$$x[n] = \begin{cases} \tilde{x}[n] & 0 \le n \le N-1 \\ 0 & \text{c.c.} \end{cases}$$

Sequência janela retangular (boxcar)



$$\mathcal{R}_N[n] = \begin{cases} 1 & 0 \le n \le N-1 \\ 0 & \text{c.c.} \end{cases}$$

Desta forma: $x[n] = \tilde{x}[n]\mathcal{R}_N[n]$

Introdução

- Por ser periódico, $\tilde{x}[n]$ tem uma representação em série discreta de Fourier (DFS);
- Existe uma forte relação entre sequências de tamanho finito e sequências periódicas;
- Ideia: Usar a série discreta de Fourier (DFS) para representar sequências periódicas obtidas de sequências de tamanho finito.

$\tilde{x}[n]$ tem uma representação em série discreta de Fourier (DFS).

$$\text{DFS de } \tilde{x}[n] = \text{DFT de } x[n]$$

$$\tilde{x}[n] = \sum_{r=-\infty}^{\infty} x[n+rN] = x\left[((n))_N\right]$$

Série Discreta de Fourier

- Sendo $\tilde{x}[n]$ periódico com período *N*, então $\tilde{x}[n]$ pode ser escrito como uma combinação linear de exponenciais complexas:

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

$$\theta_k[n] = e^{j\frac{2\pi}{N}nk} \longrightarrow$$ Periódico em *k* com período *N*

$$\theta_{k+N}[n] = e^{j\frac{2\pi}{N}n(k+N)} = e^{j\frac{2\pi}{N}nk} e^{j\frac{2\pi}{N}nN} = e^{j\frac{2\pi}{N}nk} \cancelto{1}{e^{j2\pi n}} = e^{j\frac{2\pi}{N}nk} = \theta_k[n]$$

$$\left.\begin{array}{l} \theta_0[n] = \theta_N[n] \\ \theta_1[n] = \theta_{N+1}[n] \\ \vdots \end{array}\right\}$$ Existem apenas *N* exponenciais distintas.

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

Série Discreta de Fourier

- Os coeficientes $\tilde{X}[k]$ podem ser calculados como (ver Oppenheim & Schafer, pg. 369):

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

- Em resumo:

$$\left.\begin{aligned} \tilde{x}[n] &= \frac{1}{N}\sum_{k=0}^{N-1} \tilde{X}[k] e^{j\frac{2\pi}{N}nk} \\ \tilde{X}[k] &= \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j\frac{2\pi}{N}nk} \end{aligned}\right\} \quad \tilde{x}[n] \overset{DFS}{\rightleftarrows} \tilde{X}[k]$$

- Notar que $\tilde{X}[k]$ é periódico em *k* com período *N*.

$$\tilde{X}[k+N] = \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j\frac{2\pi}{N}n(k+N)} = \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j\frac{2\pi}{N}nk} e^{-j\frac{2\pi}{N}nN} = \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j\frac{2\pi}{N}nk} \cancelto{1}{e^{-j2\pi n}} = \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j\frac{2\pi}{N}nk}$$

$$= \tilde{X}[k]$$

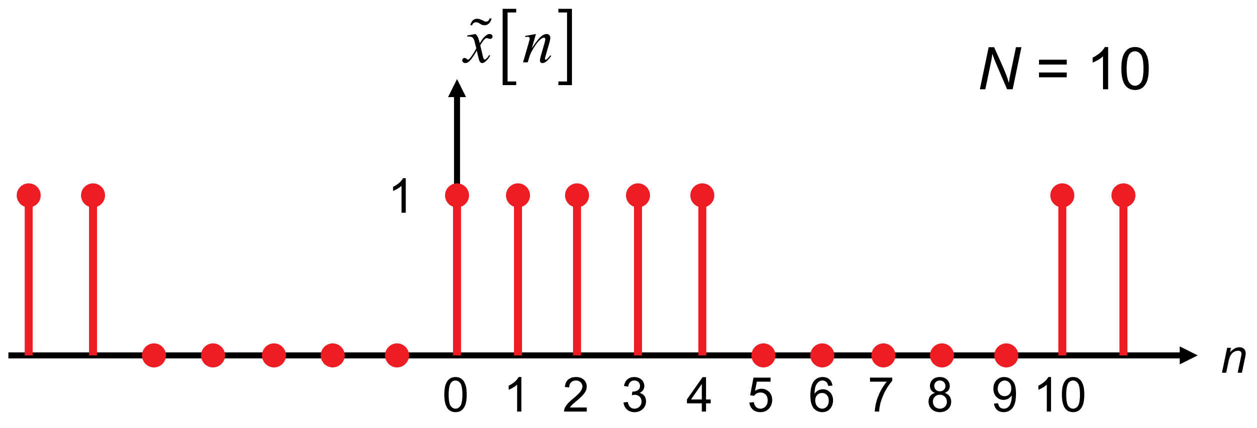
Série Discreta de Fourier

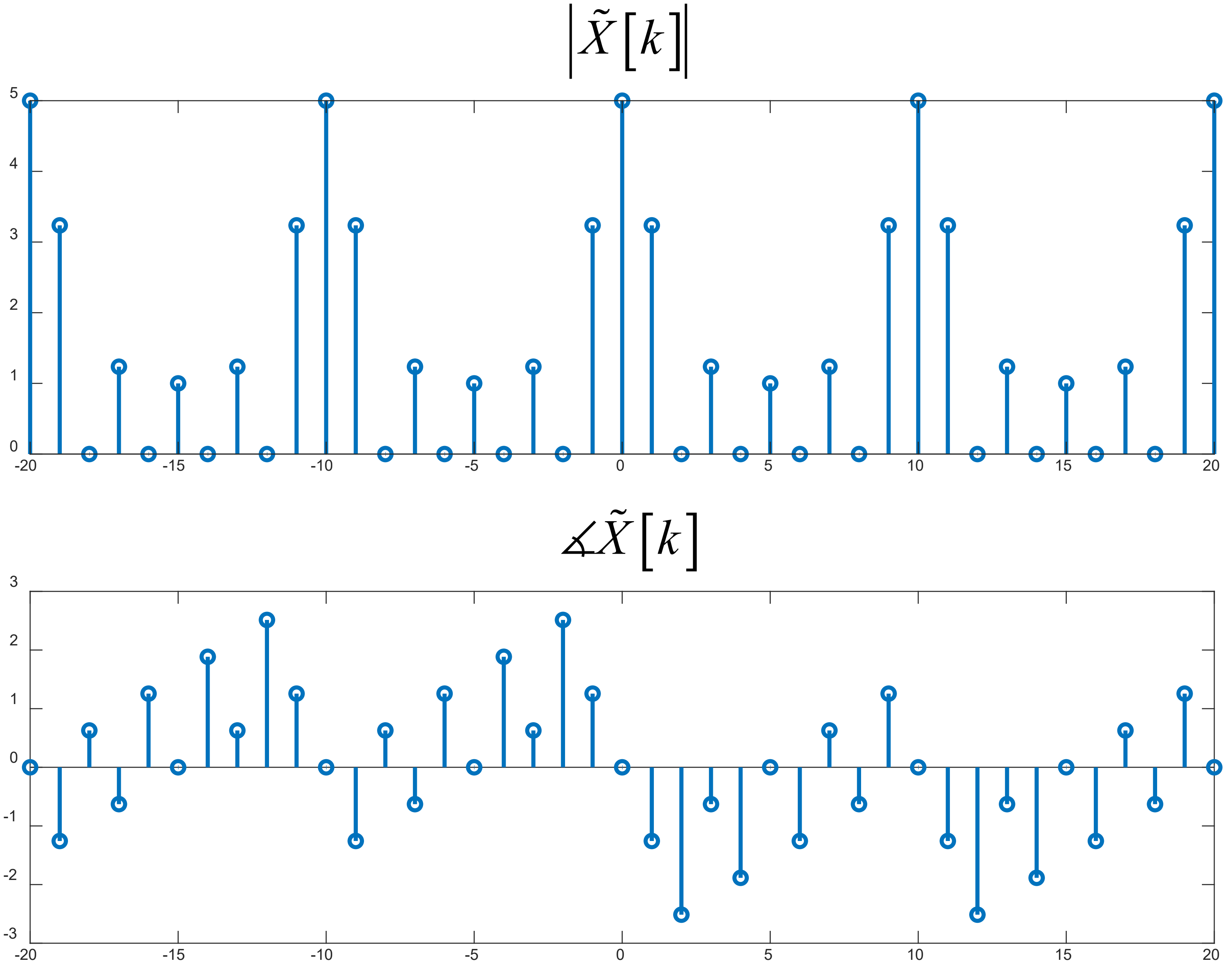
- Como $\tilde{x}[n]$ e $\tilde{X}[k]$ são periódicos com período *N*, tanto faz qual o intervalo de *n* ou de *k* que é calculada a DFS, desde que sejam *N* amostras seguidas.
- Notação simplificada:

$$W_N = e^{-j\frac{2\pi}{N}} \qquad \begin{cases} \tilde{x}[n] = \dfrac{1}{N}\displaystyle\sum_{k=0}^{N-1} \tilde{X}[k] W_N^{-nk} \\ \tilde{X}[k] = \displaystyle\sum_{n=0}^{N-1} \tilde{x}[n] W_N^{nk} \end{cases}$$

Série Discreta de Fourier

Exemplo 1: Determine a série discreta de Fourier para o sinal apresentado na Figura abaixo.




$\left|\tilde{X}[k]\right|$
5
4
3
2
1
0
-20
-15
-10
-5
0
5
10
15
20
$\measuredangle\tilde{X}[k]$
3
2
1
0
-1
-2
-3
-20
-15
-10
-5
0
5
10
15
20

Propriedades da DFS

- Linearidade:

$$\tilde{x}_1[n] \xrightleftharpoons{DFS} \tilde{X}_1[k] \text{ , com período } N$$

$$\tilde{x}_2[n] \xrightleftharpoons{DFS} \tilde{X}_2[k] \text{ , com período } N$$

$$a\tilde{x}_1[n] + b\tilde{x}_2[n] \xrightleftharpoons{DFS} a\tilde{X}_1[k] + b\tilde{X}_2[k] \text{ , com período } N$$

- Deslocamento:

$$\tilde{x}[n] \xrightleftharpoons{DFS} \tilde{X}[k]$$

$$\tilde{x}[n+m] \xrightleftharpoons{DFS} W_N^{-km}\tilde{X}[k]$$

$$W_N^{-\ell n}\tilde{x}[n] \xrightleftharpoons{DFS} \tilde{X}[n+\ell]$$

Propriedades da DFS

- Simetria: Sendo $\tilde{x}[n]$ real:

Propriedade 1: $\tilde{X}[k] = \tilde{X}^*[-k]$ (a DFS é conjugada simétrica)

Sendo: $\tilde{X}[k] = \tilde{X}_R[k] + j\tilde{X}_I[k]$

Propriedade 2: $\tilde{X}_R[k] = \tilde{X}_R[-k]$ (a parte real da DFS é simétrica par)

$$\tilde{X}_R[k-N] = \tilde{X}_R[-k+N]$$

$$\tilde{X}_R[k] = \tilde{X}_R[N-k]$$

Propriedade 3: $\tilde{X}_I[k] = -\tilde{X}_I[-k]$ (a parte imaginária da DFS é simétrica impar)

$$\tilde{X}_I[k-N] = -\tilde{X}_I[-k+N]$$

$$\tilde{X}_I[k] = -\tilde{X}_I[N-k]$$

Propriedades da DFS

- Simetria: Sendo $\tilde{x}[n]$ real:

Sendo: $\tilde{X}[k] = \left|\tilde{X}[k]\right| e^{j\measuredangle \tilde{X}[k]}$

<u>Propriedade 4:</u> $\left|\tilde{X}[k]\right| = \left|\tilde{X}[-k]\right|$ (a magnitude da DFS é simétrica par)

$$\left|\tilde{X}[k]\right| = \left|\tilde{X}[N-k]\right|$$

<u>Propriedade 5:</u> $\measuredangle \tilde{X}[k] = -\measuredangle \tilde{X}[-k]$ (a fase da DFS é simétrica impar)

$$\measuredangle \tilde{X}[k] = -\measuredangle \tilde{X}[N-k]$$

Sendo: $\tilde{x}[n] = \tilde{x}_e[n] + \tilde{x}_o[n]$ $\begin{cases} \tilde{x}_e[n] = \dfrac{1}{2}\left[\tilde{x}[n] + \tilde{x}[-n]\right] \\ \tilde{x}_o[n] = \dfrac{1}{2}\left[\tilde{x}[n] - \tilde{x}[-n]\right] \end{cases}$

Propriedades da DFS

- Simetria: Sendo $\tilde{x}[n]$ real:

  <u>Propriedade 6:</u> $\tilde{x}_e[n] \overset{DFS}{\longleftrightarrow} \tilde{X}_R[k]$ (a DFS da parte par é a parte real da DFS)

  <u>Propriedade 7:</u> $\tilde{x}_o[n] \overset{DFS}{\longleftrightarrow} j\tilde{X}_I[k]$ (a DFS da parte impar é a parte imaginária da DFS)

Propriedades da DFS

- Convolução Periódica:

$$\tilde{x}_1[n] \xleftrightarrow{DFS} \tilde{X}_1[k]$$, com período *N*

$$\tilde{x}_2[n] \xleftrightarrow{DFS} \tilde{X}_2[k]$$, com período *N*

$$\tilde{x}_3[n] \xleftrightarrow{DFS} \tilde{X}_1[k]\tilde{X}_2[k]$$

Convolução periódica: convolução em que a soma é calculada em um período.

$$\tilde{x}_3[n] = \sum_{m=0}^{N-1} \tilde{x}_1[m]\tilde{x}_2[n-m] = \sum_{m=0}^{N-1} \tilde{x}_2[m]\tilde{x}_1[n-m]$$

Da mesma forma, pode-se demonstrar que:

$$\tilde{x}_4[n] = \tilde{x}_1[n]\tilde{x}_2[n]$$

$$\tilde{x}_4[n] \xleftrightarrow{DFS} \tilde{X}_4[k]$$

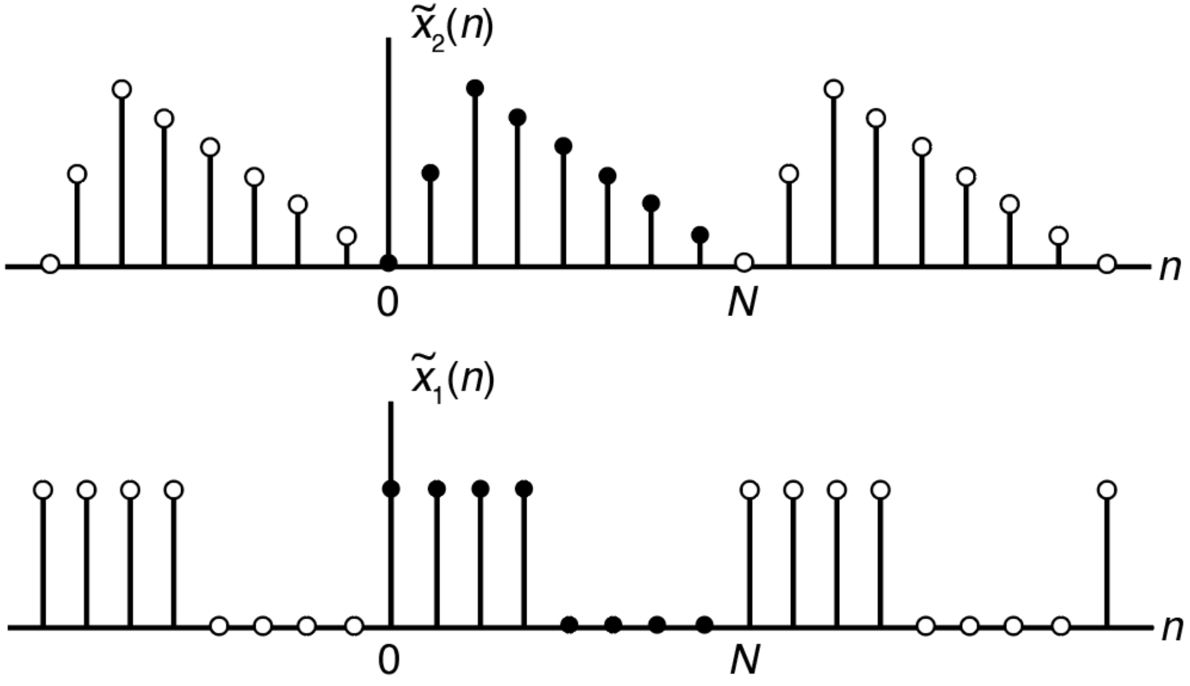
$$\tilde{X}_4[k] = \sum_{\ell=0}^{N-1} \tilde{X}_1[\ell]\tilde{X}_2[k-\ell] = \sum_{\ell=0}^{N-1} \tilde{X}_2[\ell]\tilde{X}_1[k-\ell]$$

Propriedades da DFS

- Convolução Periódica: Em resumo:

$$\sum_{m=0}^{N-1} \tilde{x}_1[m]\tilde{x}_2[n-m] \overset{DFS}{\longleftrightarrow} \tilde{X}_1[k]\tilde{X}_2[k] \qquad \tilde{x}_1[n]\tilde{x}_2[n] \overset{DFS}{\longleftrightarrow} \sum_{m=0}^{N-1} \tilde{X}_1[\ell]\tilde{X}_2[k-\ell]$$

- Interpretação gráfica da convolução periódica:

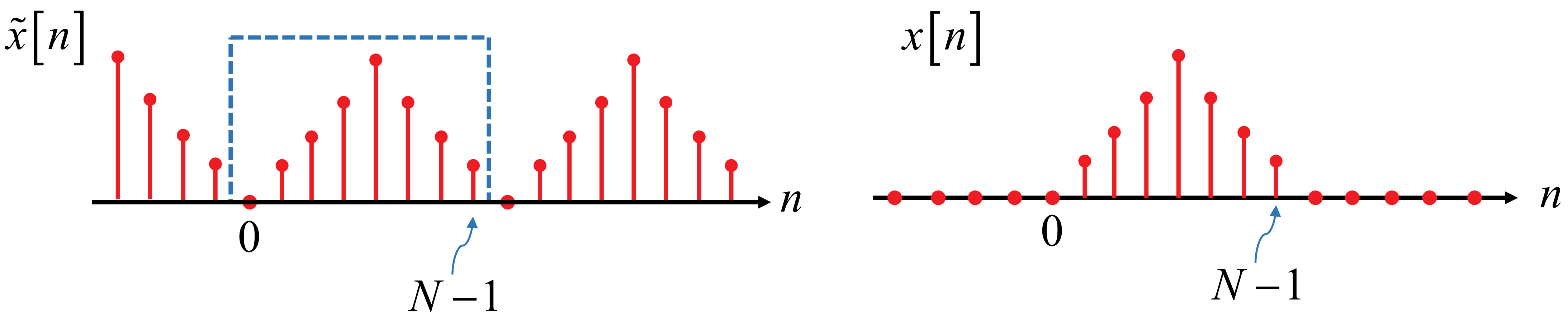


$$\tilde{x}_3[n] = \sum_{m=0}^{N-1} \tilde{x}_1[m]\tilde{x}_2[n-m]$$

A soma, neste caso é efetuada de 0 até *N-1*. O resultado da soma será $\tilde{x}_3[2]$

Relação da DFS com a DTFT

- Relação da DFS de $\tilde{x}[n]$ com a DTFT de $x[n]$:



Desta forma: $x[n] = \begin{cases} \tilde{x}[n] & 0 \le n \le N-1 \\ 0 & \text{c.c.} \end{cases}$

Transformada de Fourier de *x*[*n*]: $0 \le n \le N-1 \Rightarrow x[n] = \tilde{x}[n]$

$$X\left(e^{j\omega}\right) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} = \sum_{n=0}^{N-1} x[n]e^{-j\omega n} = \sum_{n=0}^{N-1} \tilde{x}[n]e^{-j\omega n}$$

Relação da DFS com a DTFT

- Relação da DFS de $\tilde{x}[n]$ com a DTFT de $x[n]$:

Transformada de Fourier de $x[n]$: $X\left(e^{j\omega}\right) = \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j\omega n}$

DFS de $\tilde{x}[n]$: $\tilde{X}[k] = \sum_{n=0}^{N-1} \tilde{x}[n] W_N^{nk} = \sum_{n=0}^{N-1} \tilde{x}[n] e^{-j\left(\frac{2\pi}{N}k\right)n}$

Por comparação: $\tilde{X}[k] = X\left(e^{j\omega}\right)\Big|_{\omega=\frac{2\pi}{N}k}$

A DFS de $\tilde{x}[n]$ consiste em uma versão amostrada da DTFT de $x[n]$.

Transformada Discreta de Fourier

- $\tilde{x}[n]$ periódica → Admite a série discreta de Fourier.

$$\tilde{X}[k]=\sum_{n=0}^{N-1}\tilde{x}[n]W_N^{nk} \qquad \tilde{x}[n]=\frac{1}{N}\sum_{k=0}^{N-1}\tilde{X}[k]W_N^{-nk}$$

- A DFS de $\tilde{x}[n]$ consiste em uma versão amostrada da DTFT de *x*(*n*):

$$\tilde{X}[k]=X\left(e^{j\omega}\right)\Big|_{\omega=\frac{2\pi}{N}k} \qquad X\left(e^{j\omega}\right)=\mathcal{F}\{x[n]\}$$

- Como calcular computacionalmente a DTFT de *x*[*n*] (duração finita)?
  - Converter *x*[*n*] em uma sequência periódica $\tilde{x}[n]$;
  - Calcular a DFS de $\tilde{x}[n]$, obtendo $\tilde{X}[k]$;
  - Entender os coeficientes $\tilde{X}[k]$ em um período como sendo a amostragem da DTFT de *x*[*n*] - $X\left(e^{j\omega}\right)$ em um período.

TRANSFORMADA DISCRETA DE FOURIER (DFT)

Transformada Discreta de Fourier

- DFT direta: $$X[k] = \begin{cases} \tilde{X}[k] & 0 \le k \le N-1 \\ 0 & \text{c.c.} \end{cases} = \begin{cases} \sum_{n=0}^{N-1} \tilde{x}[n] W_N^{nk} & 0 \le k \le N-1 \\ 0 & \text{c.c.} \end{cases}$$
$$= \begin{cases} \sum_{n=0}^{N-1} x[n] W_N^{nk} & 0 \le k \le N-1 \\ 0 & \text{c.c.} \end{cases}$$

- DFT inversa: $$x[n] = \begin{cases} \tilde{x}[n] & 0 \le n \le N-1 \\ 0 & \text{c.c.} \end{cases} = \begin{cases} \frac{1}{N} \sum_{k=0}^{N-1} \tilde{X}[k] W_N^{-nk} & 0 \le n \le N-1 \\ 0 & \text{c.c.} \end{cases}$$
$$= \begin{cases} \frac{1}{N} \sum_{k=0}^{N-1} X[k] W_N^{-nk} & 0 \le n \le N-1 \\ 0 & \text{c.c.} \end{cases}$$

Transformada Discreta de Fourier

Exemplo 2: Determine a DFT do seguinte sinal:

$$x[n] = a^n \quad 0 \le n \le N-1$$

Resolução Espectral

- Do exemplo 2:

$$x[n] = a^n u[n] \quad \begin{cases} X(e^{j\omega}) = \dfrac{1}{1 - ae^{-j\omega}} \\ X[k] = \dfrac{1 - a^N}{1 - aW_N^k} \quad 0 \le k \le N-1 \end{cases}$$

$$\bar{X}[k] = X(e^{j\omega})\Big|_{\omega = \frac{2\pi k}{N}} = \frac{1}{1 - ae^{-j\left(\frac{2\pi}{N}\right)k}} = \frac{1}{1 - aW_N^k}$$

$$\lim_{N\to\infty} X[k] = \lim_{N\to\infty} \frac{1 - a^N}{1 - aW_N^k} = \frac{1}{1 - aW_N^k} = \bar{X}[k] \quad \longrightarrow \quad \lim_{N\to\infty} X[k] = \bar{X}[k]$$

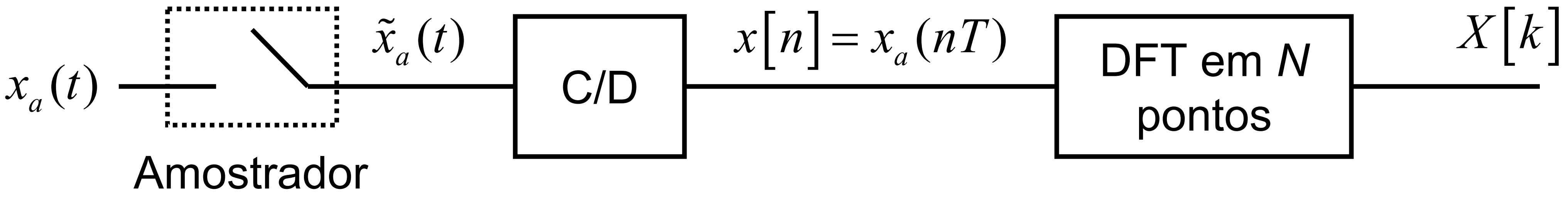
A DFT é a versão amostrada da DTFT (quando $N \to \infty$ para sinais de duração infinita)

Resolução Espectral

- Resolução Espectral → Espaço (em frequência) entre duas amostras da DFT.

$$\Delta\omega = \frac{2\pi}{N}$$

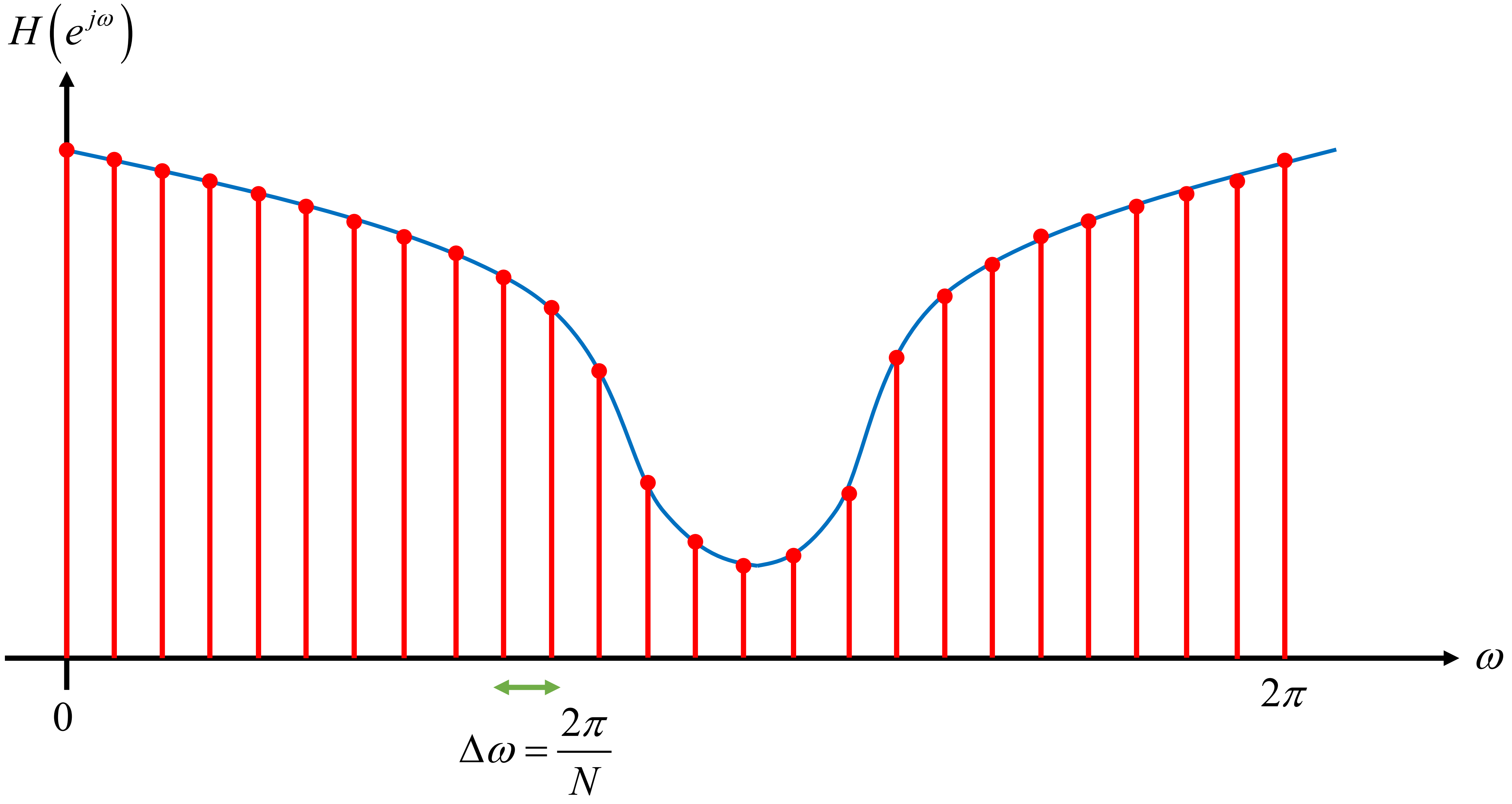
- Resolução espectral em sinais de tempo contínuo:



$$\Delta\Omega = \frac{\Delta\omega}{T} = \left(\frac{2\pi}{N}\right)\left(\frac{1}{T}\right)$$

Mas: $\Delta\Omega = 2\pi\Delta F$

$$2\pi\Delta F = \left(\frac{2\pi}{N}\right)\left(\frac{1}{T}\right) \quad \rightarrow \quad \Delta F = \frac{1}{NT} = \frac{F_s}{N}$$



$H(e^{j\omega})$
ω
0
2π
$\Delta\omega = \frac{2\pi}{N}$

Propriedades da DFT

- Linearidade:

$$\left.\begin{array}{l} x_1[n] \overset{DFT}{\rightleftharpoons} X_1[k] \\ x_2[n] \overset{DFT}{\rightleftharpoons} X_2[k] \end{array}\right\} \quad ax_1[n]+bx_2[n] \overset{DFT}{\rightleftharpoons} aX_1[k]+bX_2[k]$$

- Deslocamento Circular:

DFS: $\tilde{x}[n] \overset{DFS}{\rightleftharpoons} \tilde{X}[k]$

$$\tilde{x}_1[n]=\tilde{x}[n+m] \quad \Rightarrow \quad \tilde{x}_1[n] \overset{DFS}{\rightleftharpoons} W_N^{-km}\tilde{X}[k]$$

DFT: $x[n] \overset{DFT}{\rightleftharpoons} X[k]$

$$\tilde{x}[n]=x\left[((n))_N\right]$$

Deseja-se saber qual sequência tem como: $x_1[n] \overset{DFT}{\rightleftharpoons} W_N^{-km}X[k]$

$$x_1[n]=\tilde{x}_1[n]\mathcal{R}_N[n]$$

Propriedades da DFT

- Deslocamento Circular:

Mas: $x_1[n] = \tilde{x}_1[n]\mathcal{R}_N[n]$

$= \tilde{x}[n+m]\mathcal{R}_N[n]$

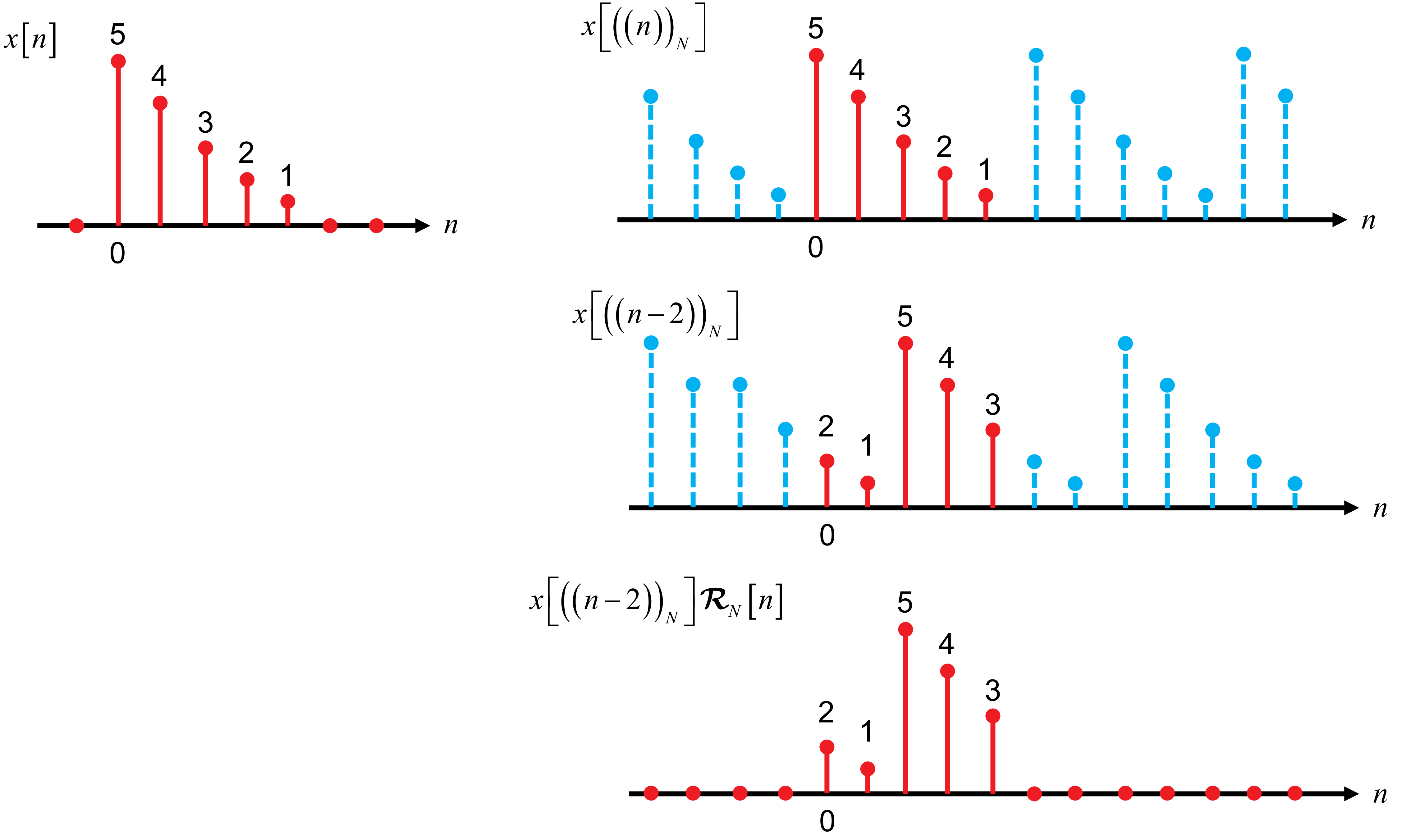
$= x\left[((n+m))_N\right]\mathcal{R}_N[n]$

Deslocamento Circular

Desta forma, $x[n] \xrightleftharpoons{DFT} X[k]$

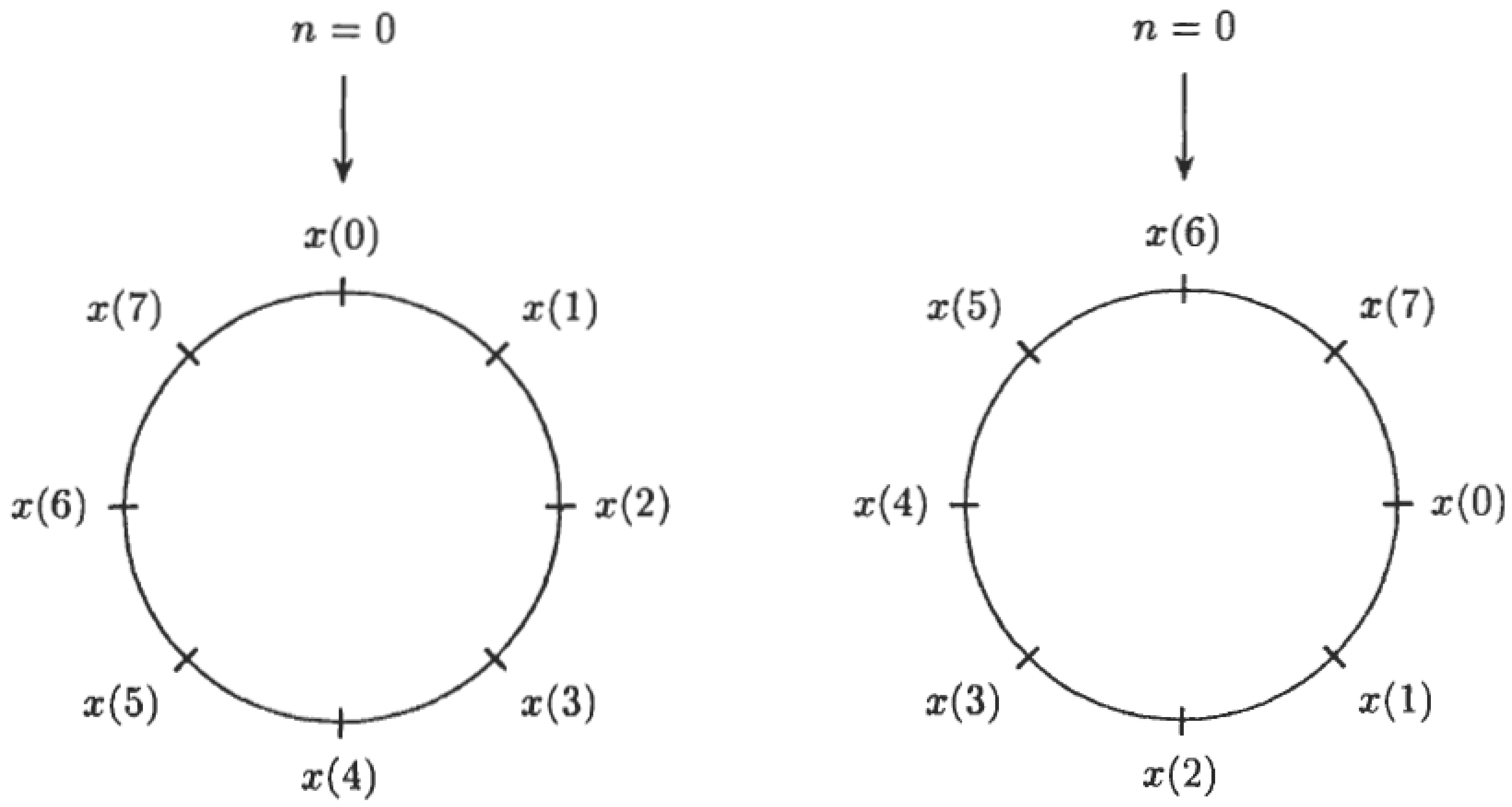
$$x\left[((n+m))_N\right]\mathcal{R}_N[n] \xrightleftharpoons{DFT} W_N^{-km}X[k]$$

De forma equivalente: $W_N^{\ell n}x[n] \xrightleftharpoons{DFT} X\left[((k+\ell))_N\right]\mathcal{R}_N[k]$



$x[n]$
5
4
3
2
1
0
n
$x\left[((n))_N\right]$
5
4
3
2
1
0
n
$x\left[((n-2))_N\right]$
5
4
3
2
1
0
n
$x\left[((n-2))_N\right]\mathcal{R}_N[n]$
5
4
3
2
1
0
n

Visualização gráfica de um deslocamento circular de duas unidades:



Propriedades da DFT

- Simetria: $x[n]$ real → $\tilde{x}[n]$ real.

Da DFS: $\tilde{X}[k] = \tilde{X}^*[-k]$

$\tilde{X}_R[k] = \tilde{X}_R[N-k]$

$\tilde{X}_I[k] = -\tilde{X}_I[N-k]$

Para a DFT: P1: $X[k] = X^*\left[((-k))_N\right]\mathcal{R}_N[k]$

P2: $X_R[k] = X_R\left[((N-k))_N\right]\mathcal{R}_N[k]$

P3: $X_I[k] = -X_I\left[((N-k))_N\right]\mathcal{R}_N[k]$

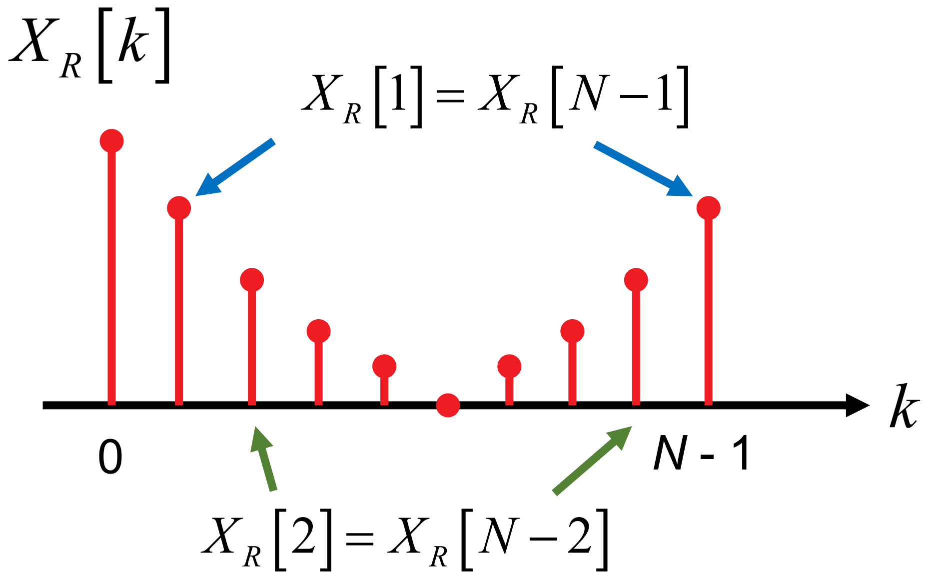
Exemplo:

$$X_R[k] = X_R\left[((N-k))_N\right]\mathcal{R}_N[k]$$

$$X_R[0] = X_R\left[((N))_N\right]\mathcal{R}_N[0] = X_R[0]$$

$$X_R[1] = X_R\left[((N-1))_N\right]\mathcal{R}_N[1] = X_R[N-1]$$

$$X_R[2] = X_R\left[((N-2))_N\right]\mathcal{R}_N[2] = X_R[N-2]$$



Propriedades da DFT

- Simetria: $x[n]$ real → $\tilde{x}[n]$ real.

Da DFS: $\left|\tilde{X}[k]\right| = \left|\tilde{X}[N-k]\right|$

$\measuredangle\tilde{X}[k] = -\measuredangle\tilde{X}[N-k]$

Para a DFT: P4: $\left|X[k]\right| = \left|X\left[((N-k))_N\right]\right|\mathcal{R}_N[k]$

P5: $\measuredangle X[k] = -\left(\measuredangle X\left[((N-k))_N\right]\right)\mathcal{R}_N[k]$

Da DFS: $\tilde{x}_e[n] \overset{DFS}{\rightleftharpoons} \tilde{X}_R[k]$

$\tilde{x}_o[n] \overset{DFS}{\rightleftharpoons} j\tilde{X}_I[k]$

$$\left\{ \tilde{x}_e[n] = \frac{1}{2}\left[\tilde{x}[n] + \tilde{x}[-n]\right] \qquad \tilde{x}_o[n] = \frac{1}{2}\left[\tilde{x}[n] - \tilde{x}[-n]\right] \right.$$

Para a DFT: P6: $x_{ep}[n] \overset{DFT}{\rightleftharpoons} X_R[k]$

P7: $x_{op}[n] \overset{DFT}{\rightleftharpoons} jX_I[k]$

$$\left\{ \begin{aligned} x_{ep}[n] &= \frac{1}{2}\left[x\left[((n))_N\right] + x\left[((-n))_N\right]\right]\mathcal{R}_N[n] \\ x_{op}[n] &= \frac{1}{2}\left[x\left[((n))_N\right] - x\left[((-n))_N\right]\right]\mathcal{R}_N[n] \end{aligned} \right.$$

Propriedades da DFT

- Convolução Circular:

$$x_1[n] \overset{DFT}{\rightleftharpoons} X_1[k]$$

$$x_2[n] \overset{DFT}{\rightleftharpoons} X_2[k]$$

$$x_3[n] \overset{DFT}{\rightleftharpoons} X_1[k]X_2[k]$$

Da DFS:

$$\tilde{x}_1[n] \overset{DFS}{\rightleftharpoons} \tilde{X}_1[k]$$

$$\tilde{x}_2[n] \overset{DFS}{\rightleftharpoons} \tilde{X}_2[k]$$

$$\tilde{x}_3[n] \overset{DFS}{\rightleftharpoons} \tilde{X}_1[k]\tilde{X}_2[k]$$

$$\tilde{x}_3[n] = \sum_{m=0}^{N-1} \tilde{x}_1[m]\tilde{x}_2[n-m] = \sum_{m=0}^{N-1} \tilde{x}_2[m]\tilde{x}_1[n-m]$$

Logo:

$$x_3[n] = \tilde{x}_3[n]\mathcal{R}_N[n]$$

$$= \left[\sum_{m=0}^{N-1} \tilde{x}_1[m]\tilde{x}_2[n-m]\right]\mathcal{R}_N[n]$$

$$0 \le m \le N-1 \quad \Rightarrow \quad m \text{ modulo } N = m$$

$$= \left[\sum_{m=0}^{N-1} x_1\left[((m))_N\right]x_2\left[((n-m))_N\right]\right]\mathcal{R}_N[n]$$

Convolução Circular

$$= \left[\sum_{m=0}^{N-1} x_1[m]x_2\left[((n-m))_N\right]\right]\mathcal{R}_N[n] = \left[x_1[n] * x_2\left[((n))_N\right]\right]\mathcal{R}_N[n]$$

Propriedades da DFT

- Convolução Circular:

$$x_3[n] = \left[\sum_{m=0}^{N-1} x_1[m]\,x_2\left[((n-m))_N\right]\right]\mathcal{R}_N[n] = x_1[n] \,\textcircled{N}\, x_2[n]$$

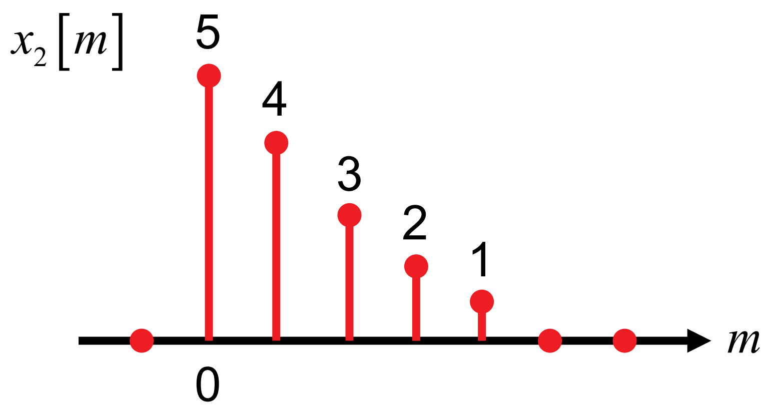
Ou seja:

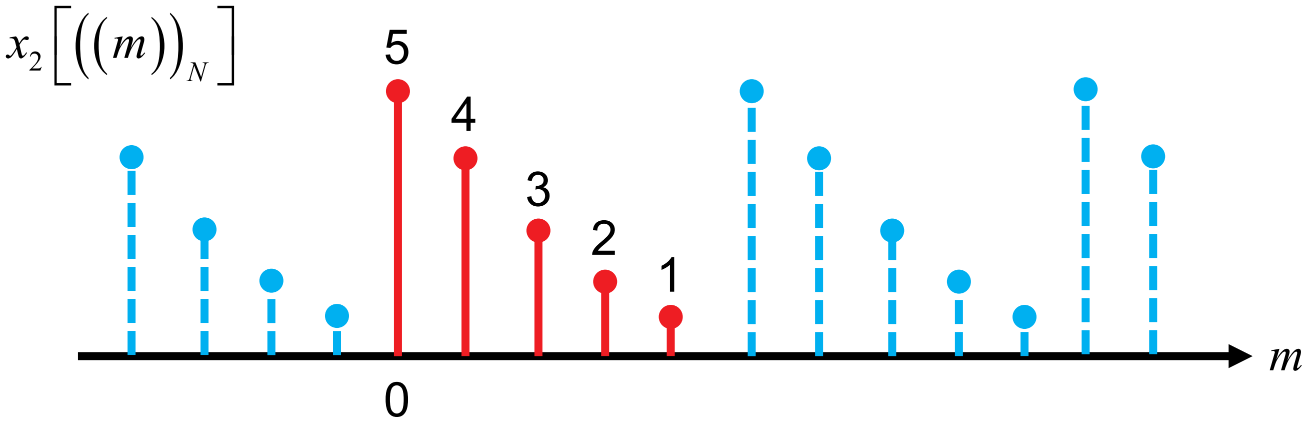
$$x_1[n] \,\textcircled{N}\, x_2[n] \quad \overset{DFT}{\rightleftharpoons} X_1[k]\,X_2[k]$$

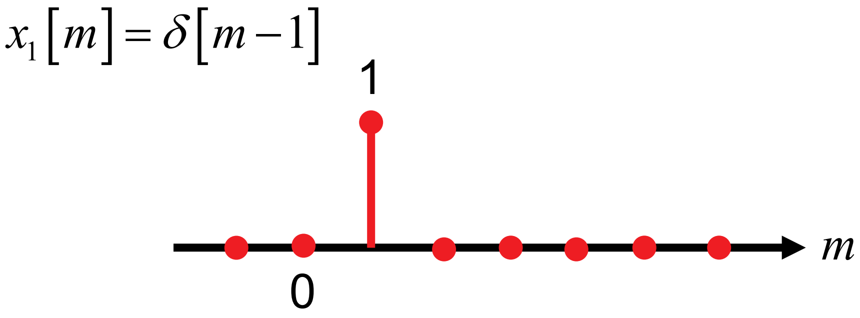
$$x_1[n]\,x_2[n] \overset{DFT}{\rightleftharpoons} \frac{1}{N}\left[\sum_{\ell=0}^{N-1} X_1[\ell]\,X_2\left[((k-\ell))_N\right]\right]\mathcal{R}_N[k] = \frac{1}{N} X_1[k] \,\textcircled{N}\, X_2[k]$$

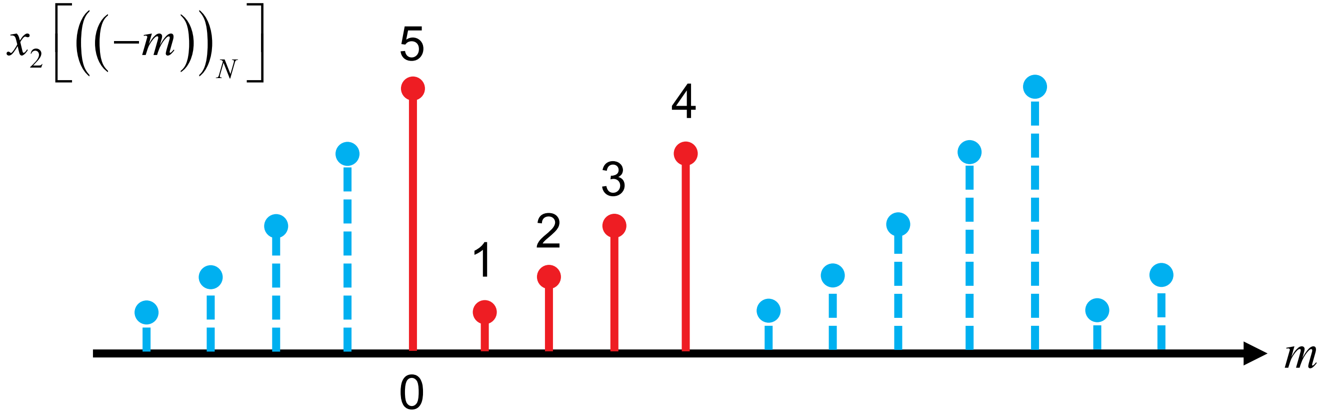
Convolução Circular

Exemplo:





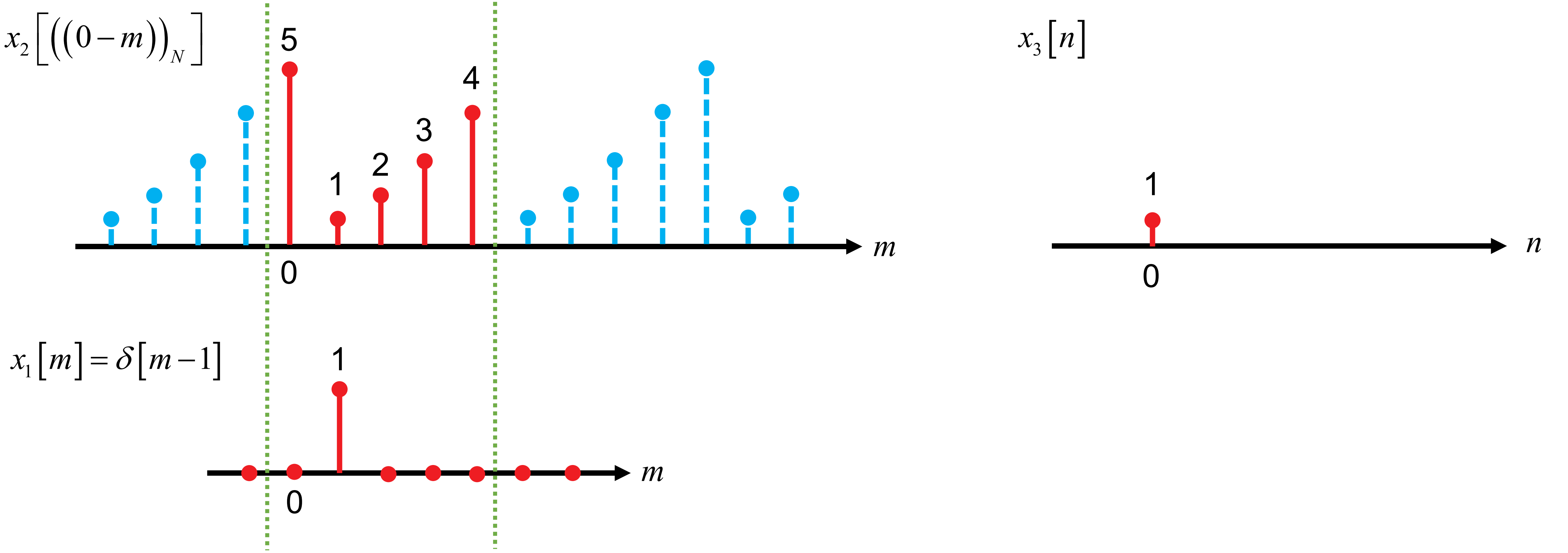




$$x_3[n]=\left[\sum_{m=0}^{N-1} x_1[m]\,x_2\left[((n-m))_N\right]\right]\mathcal{R}_N[n]=\left[x_1[n]*x_2\left[((n))_N\right]\right]\mathcal{R}_N[n]$$

Convolução Circular

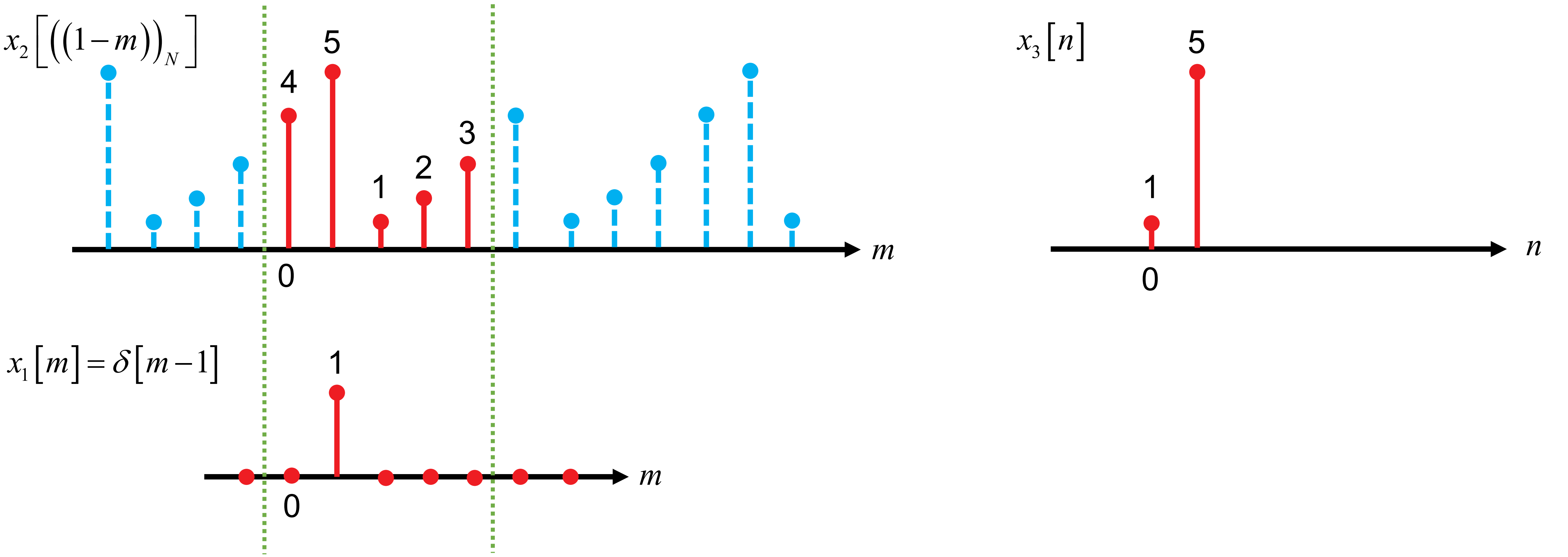
Exemplo:



Intervalo de soma

Convolução Circular

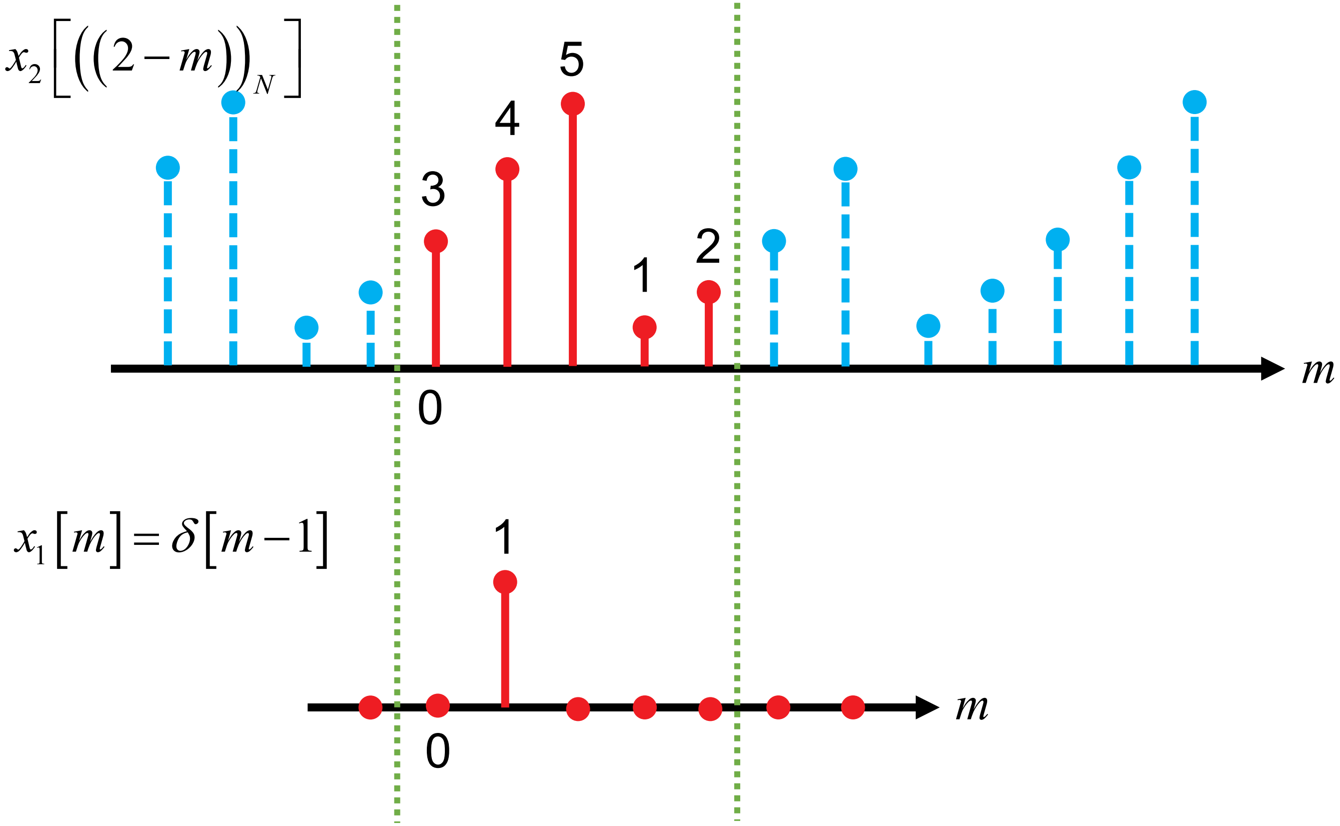
Exemplo:

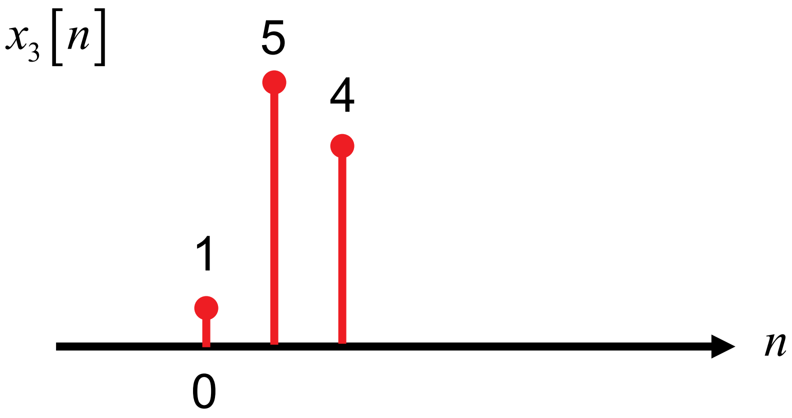


Intervalo de soma

Convolução Circular

Exemplo:

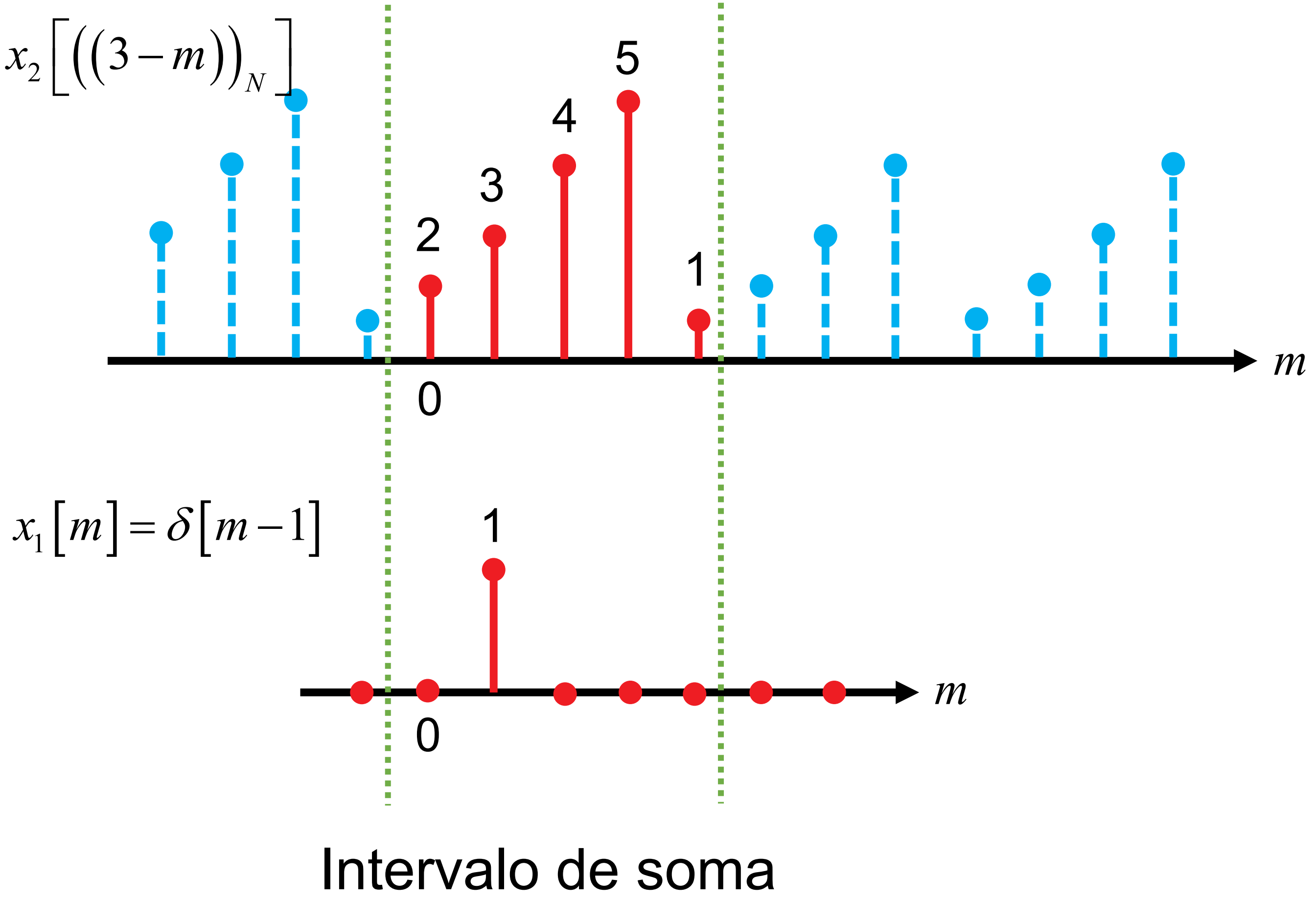


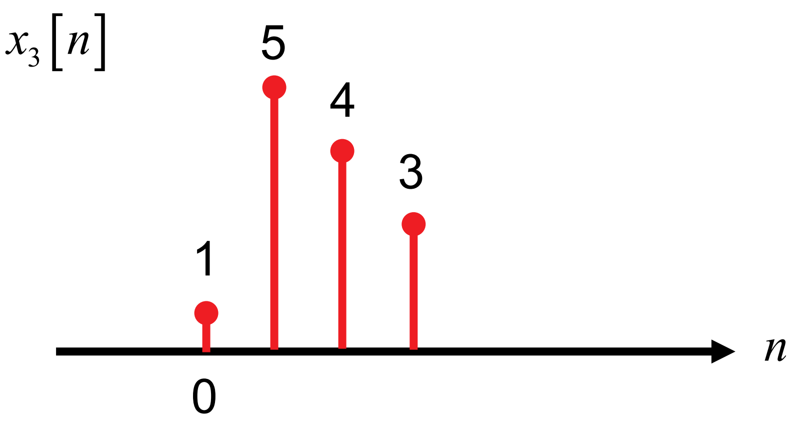


Intervalo de soma

Convolução Circular

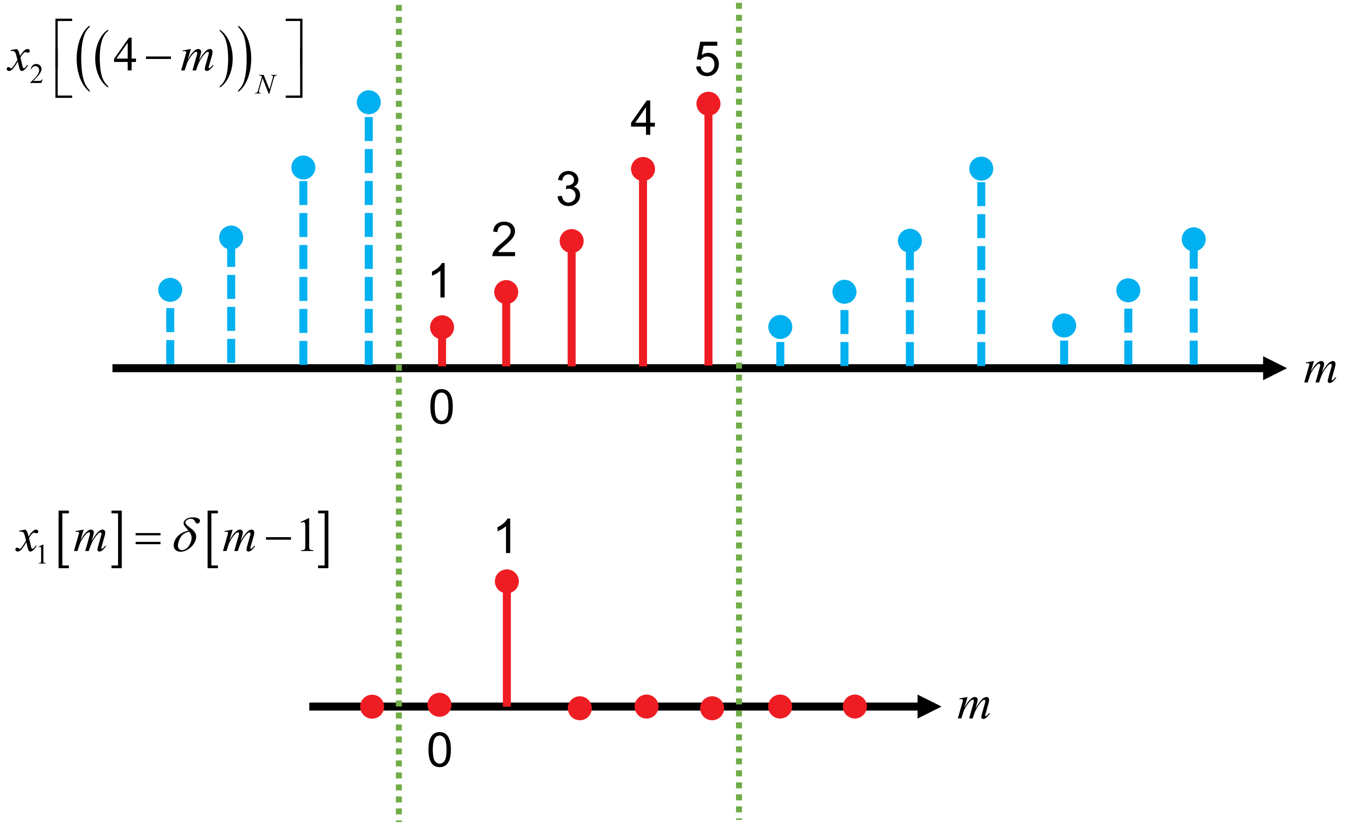
Exemplo:



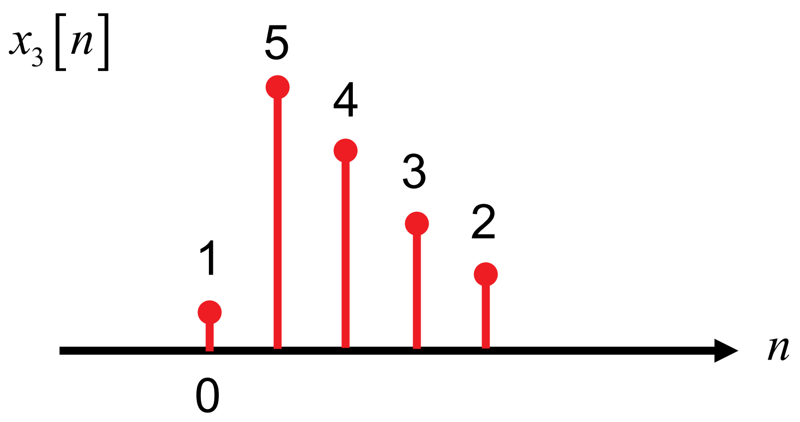


Convolução Circular

Exemplo:

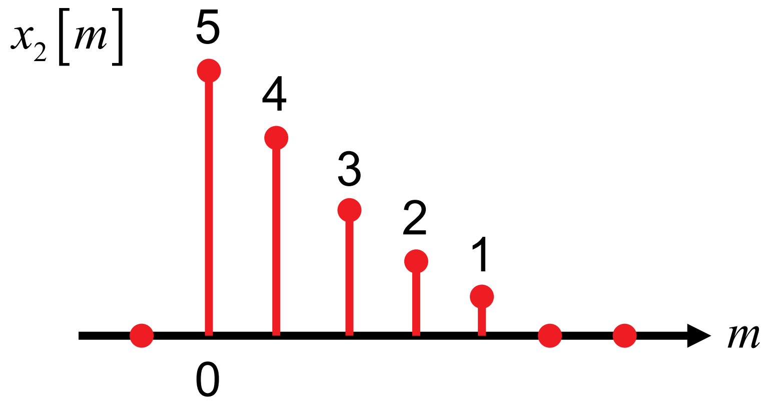


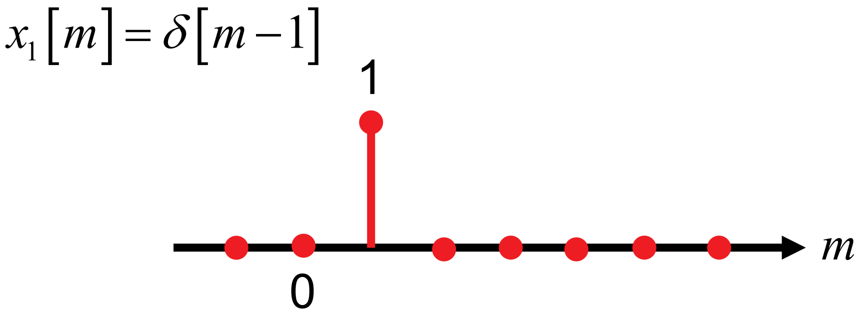
Intervalo de soma

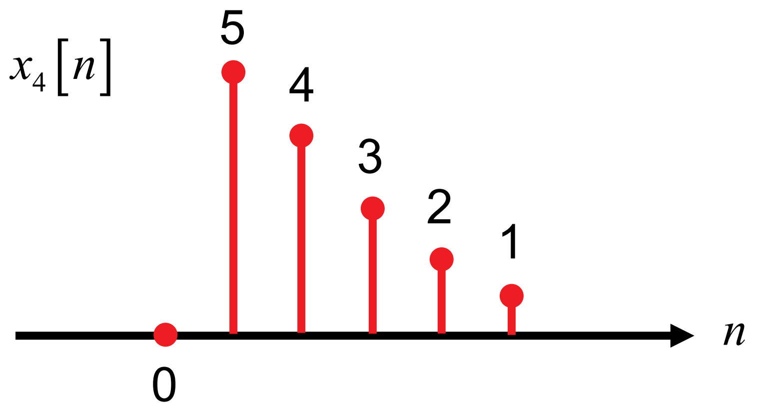


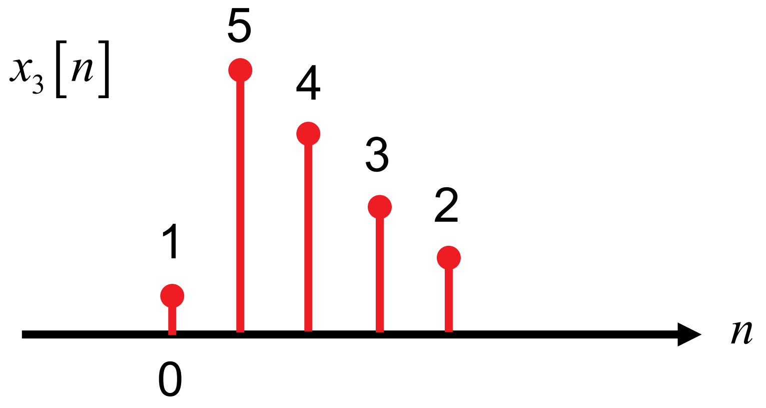
Convolução Circular

Exemplo: Convolução Linear









Convolução circular ≠ Convolução linear

$$x_4[n] = x_1[n] * x_2[n] = \delta[n-1] * x_2[n] = x_2[n-1]$$

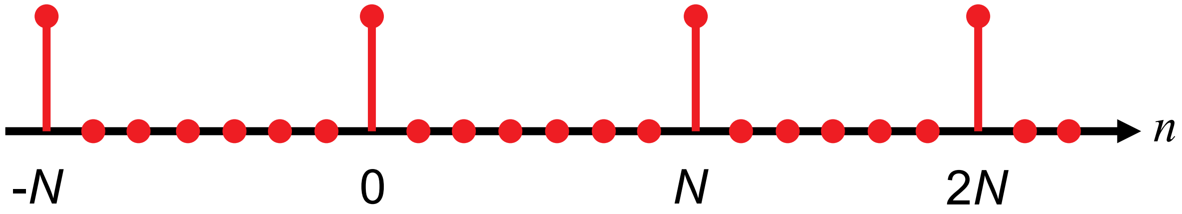
Relação entre a Convolução Circular e Convolução Linear

- Definição da convolução circular:

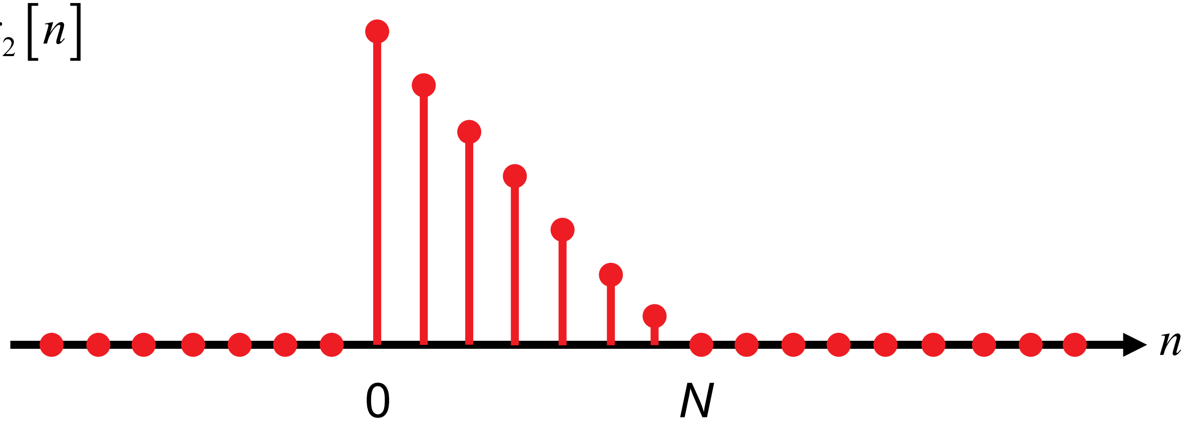
$$x_3[n] = \left[\sum_{m=0}^{N-1} x_1[m]\, x_2\left[((n-m))_N\right]\right]\mathcal{R}_N[n] = \left[x_1[n] * x_2\left[((n))_N\right]\right]\mathcal{R}_N[n]$$

- Forma para obter $x_2[((n))_N]$: $x_2\left[((n))_N\right] = x_2[n] * p_N[n]$

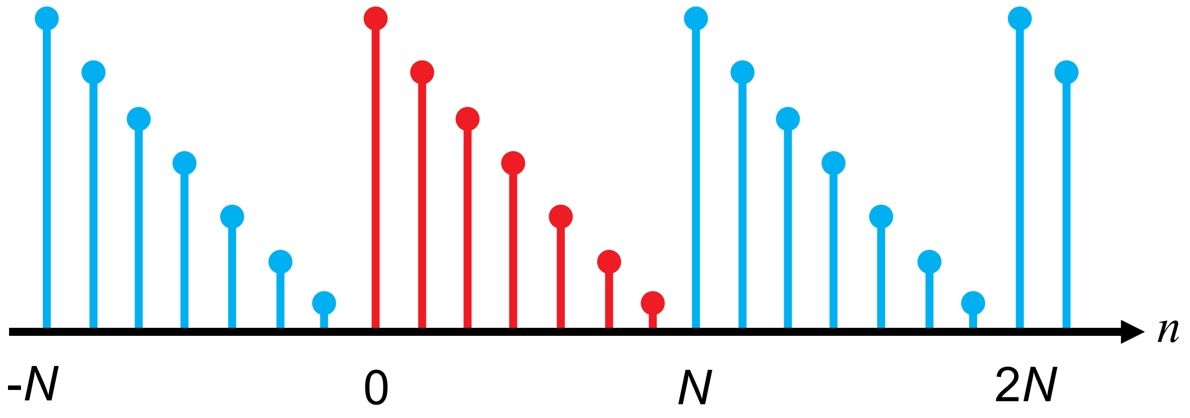
$$p_N[n] = \sum_{r=-\infty}^{\infty} \delta[n - rN]$$



$x_2[n]$

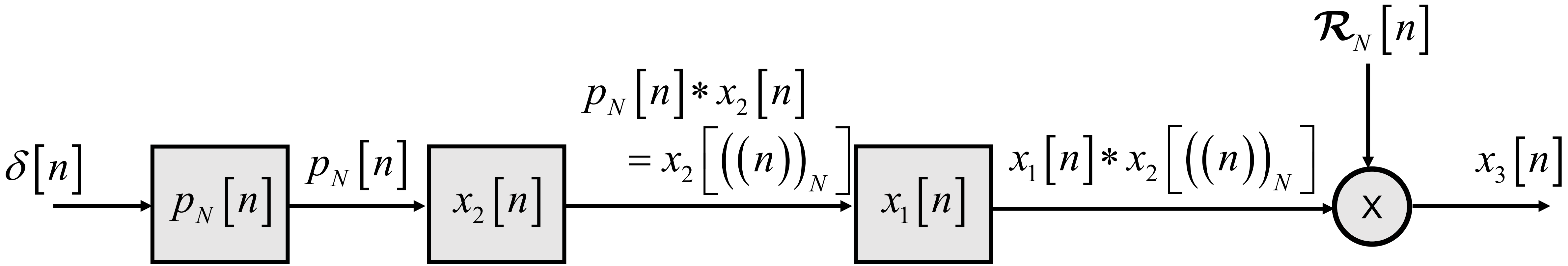


$x_2[n] * p_N[n] = x_2\left[((n))_N\right]$

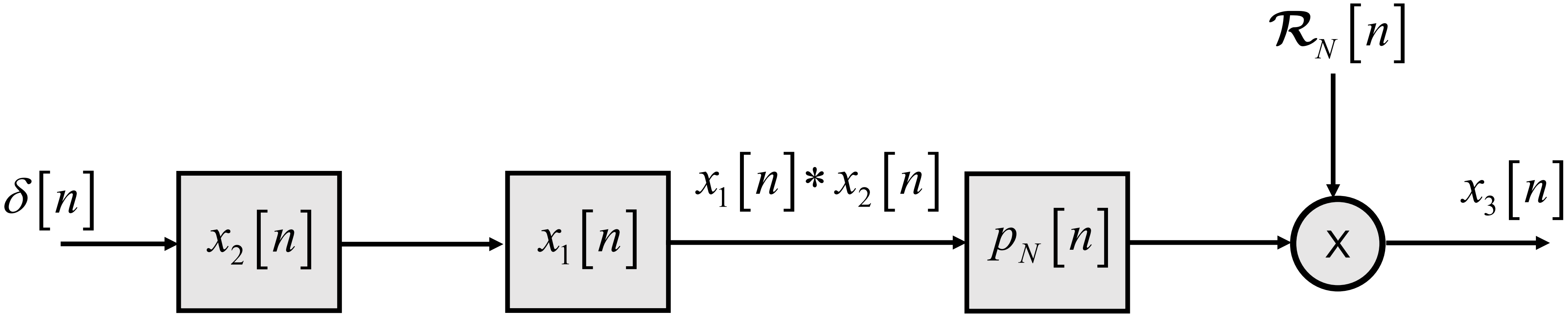


Relação entre a Convolução Circular e Convolução Linear

- Diagrama de blocos para convolução circular: $x_3[n] = \left[x_1[n] * x_2\left[((n))_N\right] \right] \mathcal{R}_N[n]$

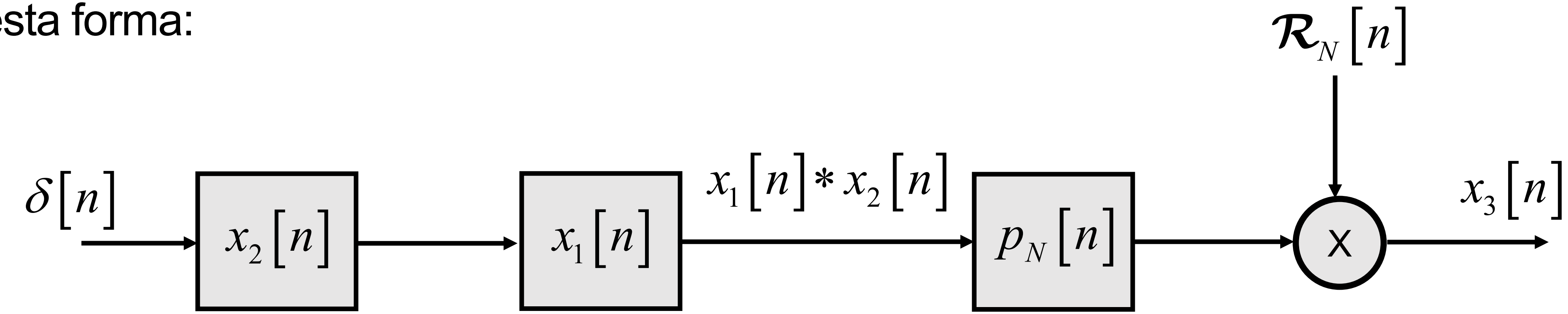


- Reorganizando (sistema LTI):



Relação entre a Convolução Circular e Convolução Linear

- Desta forma:



$$x_3[n] = x_1[n] \, \text{Ⓝ} \, x_2[n]$$

$$\hat{x}_3[n] = x_1[n] * x_2[n]$$

Convolução circular = Convolução linear + *aliasing*

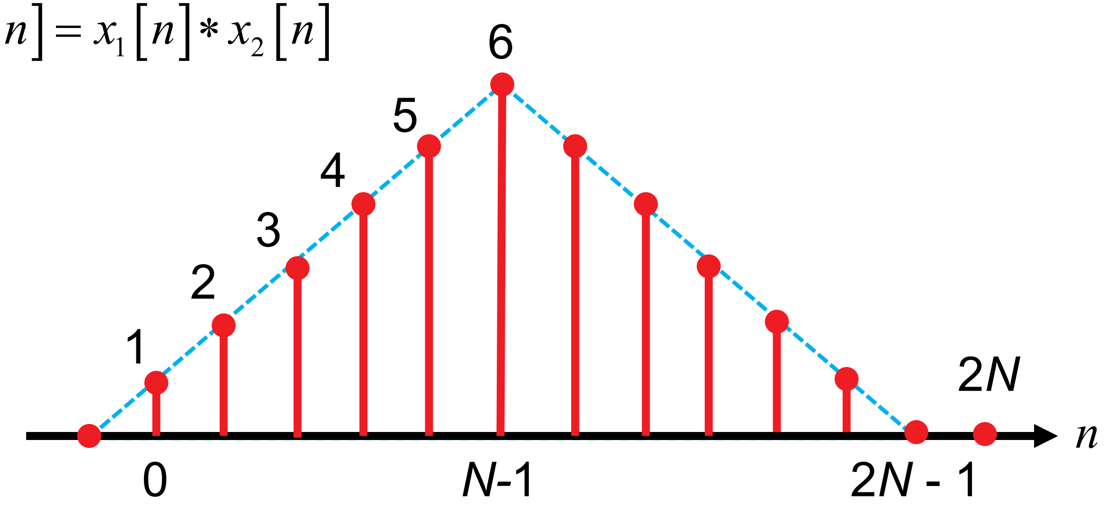
$$x_3[n] = \left[\hat{x}_3[n] * p_N[n]\right] \mathcal{R}_N[n]$$

$$= \left[\sum_{r=-\infty}^{\infty} \hat{x}_3[n + rN]\right] \mathcal{R}_N[n]$$

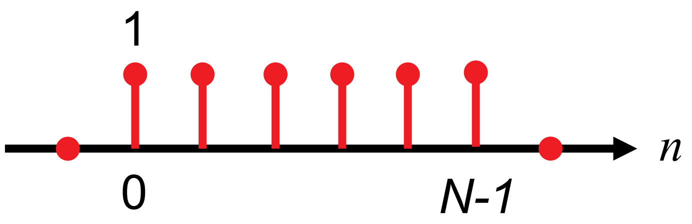
Relação entre a Convolução Circular e Convolução Linear

Exemplo:

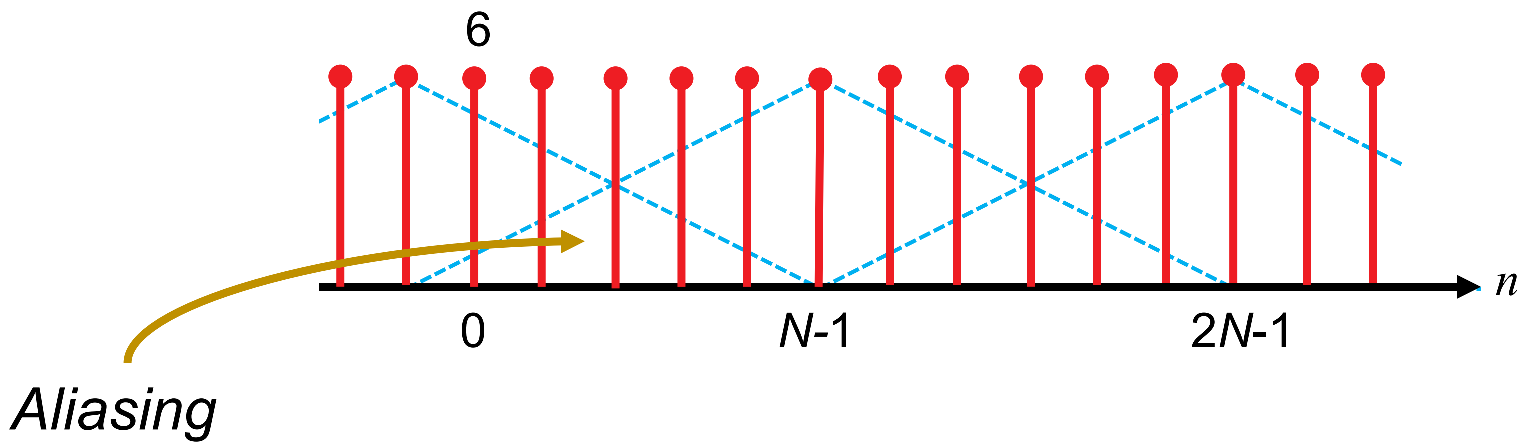
$$\hat{x}_3[n] = x_1[n] * x_2[n]$$



$$x_1[n] = x_2[n]$$



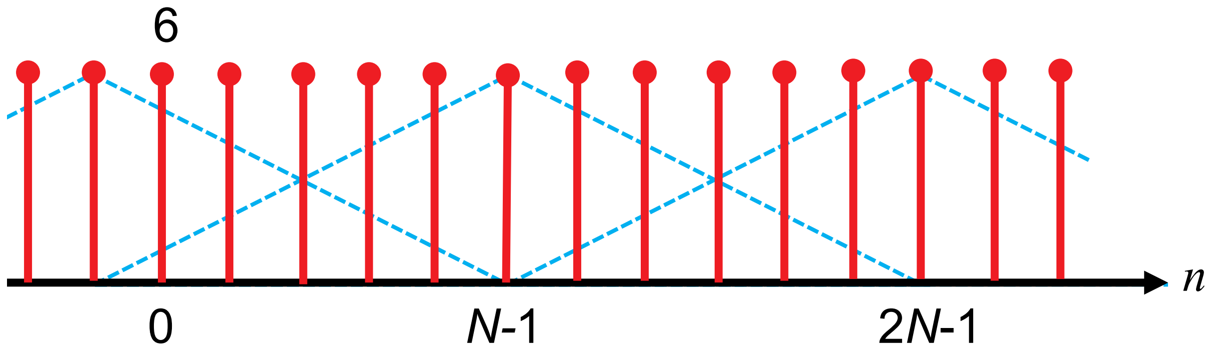
$$\left[x_1[n] * x_2[n]\right] * p_N[n] \;= \hat{x}_3[n] * p_N[n] \;= \sum_{r=-\infty}^{\infty} \hat{x}_3[n+rN]$$



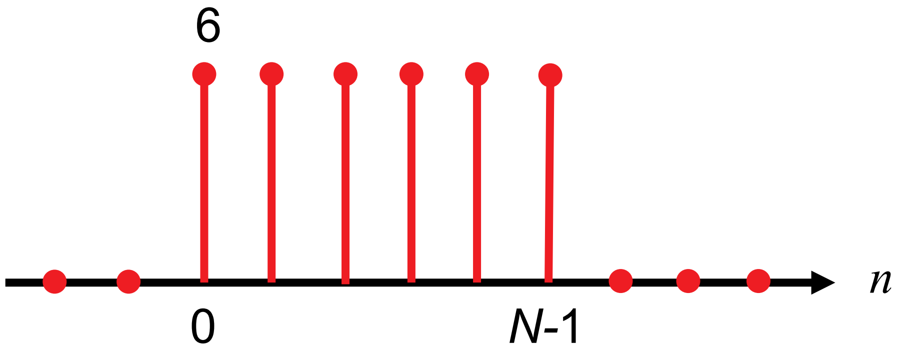
Relação entre a Convolução Circular e Convolução Linear

Exemplo:

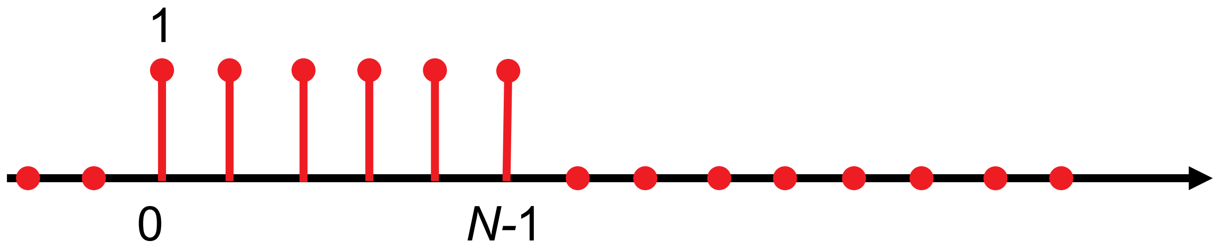
$$x_1[n] * x_2[n] * p_N[n] = \sum_{r=-\infty}^{\infty} \hat{x}_3[n+rN]$$



$$\left[\sum_{r=-\infty}^{\infty} \hat{x}_3[n+rN]\right]\mathcal{R}_N[n] = x_1[n] \; N \; x_2[n]$$



$\mathcal{R}_N[n]$



A convolução circular não é igual à convolução linear.

Cálculo da Convolução Linear utilizando Convolução Circular

- Objetivo:

Convolução linear entre $x_1[n]$ e $x_2[n]$ = Convolução circular entre $x_1[n]$ e $x_2[n]$

- Se isso for atendido, poderemos calcular computacionalmente a convolução a partir das operações:

$$X_1[k] = \mathrm{DFT}\{x_1[n]\}$$
$$X_2[k] = \mathrm{DFT}\{x_2[n]\}$$
$$X_3[k] = X_1[k]X_2[k]$$

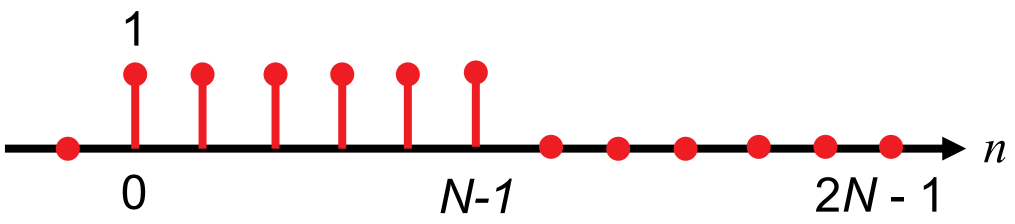
$$x_3[n] = \mathrm{IDFT}\{X_3[k]\} = x_1[n] * x_2[n]$$

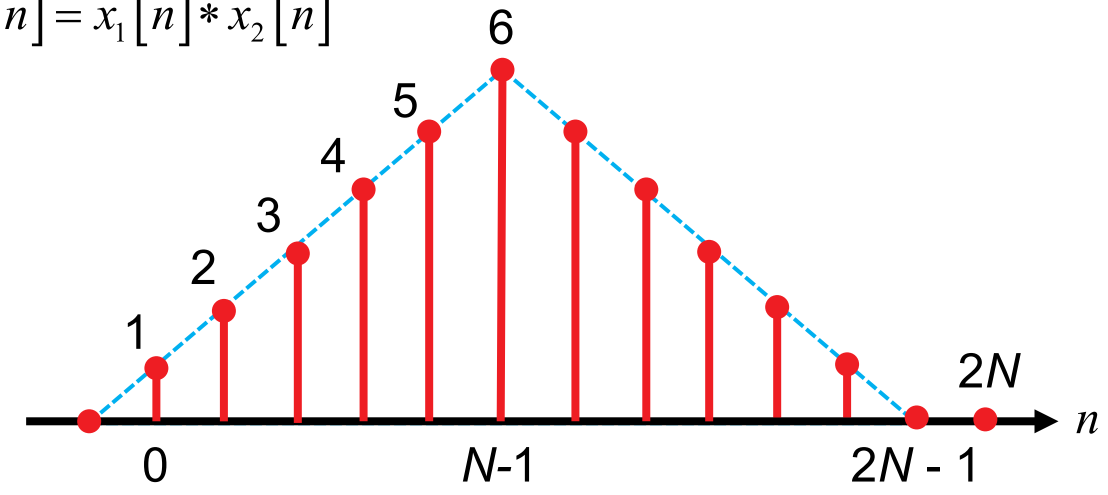
Cálculo da Convolução Linear utilizando Convolução Circular

Exemplo:

$$\hat{x}_3[n] = x_1[n] * x_2[n]$$

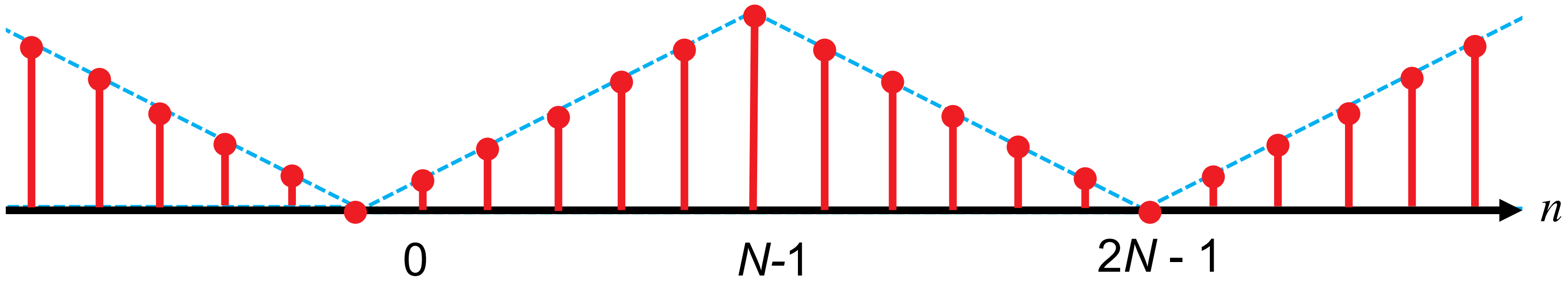
$$x_1[n] = x_2[n]$$





$$\left[x_1[n] * x_2[n]\right] * p_{2N}[n] = \hat{x}_3[n] * p_{2N}[n] = \sum_{r=-\infty}^{\infty} \hat{x}_3[n + r2N]$$

Convolução circular calculada em $2N$

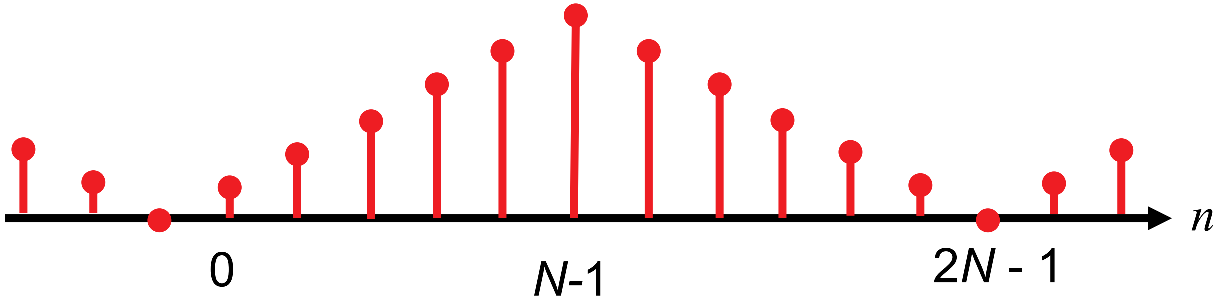


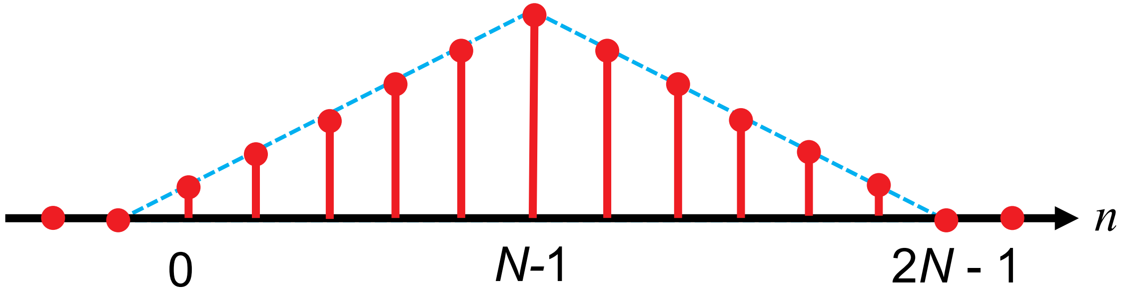
Cálculo da Convolução Linear utilizando Convolução Circular

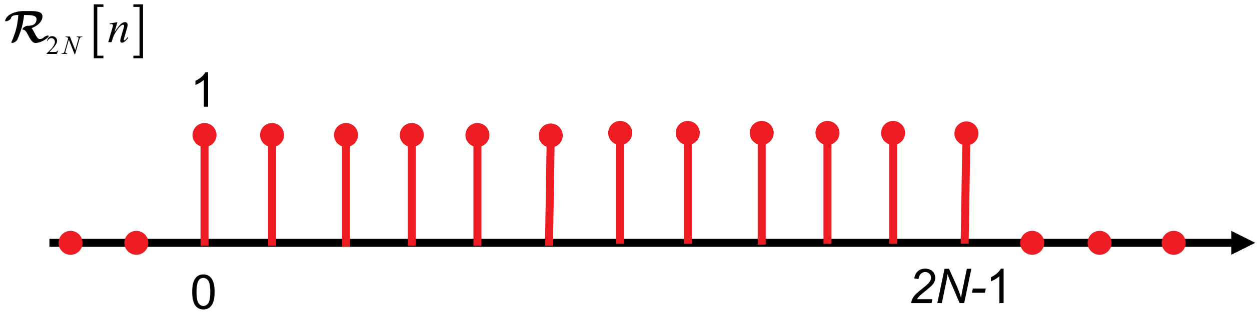
Exemplo:

$$\left[x_1[n] * x_2[n]\right] * p_{2N}[n] = \sum_{r=-\infty}^{\infty} \hat{x}_3[n + r2N]$$

$$\left[\sum_{r=-\infty}^{\infty} \hat{x}_3(n + r2N)\right]\mathcal{R}_{2N}[n] = x_1[n] \; 2N \; x_2[n]$$







A convolução circular é igual à convolução linear.

Cálculo da Convolução Linear utilizando Convolução Circular

- Para que a convolução circular seja igual à convolução linear, basta que não haja sobreposições das réplicas (ou seja, não ocorrer o *aliasing*);
- Regra:

$$N \geq L \quad \begin{cases} N \text{ é o tamanho da convolução circular} \\ L \text{ é o tamanho da convolução linear} \end{cases}$$

- Sendo:

$$\left.\begin{array}{l} x_1[n] \rightarrow \text{tamanho } N_1 \\ x_2[n] \rightarrow \text{tamanho } N_2 \end{array}\right\} L = N_1 + N_2 - 1$$

$$N \geq N_1 + N_2 - 1$$

Cálculo da Convolução Linear utilizando Convolução Circular

- Procedimento de Cálculo da Convolução Linear utilizando a DFT:
  - Sendo: $x_1[n] \longrightarrow$ tamanho N_1
    $x_2[n] \longrightarrow$ tamanho N_2

    $$\left.\begin{matrix} x_1[n] \rightarrow \text{tamanho } N_1 \\ x_2[n] \rightarrow \text{tamanho } N_2 \end{matrix}\right\} \begin{matrix} L = N_1 + N_2 \\ N \geq N_1 + N_2 - 1 \end{matrix}$$
  - Passo 1 – Realizar o preenchimento nulo de $x_1[n]$ e $x_2[n]$ de forma que o tamanho resultante seja igual a N.
  - Passo 2 – Calcular a DFT de $x_1[n]$ e $x_2[n]$: $\begin{cases} X_1[k] = \text{DFT}\{x_1[n]\} \\ X_2[k] = \text{DFT}\{x_2[n]\} \end{cases}$
  - Passo 3 – Calcular $X_3[k]$ como: $X_3[k] = X_1[k]X_2[k]$
  - Passo 4 – Calcular $x_3[n]$ como:

$$x_3[n] = \text{IDFT}\{X_3[k]\} = x_1[n] * x_2[n]$$

Como não há *aliasing*, então $x_3[n]$ é igual a convolução linear de $x_1[n]$ e $x_2[n]$.

Convolução em Bloco

- Problema: convolução de um sinal $h[n]$ de tamanho finito com um sinal $x[n]$ de tamanho indefinido, infinito ou muito grande;
  - Atraso;
  - Processamento computacional.

- Solução: Convolução em bloco;
  - *Overlap and add*;
  - *Overlap and save*;

Convolução em Bloco

h[*n*] com tamanho *M*: $h[n]=0 \begin{cases} n > M-1 \\ n < 0 \end{cases}$

x[*n*] dividido em blocos de *L* amostras, com: $L \gg M$

- *Overlap and add*: Dividir *x*[*n*] em blocos de *L* amostras e realizar a DFT em *N* = *L* + *M* – 1 pontos (evitando *aliasing*)

$$x_0[n]=\{\overbrace{\underbrace{x[0],x[1],\cdots,x[L-1]}_{L\ \text{amostras}},\underbrace{0,0,\cdots,0}_{M-1\ \text{zeros}}}^{L+M-1\ \text{amostras}}\} \qquad X_0[k]=\mathrm{DFT}\{x_0[n]\}$$

$$x_1[n]=\{x[L],x[L+1],\cdots,x[2L-1],\underbrace{0,0,\cdots,0}_{M-1\ \text{zeros}}\} \qquad X_1[k]=\mathrm{DFT}\{x_1[n]\}$$

$$x_2[n]=\{x[2L],x[2L+1],\cdots,x[3L-1],\underbrace{0,0,\cdots,0}_{M-1\ \text{zeros}}\} \qquad X_2[k]=\mathrm{DFT}\{x_2[n]\}$$

Convolução em Bloco

- *Overlap and add*:

$$h[n] = \{\overbrace{\underbrace{h[0], h[1], \cdots, h[M-1]}_{M \text{ amostras}}, \underbrace{0, 0, \cdots, 0}_{L-1 \text{ zeros}}}^{L+M-1 \text{ amostras}}\}$$

$$H[k] = \text{DFT}\{h[n]\}$$

Realizar a filtragem de cada um dos blocos:

$$Y_0[k] = X_0[k]H[k] \longrightarrow \text{Tamanho } L + M - 1$$

$$Y_1[k] = X_1[k]H[k]$$

$$\vdots$$

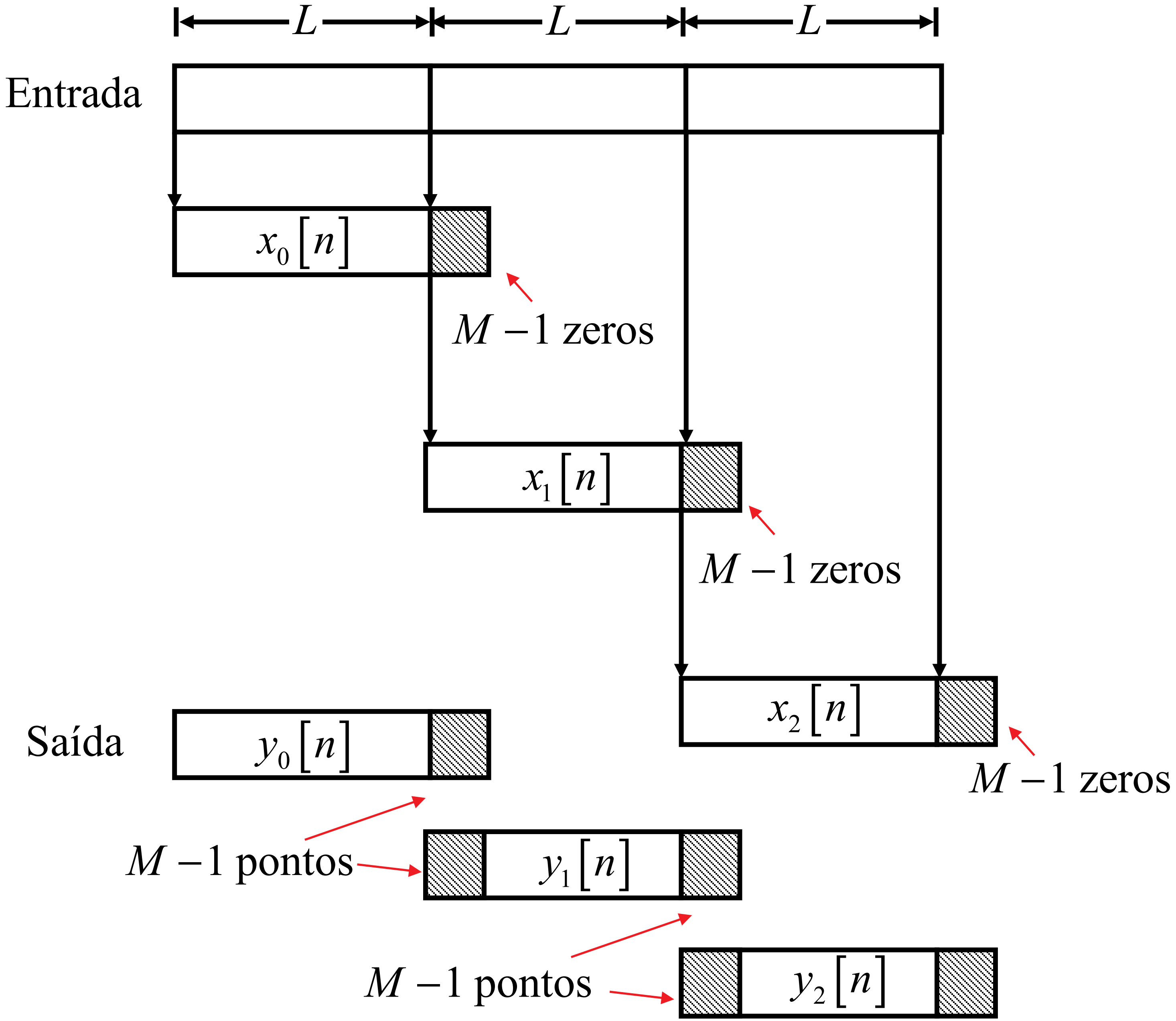
$$Y_m[k] = X_m[k]H[k]$$

$$y_m[n] = \text{IDFT}\{Y_m[k]\} \longrightarrow \text{Tamanho } L + M - 1$$

Convolução em Bloco

- *Overlap and add*:
  - Como os blocos filtrados possuem tamanho $L + M - 1$ e os blocos originais da entrada tem tamanho L, então as $M - 1$ ultimas amostras de um bloco se sobrepõem com as $M - 1$ primeiras amostras do bloco seguinte.
  - A saída é obtida realizando-se a soma, considerando as sobreposições:

$$y[n] = \begin{Bmatrix} y_0[0], y_0[1], \cdots, y_0[L-1], y_0[L] + y_1[0], y_0[L+1] + y_1[1], \\ \cdots, y_0[N-1] + y_1[M-1], y_1[M], \cdots \end{Bmatrix}$$



L
L
L
Entrada
$x_0[n]$
$M-1$ zeros
$x_1[n]$
$M-1$ zeros
$x_2[n]$
$M-1$ zeros
Saída
$y_0[n]$
$M-1$ pontos
$y_1[n]$
$M-1$ pontos
$y_2[n]$

Cálculo Computacional da DFT

- Implementação direta;
- Algoritmo de Goertzel;
- Transformada Rápida de Fourier;

Cálculo Computacional da DFT

Implementação Direta da DFT:

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{kn} \quad 0 \le k \le N-1$$

$$W_N^{kn} = e^{-j\frac{2\pi}{N}nk} = \cos\left(2\pi k \frac{n}{N}\right) - j\sin\left(2\pi k \frac{n}{N}\right)$$

$$X[k] = \sum_{n=0}^{N-1} x[n] \left\{ \cos\left(2\pi k \frac{n}{N}\right) - j\sin\left(2\pi k \frac{n}{N}\right) \right\}$$

$$= \sum_{n=0}^{N-1} \left\{ x_r[n] + jx_i[n] \right\} \left\{ \cos\left(2\pi k \frac{n}{N}\right) - j\sin\left(2\pi k \frac{n}{N}\right) \right\}$$

$$= \sum_{n=0}^{N-1} x_r[n]\cos\left(2\pi k \frac{n}{N}\right) + x_i[n]\sin\left(2\pi k \frac{n}{N}\right) - j\sum_{n=0}^{N-1} x_r[n]\sin\left(2\pi k \frac{n}{N}\right) - x_i[n]\cos\left(2\pi k \frac{n}{N}\right)$$

Cálculo Computacional da DFT

Implementação Direta da DFT: $x[n]$ real,

$$X[k] = \underbrace{\sum_{n=0}^{N-1} x[n]\cos\left(2\pi k \frac{n}{N}\right)}_{X_R[k]} - j\underbrace{\sum_{n=0}^{N-1} x[n]\sin\left(2\pi k \frac{n}{N}\right)}_{X_I[k]}$$

$\cos\left(2\pi k \frac{n}{N}\right), \sin\left(2\pi k \frac{n}{N}\right)$ podem ser armazenados em uma LUT ($2N^2$ elementos).

- Problemas:
  - Alta complexidade computacional;
  - Necessidade de armazenar $2N^2$ elementos.

Cálculo Computacional da DFT

```
algoritmo DFT_direta (x, N)
    para k de 0 até N-1 faça
        temp_R ← 0;
        temp_I ← 0;
        para n de 0 até N-1 faça
                temp_R ← temp_R + x[n]*cos(2*pi*k*n/N)
                temp_I ← temp_I + x[n]*sin(2*pi*k*n/N)
        fim-para
        X[k] = temp_R - j*temp_I;
    fim-para
    retorne X
fim-algoritmo
```

Cálculo Computacional da DFT

Algoritmo de Goertzel:

- Diminui o requisito de memória para a implementação da DFT;
- Requer mais operações matemáticas;

$$W_N^{-kN} = e^{-j\frac{2\pi}{N}(-N)k} = e^{j2\pi k} = 1$$

$$X[k] = \sum_{r=0}^{N-1} x[r]W_N^{kr} = 1 \times \left(\sum_{n=0}^{N-1} x[r]W_N^{kr} \right) = W_N^{-kN} \times \left(\sum_{r=0}^{N-1} x[r]W_N^{kr} \right) = \sum_{r=0}^{N-1} x[r]W_N^{kr}W_N^{-kN}$$

$$= \sum_{r=0}^{N-1} x[r]W_N^{-k(N-r)}$$

Definindo um sistema LIT com resposta ao impulso tal que: $h_k[n] = W_N^{-kn}u[n]$

$$y_k[n] = x[n] * h_k[n] = \sum_{r=-\infty}^{\infty} x[r]h_k[n-r] = \sum_{r=-\infty}^{\infty} x[r]W_N^{-k(n-r)}u[n-r]$$

Cálculo Computacional da DFT

Algoritmo de Goertzel:

$$y_k[n] = \sum_{r=-\infty}^{\infty} x[r] W_N^{-k(n-r)} u[n-r] \qquad x[r]=0 \quad r<0$$

$$= \sum_{r=0}^{N-1} x[r] W_N^{-k(n-r)} \qquad x[r]=0 \quad r>N-1$$

$$y_k[N] = \sum_{r=0}^{N-1} x[r] W_N^{-k(N-r)}$$

mas: $X[k] = \sum_{r=0}^{N-1} x[r] W_N^{-k(N-r)}$

logo: $X[k] = y_k[N]$

$$h[n] = W_N^{-kn} u[n]$$

$$H(z) = \sum_{n=-\infty}^{\infty} h[n] z^{-n} = \sum_{n=0}^{\infty} W_N^{-kn} z^{-n} = \frac{1}{1 - W_N^{-k} z^{-1}} = \frac{Y_k(z)}{X(z)}$$

$$Y_k(z)\left\{1 - W_N^{-k} z^{-1}\right\} = X(z)$$

Cálculo Computacional da DFT

Algoritmo de Goertzel:

$$Y_k(z) - W_N^{-k} z^{-1} Y_k(z) = X(z)$$
$$y_k[n] - W_N^{-k} y_k[n-1] = x[n]$$

Em resumo,

$$\begin{cases} y_k[n] = x[n] + W_N^{-k} y_k[n-1] \\ X[k] = y_k[N] \end{cases}$$

→ O cálculo é feito recursivamente, até a amostra $n = N$
→ Quando $n = N$, então $x[N] = 0$;

Cálculo Computacional da DFT

```
algoritmo Goertzel(x,N,k)
    yk_m1 ← 0
    para n de 0 até N-1 faça
        yk ← x[n] + exp(j*2*pi/N*k)*yk_m1
        yk_m1 ← yk
    fim-para
    yk ← exp(j*2*pi/N*k)*yk_m1
    retorne yk
fim-algoritmo
```

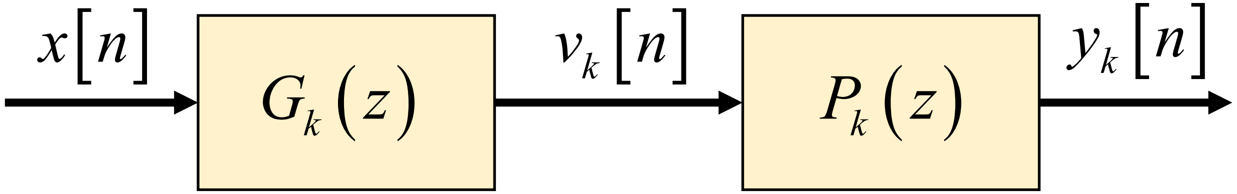
Cálculo Computacional da DFT

Algoritmo de Goertzel:

$$\frac{Y_k(z)}{X(z)} = \frac{1}{1-W_N^{-k}z^{-1}} \times \frac{1-W_N^k z^{-1}}{1-W_N^k z^{-1}} = \frac{1-W_N^k z^{-1}}{\left(1-W_N^{-k}z^{-1}\right)\left(1-W_N^k z^{-1}\right)} = \frac{1-W_N^k z^{-1}}{1-2\cos\left(\frac{2\pi k}{N}\right)z^{-1}+z^{-2}}$$

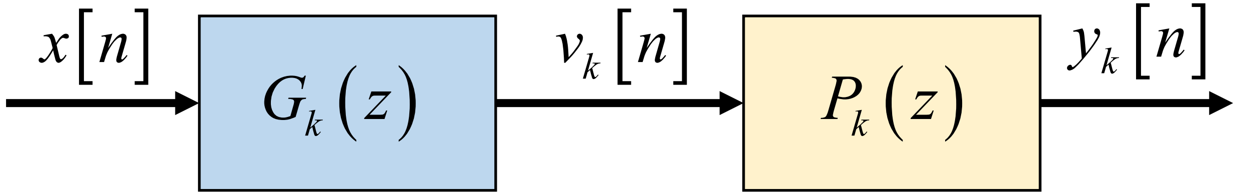
$$= G_k(z)P_k(z)$$

$$\begin{cases} G_k(z) = \dfrac{1}{1-2\cos\left(\dfrac{2\pi k}{N}\right)z^{-1}+z^{-2}} \\ P_k(z) = 1-W_N^k z^{-1} \end{cases}$$



Cálculo Computacional da DFT

Algoritmo de Goertzel:



$$G_k(z) = \frac{V_k(z)}{X(z)} = \frac{1}{1 - 2\cos\left(\frac{2\pi k}{N}\right)z^{-1} + z^{-2}}$$

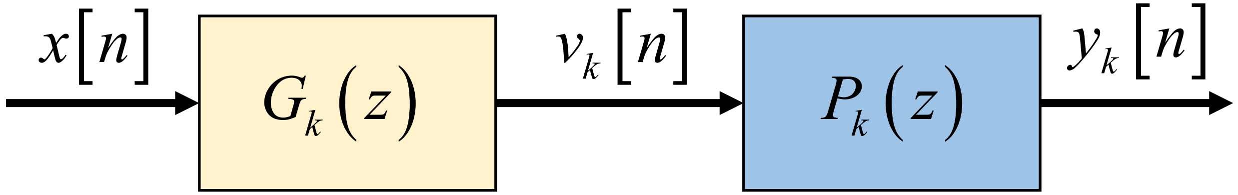
$$V_k(z)\left\{1 - 2\cos\left(\frac{2\pi k}{N}\right)z^{-1} + z^{-2}\right\} = X(z)$$

$$v_k[n] - 2\cos\left(\frac{2\pi k}{N}\right)v_k[n-1] + v_k[n-2] = x[n]$$

$$v_k[n] = x[n] + 2\cos\left(\frac{2\pi k}{N}\right)v_k[n-1] - v_k[n-2]$$

Cálculo Computacional da DFT

Algoritmo de Goertzel:



$$P_k(z) = \frac{Y_k(z)}{V_k(z)} = 1 - W_N^k z^{-1}$$

$$V_k(z)\left\{1 - W_N^k z^{-1}\right\} = Y_k(z)$$

$$y[n] = v_k[n] - W_N^k v_k[n-1]$$

Em resumo:

$$v_k[n] = x[n] + 2\cos\left(\frac{2\pi k}{N}\right) v_k[n-1] - v_k[n-2]$$

$$y[n] = v_k[n] - W_N^k v_k[n-1]$$

$$X[k] = y_k[N]$$

$$= v_k[N] - W_N^k v_k[N-1]$$

y[*n*] só precisa ser calculado para *n* = *N*

Cálculo Computacional da DFT

```
algoritmo Goertzel2(x,N,k)
    vk_m1 ← 0
    vk_m2 ← 0
    Const ← 2*cos(2*pi*k/N)
    Wn ← exp(-j*2*pi/N*k)
    para n de 0 até N-1 faça
        vk ← x[n] + Const*vk_m1 – vk_m2;
        vk_m2 ← vk_m1
        vk_m1 ← vk
    fim-para
    vk ← Const*vk_m1 – vk_m2
    Xk ← vk – Wn*vk_m1
    retorne Xk
fim-algoritmo
```

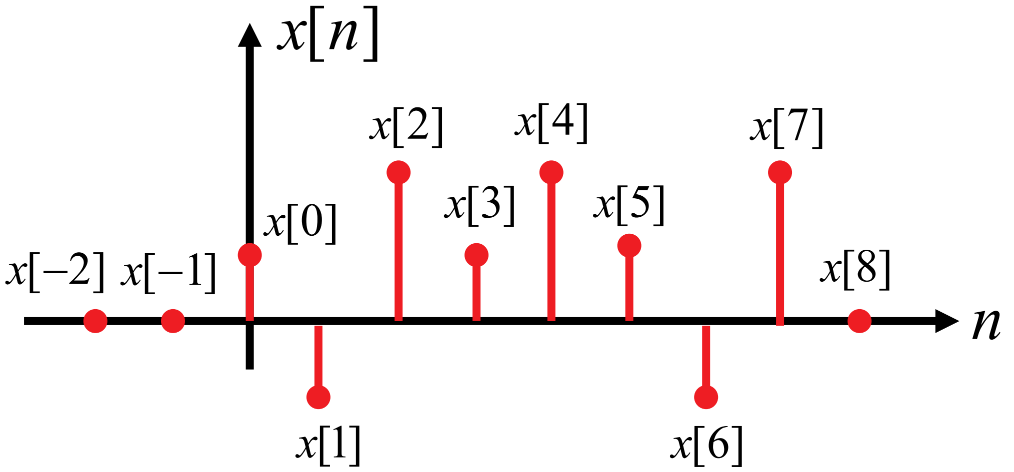
Cálculo Computacional da DFT

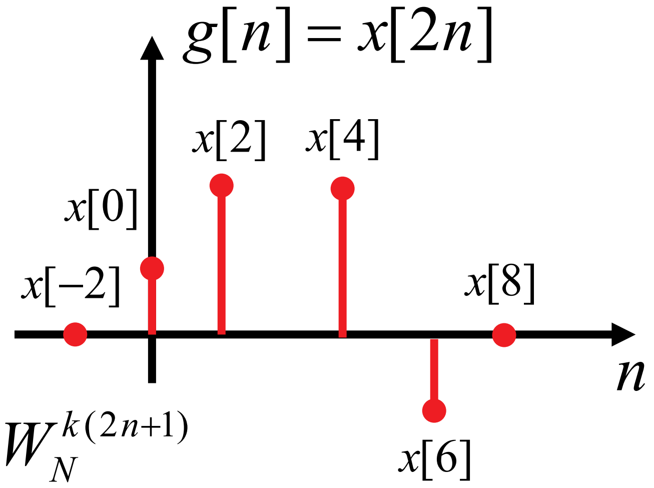
Transformada Rápida de Fourier (FFT):

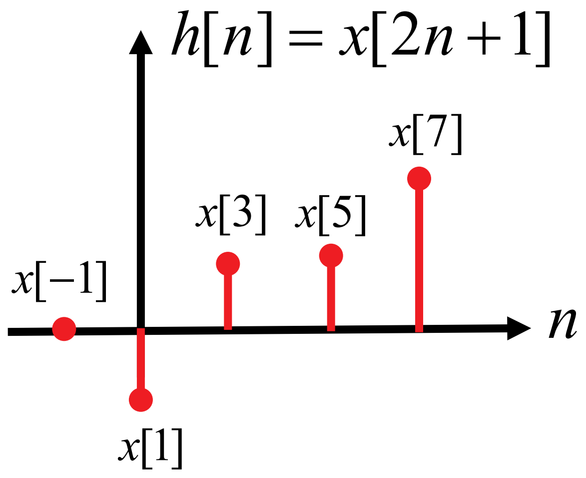
$$X[k]=\sum_{n=0}^{N-1}x[n]W_N^{kn} \qquad 0\le k\le N-1$$

$$g[n]=x[2n] \qquad n=0,1,\ldots,N/2-1$$
$$h[n]=x[2n+1]$$

$$X[k]=\sum_{n=0}^{N-1}x[n]W_N^{kn}$$
$$=\sum_{n\,\text{par}}x[n]W_N^{kn}+\sum_{n\,\text{ímpar}}x[n]W_N^{kn}$$
$$=\sum_{n=0}^{N/2-1}x[2n]W_N^{k(2n)}+\sum_{n=0}^{N/2-1}x[2n+1]W_N^{k(2n+1)}$$
$$=\sum_{n=0}^{N/2-1}g[n]W_N^{k(2n)}+\sum_{n=0}^{N/2-1}h[n]W_N^{k(2n+1)}$$







Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):

$$X[k] = \sum_{n=0}^{N/2-1} g[n] W_N^{k(2n)} + \sum_{n=0}^{N/2-1} h[n] W_N^{k(2n)} W_N^k$$

$$= \sum_{n=0}^{N/2-1} g[n] W_N^{k(2n)} + W_N^k \sum_{n=0}^{N/2-1} h[n] W_N^{k(2n)}$$

$$= \underbrace{\sum_{n=0}^{N/2-1} g[n] W_{N/2}^{nk}}_{G[k]} + W_N^k \underbrace{\sum_{n=0}^{N/2-1} h[n] W_{N/2}^{nk}}_{H[k]}$$

$$W_N^{k(2n)} = e^{-j(2\pi/N)(2n)k} = e^{-j(4\pi/N)nk} = e^{-j[2\pi/(N/2)]nk} = W_{N/2}^{nk}$$

$$= G[k] + W_N^k H[k] \quad k = 0, 1, \cdots, N/2-1$$

$$\begin{cases} G[k] = \sum_{n=0}^{N/2-1} g[n] W_{N/2}^{nk} \\ H[k] = \sum_{n=0}^{N/2-1} h[n] W_{N/2}^{nk} \end{cases} \quad k = 0, 1, \cdots, N/2-1$$

Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):

$$G\left[k+\frac{N}{2}\right]=G[k] \qquad H\left[k+\frac{N}{2}\right]=H[k]$$

$$\begin{aligned} W_N^{k+N/2} &= W_N^k W_N^{N/2} \\ &= W_N^k e^{-j(2\pi/N)(N/2)} \\ &= W_N^k e^{-j\pi} = -W_N^k \end{aligned}$$

$$X[k]=G[k]+W_N^k H[k]$$

$$\begin{aligned} X[k+N/2] &= G[k+N/2]+W_N^{k+N/2}H[k+N/2] \\ &= G[k]+W_N^{k+N/2}H[k] \\ &= G[k]-W_N^k H[k] \qquad k=0,1,\cdots,N/2-1 \end{aligned}$$

Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):

$$X[k] = G[k] + W_N^k H[k] \qquad k = 0,1,\cdots,N/2-1$$

$$X\left[k + \frac{N}{2}\right] = G[k] - W_N^k H[k]$$

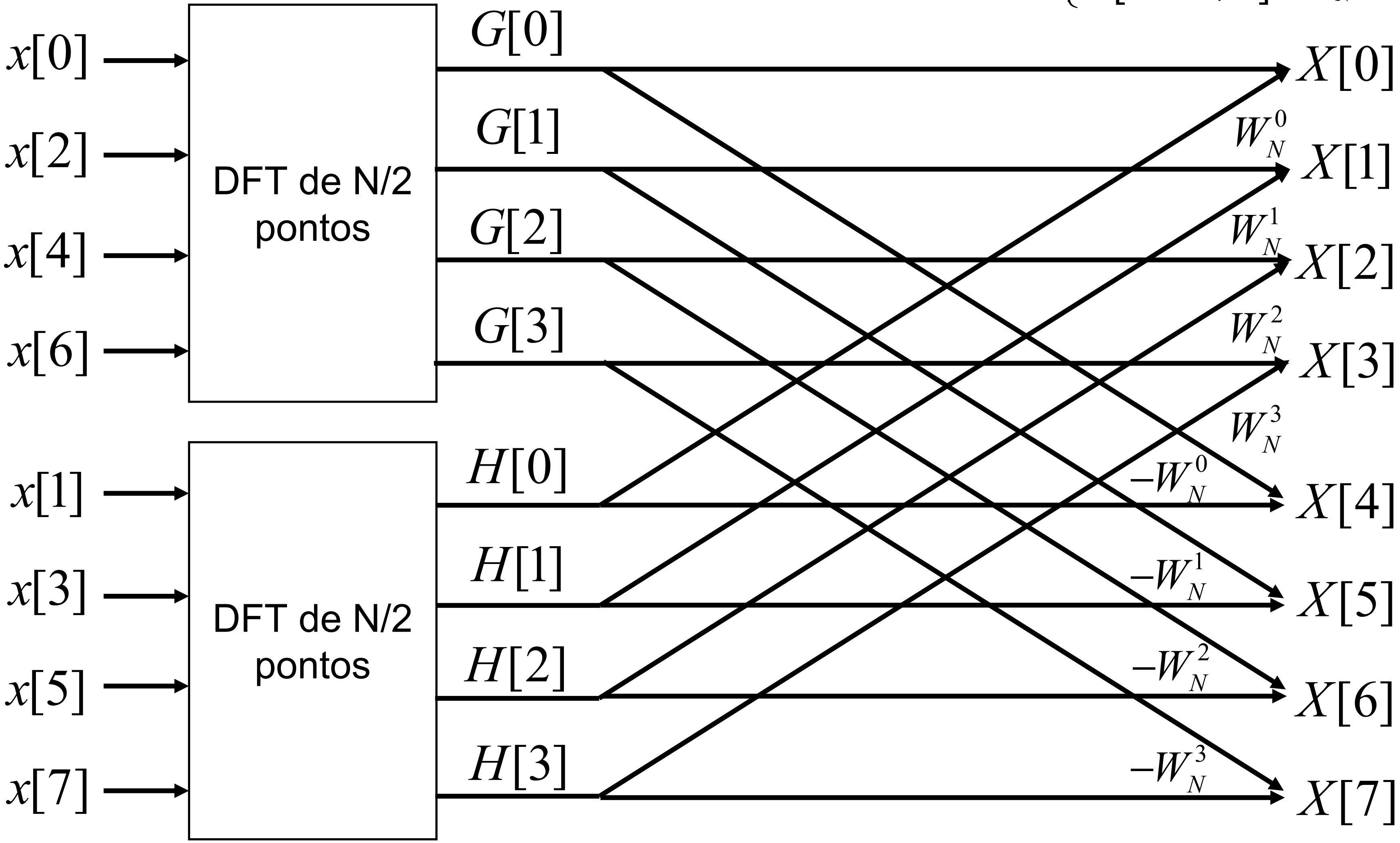
$$\Downarrow$$

$$X[k] = \begin{cases} G[k] + W_N^k H[k] & k = 0,\cdots,N/2-1 \\ G[k-N/2] - W_N^{(k-N/2)} H[k-N/2] & k = N/2,\cdots,N-1 \end{cases}$$

Cálculo Computacional da DFT

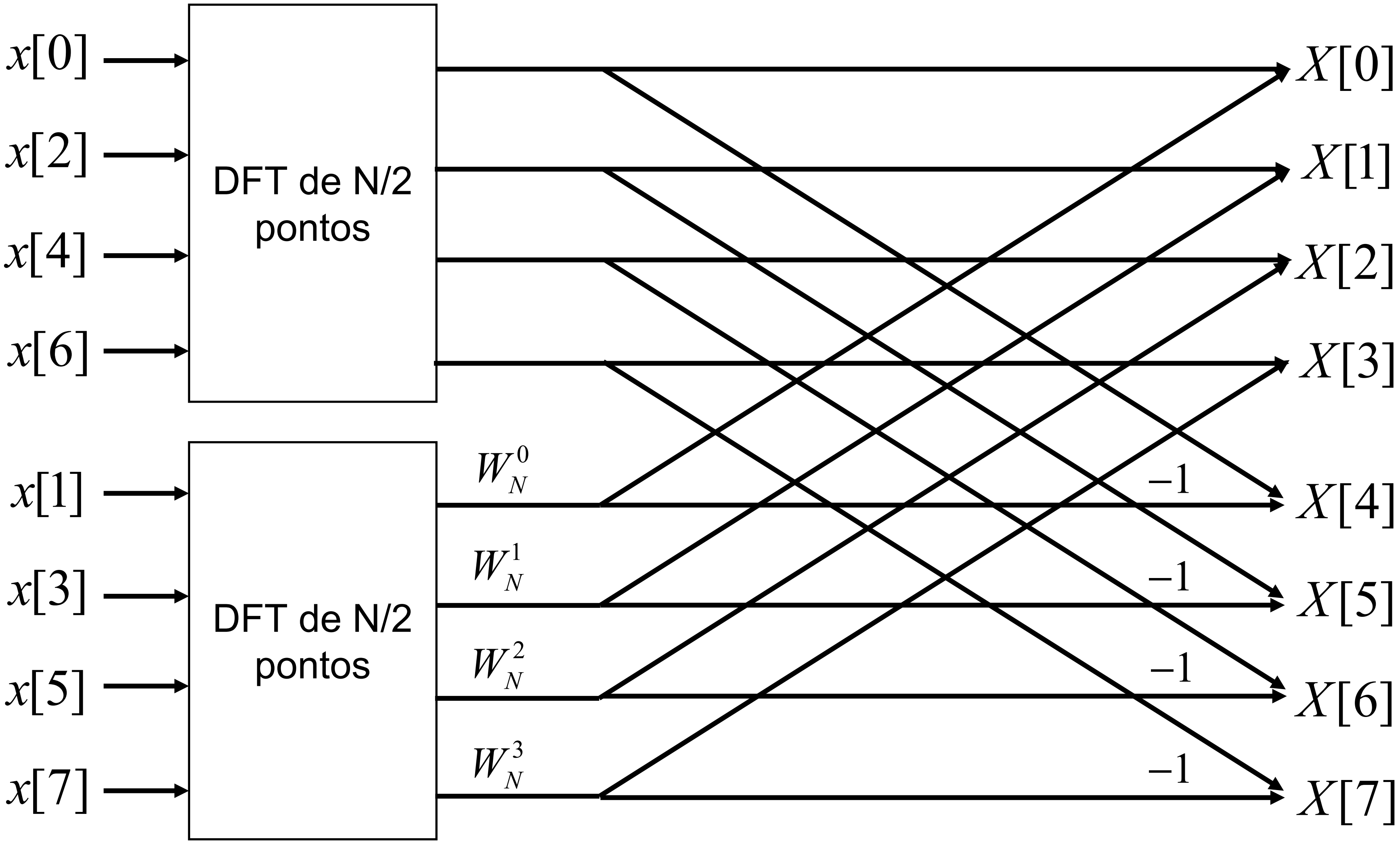
Transformada Rápida de Fourier (FFT):

$$X[k]=\begin{cases} G[k]+W_N^k H[k] & k=0,\cdots,N/2-1 \\ G[k-N/2]-W_N^{(k-N/2)}H[k-N/2] & k=N/2,\cdots,N-1 \end{cases}$$



Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):



Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):

$$G[k] = \sum_{n\,\text{par}} g[n] W_{N/2}^{nk} + \sum_{n\,\text{ímpar}} g[n] W_{N/2}^{nk}$$

$$= \sum_{n=0}^{N/4-1} g[2n] W_{N/2}^{2nk} + \sum_{n=0}^{N/4-1} g[2n+1] W_{N/2}^{(2n+1)k}$$

$$= \sum_{n=0}^{N/4-1} g[2n] W_{N/4}^{nk} + W_{N/2}^{k} \sum_{n=0}^{N/4-1} g[2n+1] W_{N/4}^{nk}$$

$$= \underbrace{\sum_{n=0}^{N/4-1} g[2n] W_{N/4}^{nk}}_{G'[k]} + W_{N}^{2k} \underbrace{\sum_{n=0}^{N/4-1} g[2n+1] W_{N/4}^{nk}}_{G''[k]}$$

$$= G'[k] + W_N^{2k} G''[k] \qquad k = 0, \cdots, N/4-1$$

$$W_{N/2}^{k} = e^{-j[2\pi/(N/2)]k}$$

$$= e^{-j[2\pi/N]2k} = W_N^{2k}$$

Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):

$$\begin{cases} G'[k+N/4] = G'[k] \\ G''[k+N/4] = G''[k] \\ W_N^{2(k+N/4)} = -W_N^{2k} \end{cases} \Rightarrow \begin{aligned} G[k] &= G'[k] + W_N^{2k} G''[k] \\ G[k+N/4] &= G'[k] - W_N^{2k} G''[k] \end{aligned} \qquad k = 0,\cdots,N/4-1$$

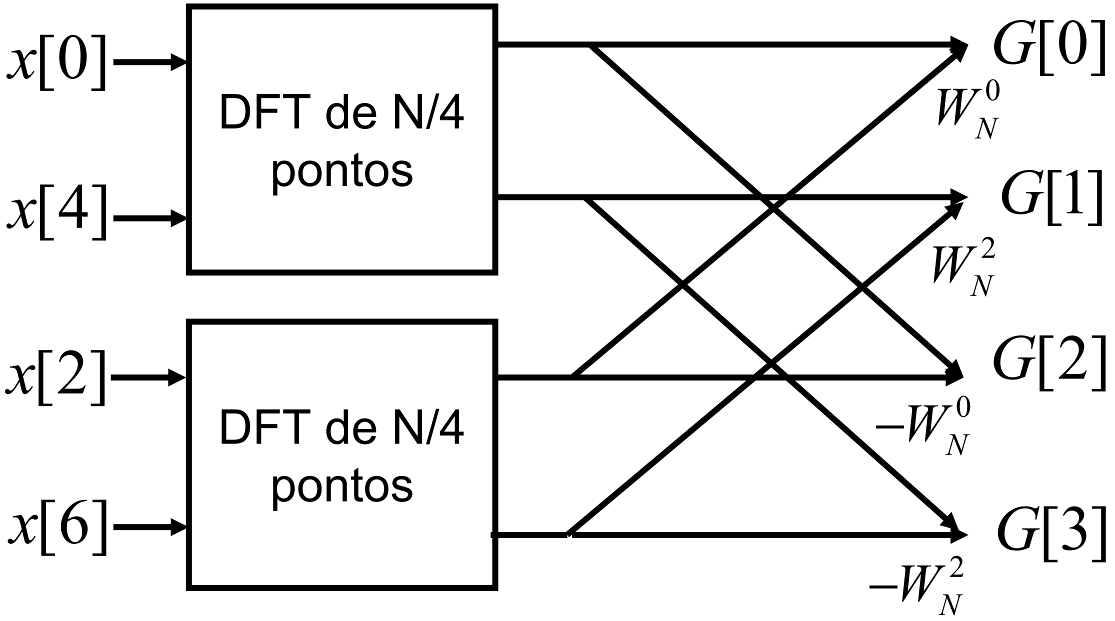
$$\Downarrow$$

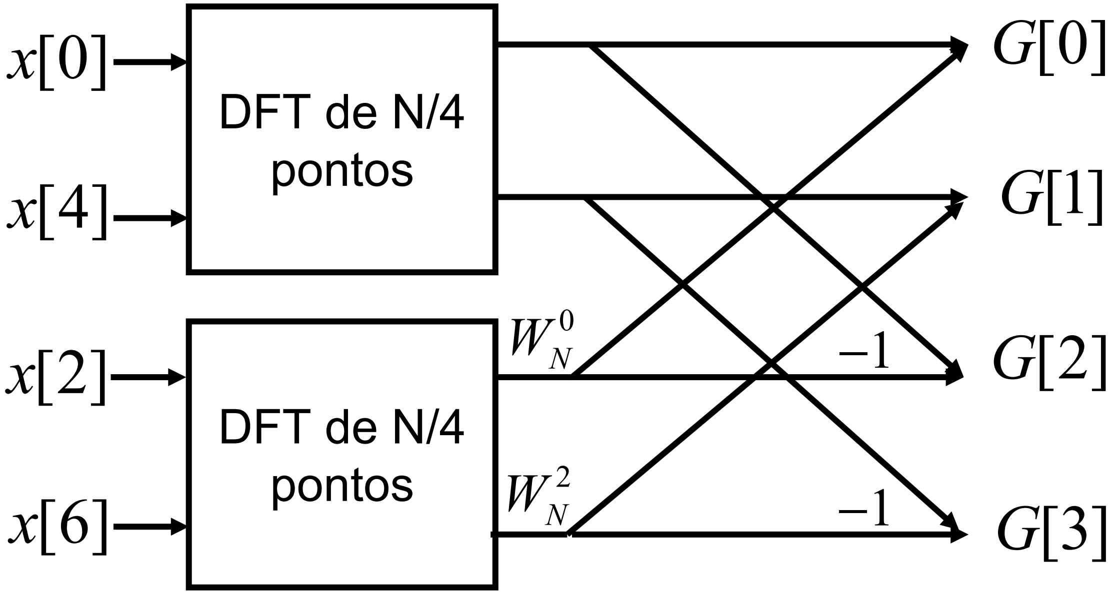
$$G[k] = \begin{cases} G'[k] + W_N^{2k} G''[k] & k = 0,\cdots,N/4-1 \\ G'[k-N/4] - W_N^{2(k-N/4)} G''[k-N/4] \end{cases}$$

Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):

$$G[k]=\begin{cases} G'[k]+W_N^{2k}G''[k] & k=0,\cdots,N/4-1 \\ G'[k-N/4]-W_N^{2(k-N/4)}G''[k-N/4] \end{cases}$$

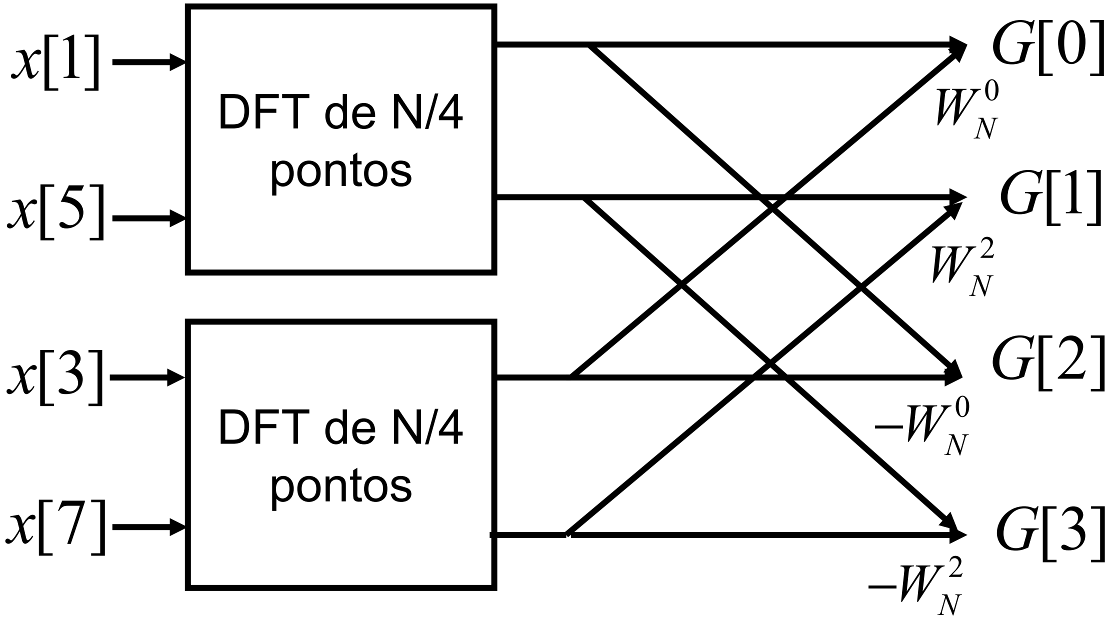


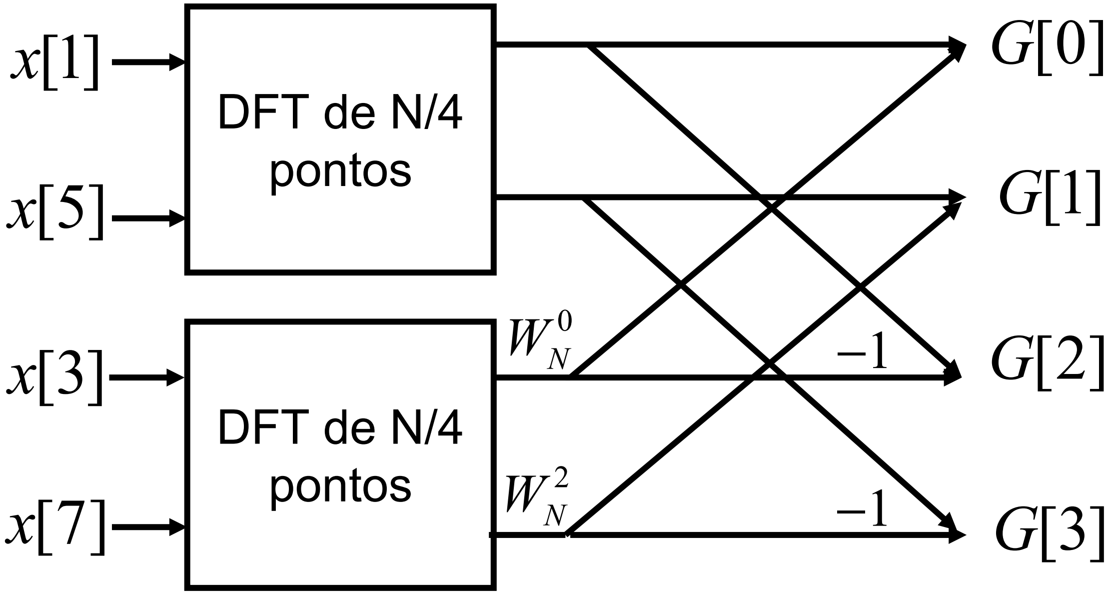


Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):

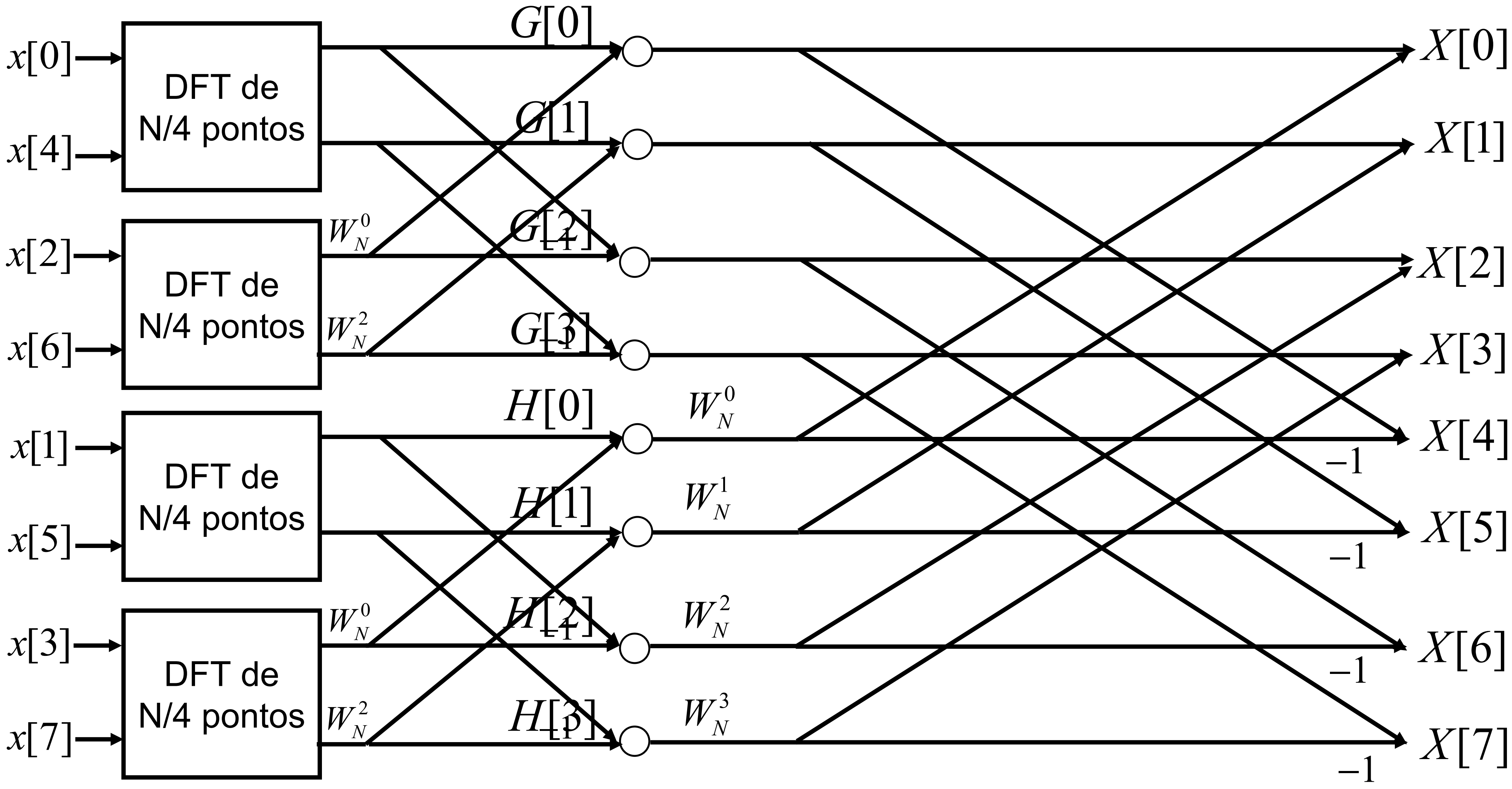
$$H[k] = \begin{cases} H'[k] + W_N^{2k} H''[k] \\ H'[k - N/4] - W_N^{2(k-N/4)} H''[k - N/4] \end{cases} \qquad k = 0, \cdots, N/4 - 1$$





Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):



Cálculo Computacional da DFT

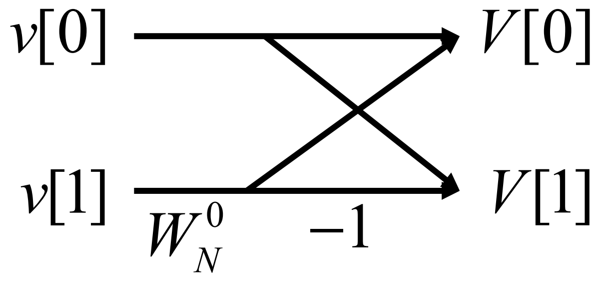
Transformada Rápida de Fourier (FFT):

- A decimação é feita até chegar em dois pontos;
- A DFT de dois pontos é:

$$V[k]=\sum_{n=0}^{1} v[n]W_2^{kn} = v[0]W_2^{k0}+v[1]W_2^{k}$$
$$= v[0]+v[1]W_2^{k}$$

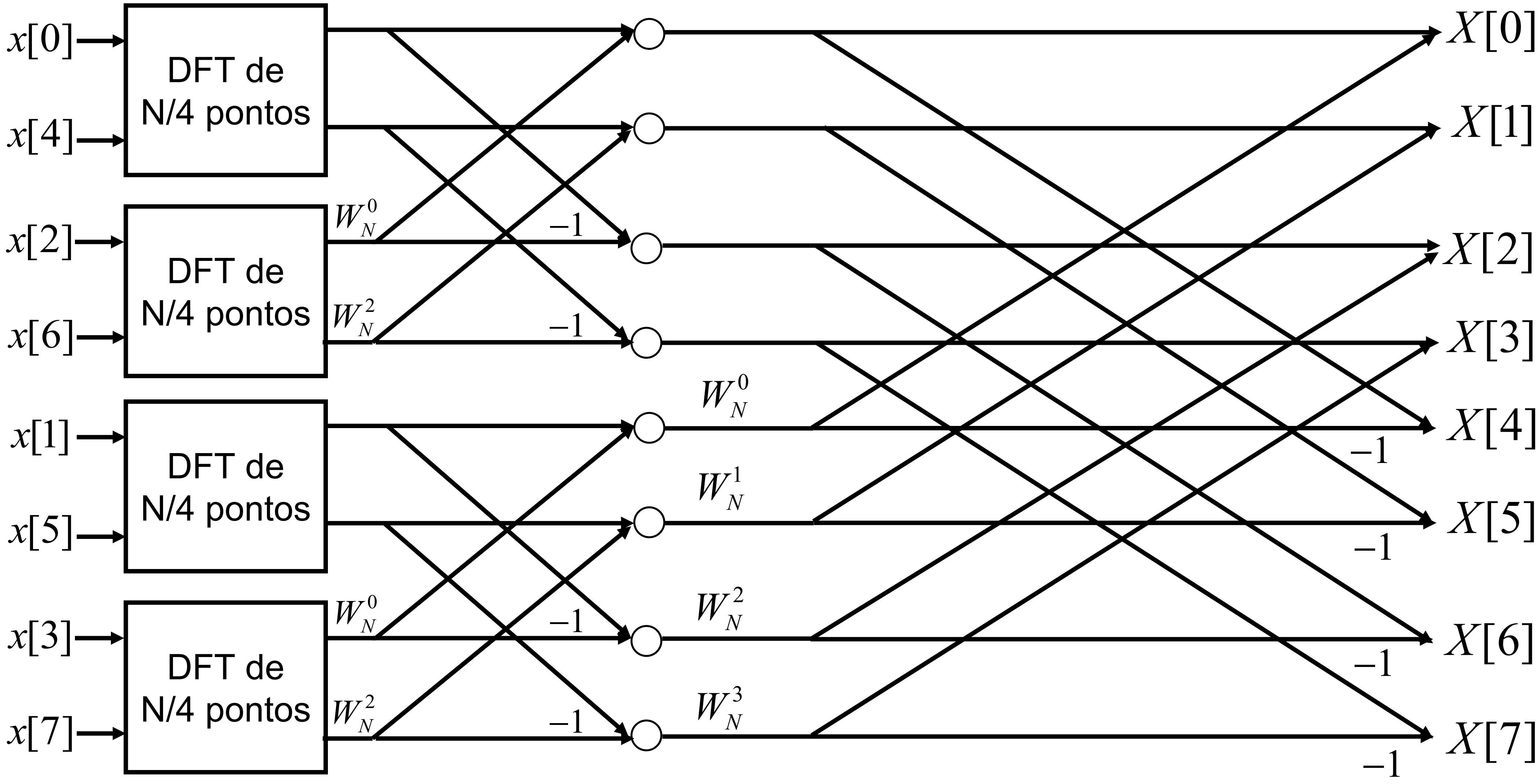
$$V[0]=v[0]+v[1]W_2^{0}=v[0]+v[1]$$
$$V[1]=v[0]+v[1]W_2^{1}=v[0]+v[1]e^{-j(2\pi/2)}$$
$$=v[0]+v[1]e^{-j\pi}=v[0]-v[1]$$

$$V[k]=\begin{cases} v[0]+v[1]W_N^{0} & k=0 \\ v[0]-v[1]W_N^{0} & k=1 \end{cases}$$



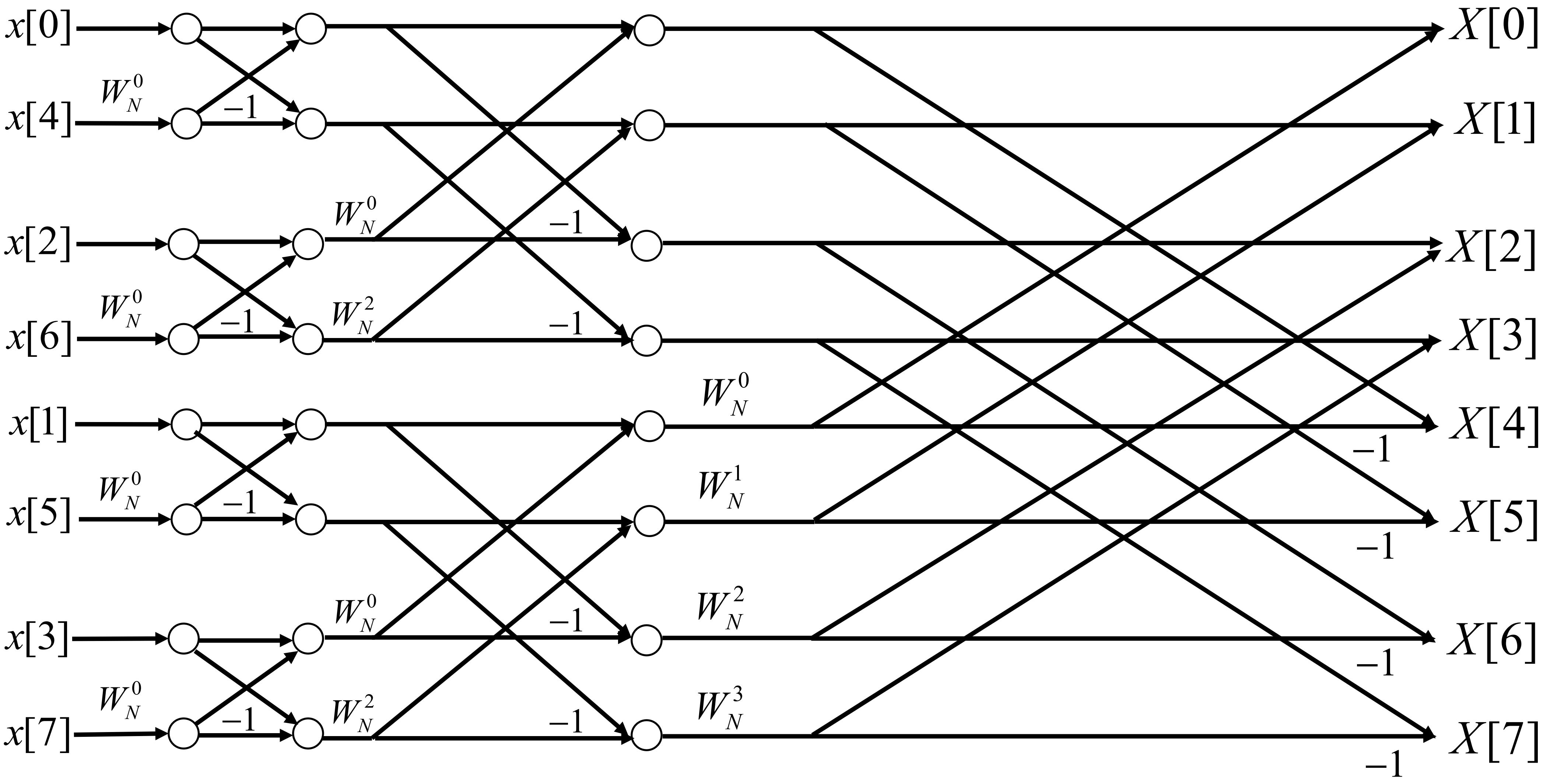
Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):



Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):



Cálculo Computacional da DFT

```
algoritmo fft_recursive(x,N)
    se N==1 então faça
        retorne x
    Wn ← exp(-j*2*pi/N);
    W ← 1;
    x_even ← x[0:2:N-1];
    Gk ← fft_recursive(x_even,N/2);
    x_odd ← x[1:2:N-1];
    Hk ← fft_recursive(x_odd,N/2);
    para k de 0 até N/2-1 faça
        Xk[k] ← Gk[k] + W*Hk[k];
        Xk[k+N/2] ← Gk[k] - W*Hk[k];
        W ← W*Wn;
    fim-para
    retorne Xk
fim-algoritmo
```

Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):

- Computação *In-Place*:

$X_s[k]$ *Coeficientes de entrada do m-ésimo estágio*

$$k = 0, 1, \ldots, N-1$$
$$s = 0, 1, \ldots, v$$
$$v = \log_2(N)$$

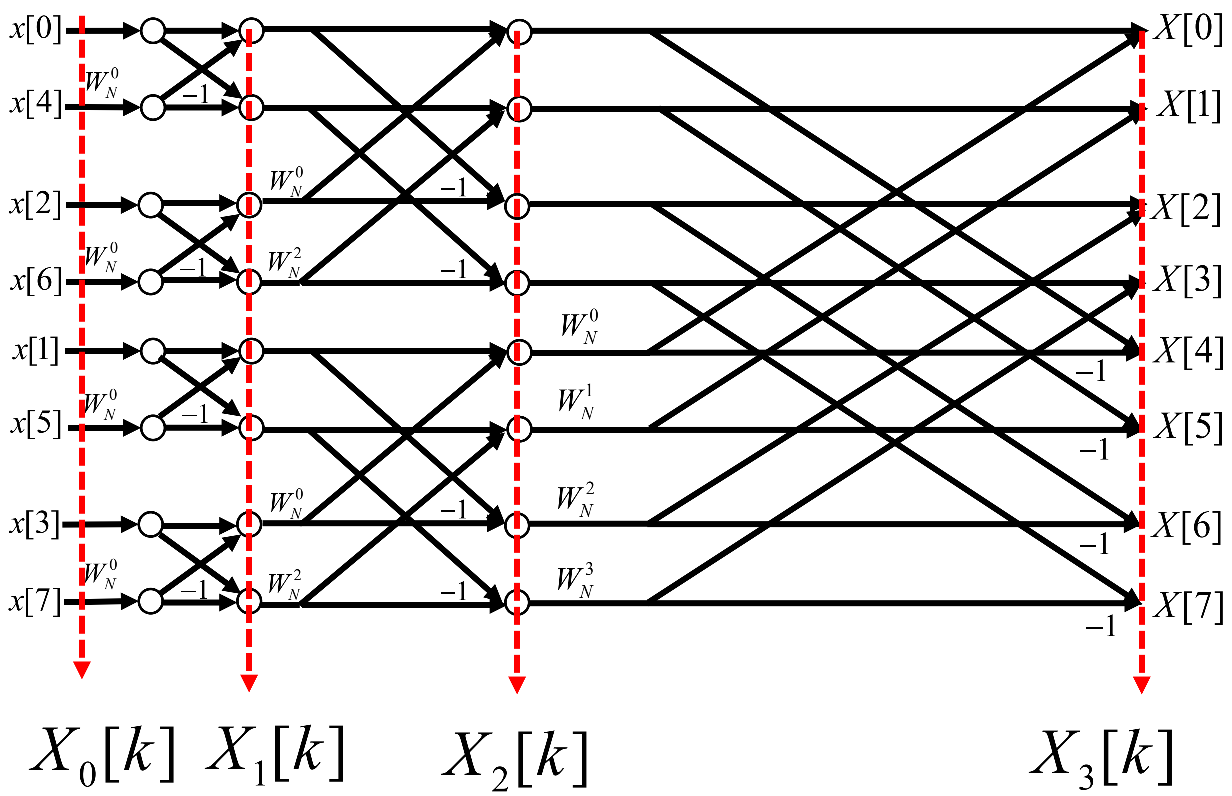
Estágio 0 → Entrada

$X_0[0] = x[0]$ $\quad$ $X_0[1] = x[4]$

$X_0[2] = x[2]$ $\quad$ $X_0[3] = x[6]$

$X_0[4] = x[1]$ $\quad$ $X_0[5] = x[5]$

$X_0[6] = x[3]$ $\quad$ $X_0[7] = x[7]$



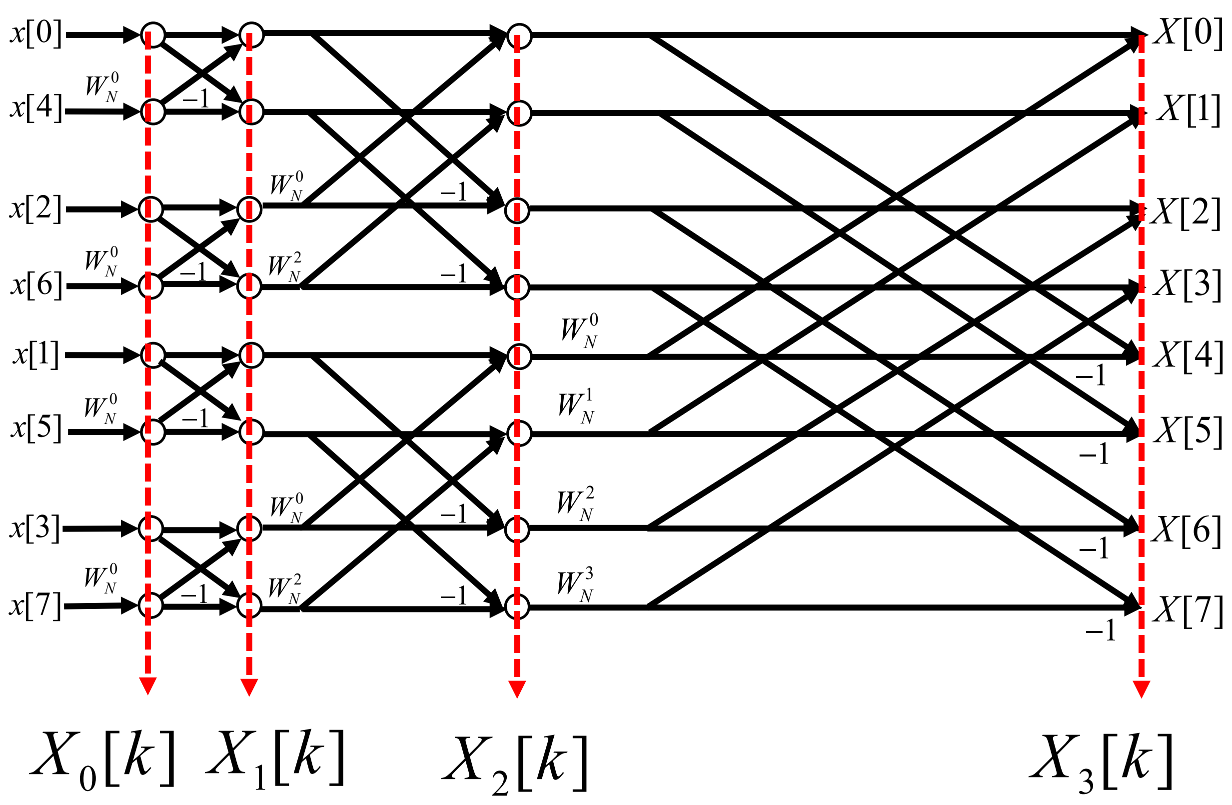
Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):

- Computação *In-Place*:

Estágio 0 → Entrada

$$X_0[0] = x[0] \qquad X_0[1] = x[4]$$
$$X_0[2] = x[2] \qquad X_0[3] = x[6]$$
$$X_0[4] = x[1] \qquad X_0[5] = x[5]$$
$$X_0[6] = x[3] \qquad X_0[7] = x[7]$$



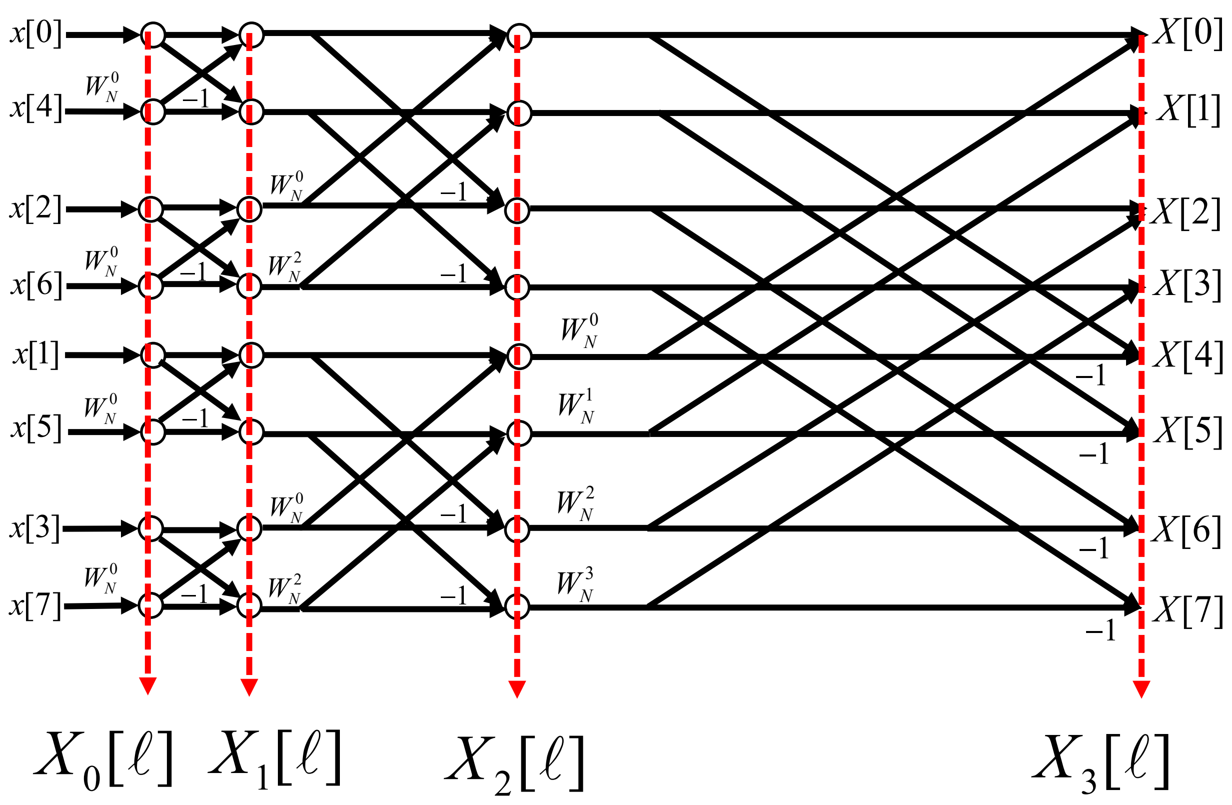
Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):

- Computação *In-Place*:

$$X_0[000] = x[000] \qquad X_0[001] = x[100]$$
$$X_0[010] = x[010] \qquad X_0[011] = x[110]$$
$$X_0[100] = x[001] \qquad X_0[101] = x[101]$$
$$X_0[110] = x[011] \qquad X_0[111] = x[111]$$

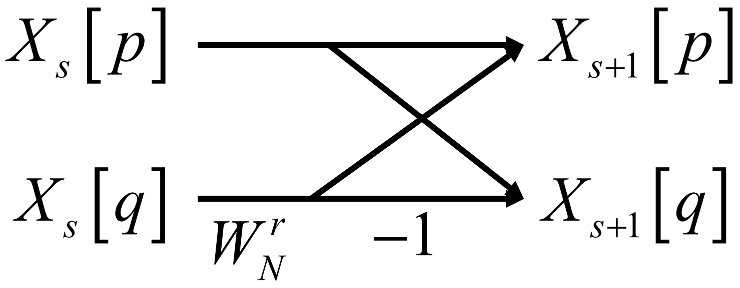
Operação bit-reverso



Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):

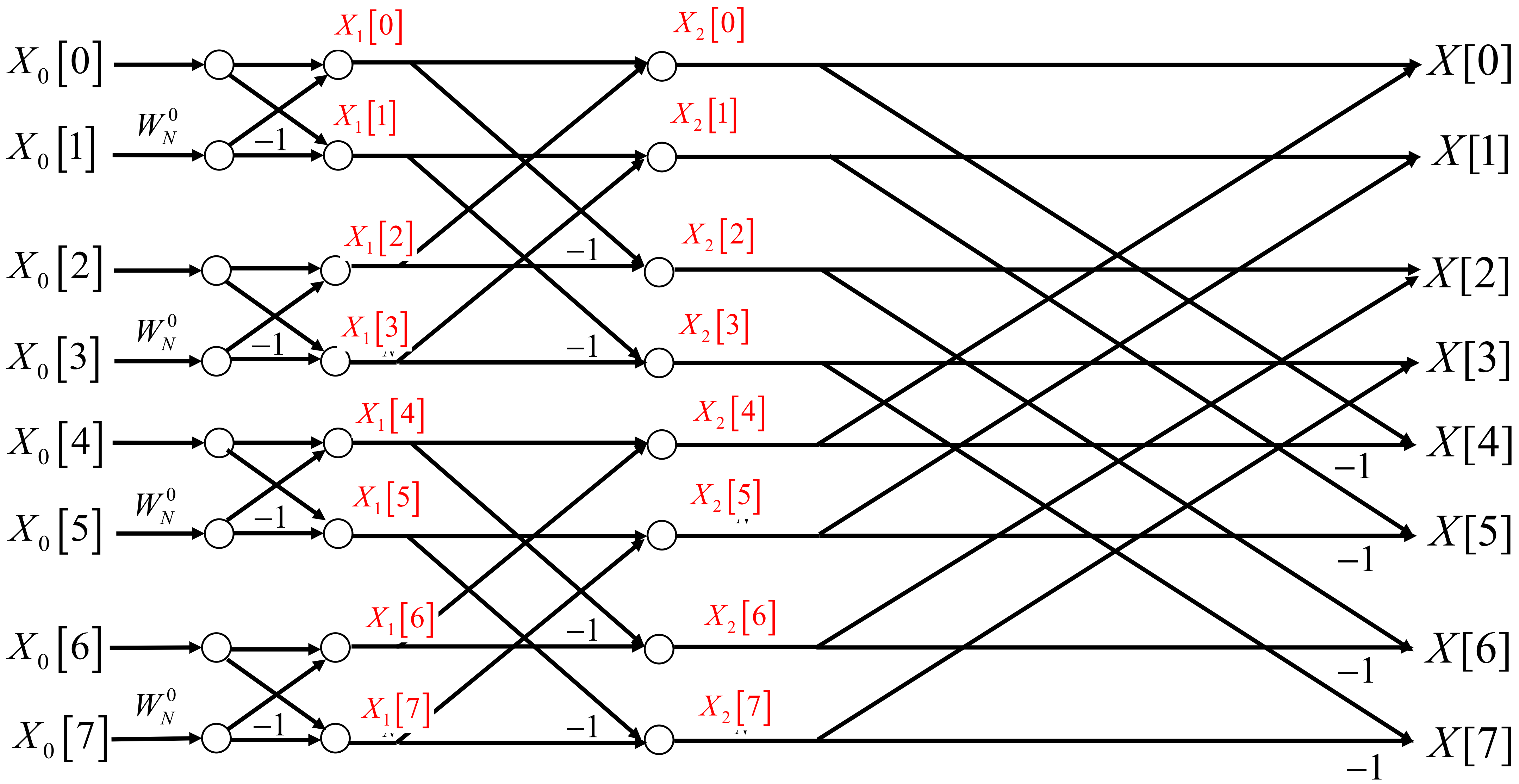
- Computação *In-Place*: Cada estágio pode ser entendido como:



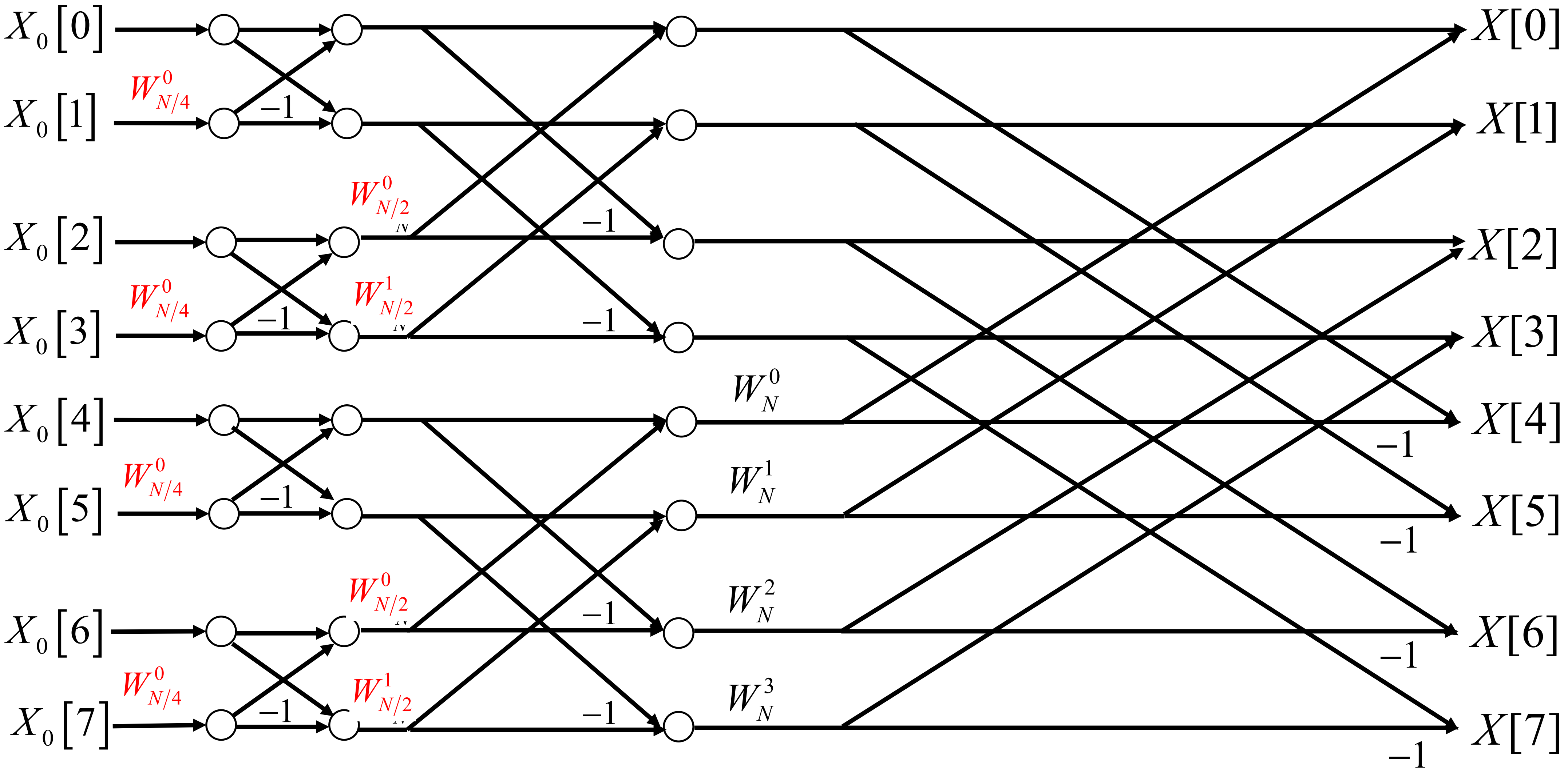
$$X_{s+1}[p] = X_s[p] + W_N^r X_s[q]$$

$$X_{s+1}[q] = X_s[p] - W_N^r X_s[q]$$

p, *q* e *r* mudam de acordo com o estágio


$X_0[0]$
$X_0[1]$
$X_0[2]$
$X_0[3]$
$X_0[4]$
$X_0[5]$
$X_0[6]$
$X_0[7]$
W_N^0
W_N^0
W_N^0
W_N^0
-1
$X_1[0]$
$X_1[1]$
$X_1[2]$
$X_1[3]$
$X_1[4]$
$X_1[5]$
$X_1[6]$
$X_1[7]$
$X_2[0]$
$X_2[1]$
$X_2[2]$
$X_2[3]$
$X_2[4]$
$X_2[5]$
$X_2[6]$
$X_2[7]$
$X[0]$
$X[1]$
$X[2]$
$X[3]$
$X[4]$
$X[5]$
$X[6]$
$X[7]$

$$W_{N/k}^{\ell} = W_N^{\ell k}$$



Cálculo Computacional da DFT

Transformada Rápida de Fourier (FFT):

- Computação *In-Place*: Logo,

$$X_s[k+\ell] = X_{s-1}[k+\ell] + W_m^{\ell} X_{s-1}[k+\ell+m]$$

$$X_s[k+\ell+m] = X_{s-1}[k+\ell] - W_m^{\ell} X_{s-1}[k+\ell+m]$$

$$m = 2^s$$

$$\ell = 0, 1, \cdots, m/2 - 1$$

$$k = 0, m, 2m, \cdots, N-1$$

Cálculo Computacional da DFT

```
algoritmo fft_interactive(x,N)
    X ← bit-reverso(x)
    v ← log2(N)
    para s de 1 até v faça
        m ← 2^v
        wm ← exp(-j*2*pi/m)
        para k de 0 até N-1 por m faça
                w ← 1;
                para l de 0 até m/2 - 1 faça
                        t ← w*X[k + l + m/2];
                        u ← X[k + l];
                        X[k + l] ← u + t;
                        X[k + l + m/2] ← u - t;
                        w ← w*wm;
                fim-para
        fim-para
    fim-para
    retorne X
fim-algoritmo
```

Cálculo Computacional da DFT

- **Transformada Rápida de Fourier (FFT):**
  - Formas alternativas:
    - Dizimação na frequência;
    - Entrada na ordem normal e saída na ordem *bit-reversa*;
    - Entrada na ordem normal e saída na ordem normal;
    - Algoritmos para valores compostos de *N* (não base 2);

Cálculo Computacional da DFT

Análise da complexidade computacional:

Transformada Discreta de Fourier → $O\left(N^2\right)$

Transformada Rápida de Fourier → $O\left(N \log_2 N\right)$