

① ③ Algoritmi fundamentali de tip FFT



Algoritmul lui Goertzel

Prima variantă de calcul a TFD

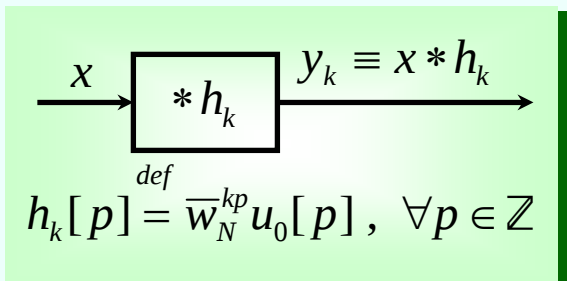
- Exprimare echivalentă a TFD

$$X[k] = \sum_{n=0}^{N-1} x[n] \bar{w}_N^{k(N-n)}, \quad \forall k \in \overline{0, N-1}$$

$$\begin{pmatrix} w_N^{kN} = 1 \\ \forall k \in \overline{0, N-1} \end{pmatrix}$$

periodicitate sumă de convoluție

Ieșirea la momentul N
a unui sistem liniar



$$(Supp\ x = \overline{0, N-1})$$

$$y_k[p] = \sum_{n \geq 0} x[n] h_k[p-n] = \sum_{n=0}^{N-1} x[n] \bar{w}_N^{k(p-n)}, \quad \forall p \in \overline{0, N-1}$$

$$X[k] = y_k[N], \quad \forall k \in \overline{0, N-1}$$

- Funcția de transfer a sistemului

$$H_k(z) \stackrel{\text{def}}{=} \sum_{p \geq 0} h_k[p] z^{-p} = \sum_{p \geq 0} (\bar{w}_N^k z^{-1})^p = \frac{1}{1 - \bar{w}_N^k z^{-1}}, \quad \forall k \in \overline{0, N-1}$$

$$|z| > 1$$

Teorema întârzierii

$$z^{-1} \mathcal{L}(f)(z) = \mathcal{L}(q^{-1} f)(z)$$

Ecuția recursivă a ieșirii

$$y_k[n] - \bar{w}_N^k y_k[n-1] = x[n], \quad \forall n \in \mathbb{Z}$$

Ecuția recursivă a TFD

$$\begin{cases} y_k[0] = x[0] \\ y_k[1] = \bar{w}_N^k y_k[0] + x[1] \\ \vdots \\ y_k[N] = \bar{w}_N^k y_k[N-1] + x[N] = \bar{w}_N^k y_k[N-1] = X[k] \end{cases}$$

$$\forall k \in \overline{0, N-1}$$

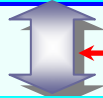
① ③ Algoritmi fundamentali de tip FFT

Algoritmul lui Goertzel

A doua variantă de calcul a TFD
(îmbunătățită)

- Exprimare echivalentă ecuației recursive anterioare

$$(1 - \bar{w}_N^k q^{-1}) y_k[n] = x[n], \quad \forall n \in \overline{0, N}, \quad \forall k \in \overline{0, N-1}$$



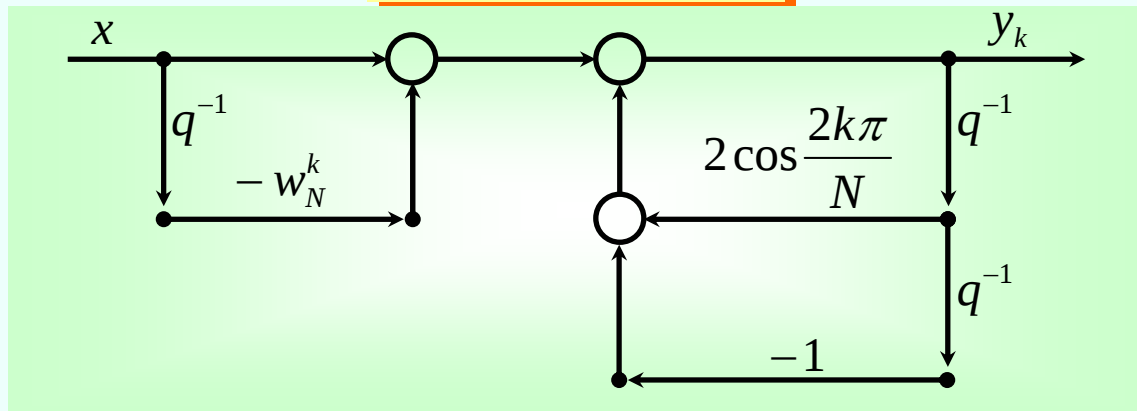
← înmulțire forțată cu $(1 - w_N^k q^{-1})$

$$\left(1 - 2 \cos \frac{2k\pi}{N} q^{-1} + q^{-2}\right) y_k[n] = (1 - w_N^k q^{-1}) x[n], \quad \forall n \in \overline{0, N}, \quad \forall k \in \overline{0, N-1}$$



$$y_k[n] = 2 y_k[n-1] \cos \frac{2k\pi}{N} - y_k[n-2] + x[n] - w_N^k x[n-1], \quad \forall n \in \overline{0, N}, \quad \forall k \in \overline{0, N-1}$$

Schema de calcul



Inițializare

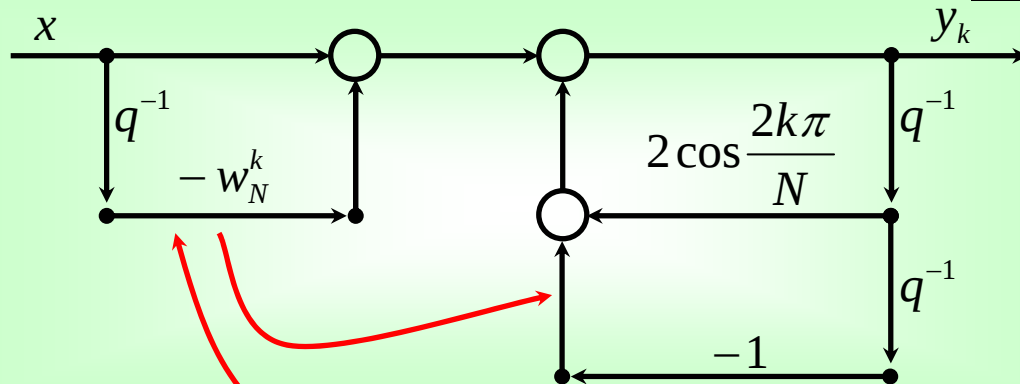
$$y_k[-1] = y_k[-2] = 0 \\ \forall k \in \overline{0, N-1}$$

① ③ Algoritmi fundamentali de tip FFT

Algoritmul lui Goertzel

Varianta eficientă de calcul a TFD

Schema anterioară de calcul se poate transforma echivalent, folosind **Teorema lui TELLEGEN**.



Algoritmul lui Goertzel

$$\begin{aligned} v_k[-2] &= v_k[-1] = 0 \\ &\vdots \\ v_k[n] &= 2 \cos \frac{2k\pi}{N} v_k[n-1] - v_k[n-2] + x[n] \\ &\vdots \\ v_k[N] &= 2 \cos \frac{2k\pi}{N} v_k[N-1] - v_k[N-2] \end{aligned}$$

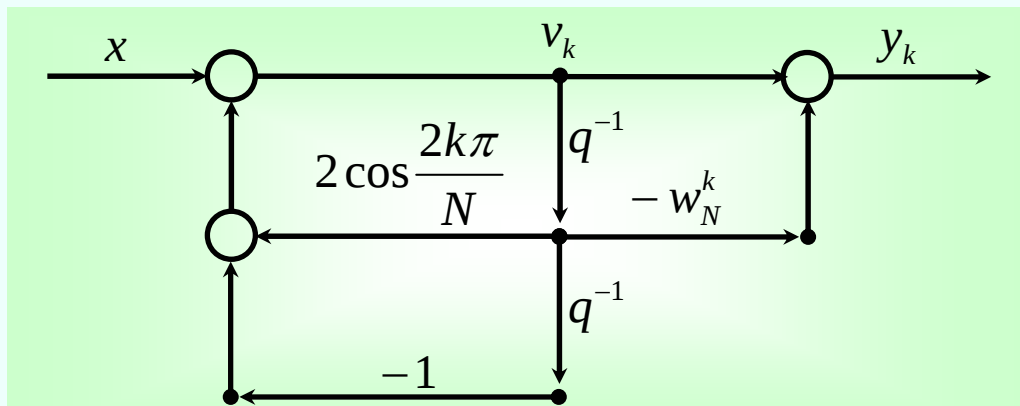
$$X[k] = y_k[N] = v_k[N] - w_N^k v_k[N-1]$$

$$\forall k \in \overline{0, N-1}$$

Număr de operații

$$\mathcal{O}_1[N] = \left[\left\lceil \frac{N-1}{2} \right\rceil (2N+5) \right]_{\bullet} + [4N(N+1)]_{+} \sim N^2$$

⚡ de 4 ori mai mic



① ③ Algoritmi fundamentali de tip FFT

Algoritmul fundamental bazat pe segmentarea în timp (FFT-t)

- Descriș tot în cadrul lucrărilor de laborator ale pachetului FFT.

Algoritmul FFT bazat pe segmentarea în timp

Principiul segmentării în timp

Fast Fourier Transform

$X[k] = \sum_{n=0}^{N-1} x[n]w_N^{kn}, \quad \forall k \in 0, N-1$

Segment cu ecuație de ordin par: $X[k] = \sum_{m=0}^{M-1} x[2m]w_{2M}^{2km} + \sum_{m=0}^{M-1} x[2m+1]w_{2M}^{2k(m+1)}$

Segment cu ecuație de ordin impar: $X[k] = \sum_{m=0}^{M-1} x[2m]w_{2M}^{2km} + w_{2M}^{2k} \sum_{m=0}^{M-1} x[2m+1]w_{2M}^{2km}$

Revenire calculată a TFD, pentru $N=2M$

În calculul TFD_{2M} se poate utiliza o pereche de TFD_M aplicate separatelor par și impar ale semnalului original.

Scheza de calcul

Segment par: $x[0], x[2], \dots, x[2M-2]$ → TFD_M → G[k]

Segment impar: $x[1], x[3], \dots, x[2M-1]$ → TFD_M → H[k]

Recombinația lui w_{2M}^{2k}

Recombinația lui w_{2M}^{2k}

$G[k] = \sum_{m=0}^{M-1} x[2m]w_{2M}^{2km}$

$H[k] = \sum_{m=0}^{M-1} x[2m+1]w_{2M}^{2km}$

$X[k] = G[k] + w_{2M}^{2k} H[k]$

$X[k+M] = G[k] - w_{2M}^{2k} H[k]$

$\forall k \in 0, N-1$

Algoritmul FFT bazat pe segmentarea în timp

Principiul segmentării în timp

Dacă $N=8M$, semnalul poate fi partajat în 4 segmente (fiecărui segment dă o pereche unică dintr-o hupă de 8-a sau nouă pereche de segmente, mai scurte). Rezultă că TFD_{8M} poate fi evaluat cu ajutorul a 4 TFD_{2M}.

În general, dacă $N=2^L$, semnalul poate fi partajat în 2^{L-1} segmente (fiecărui segment are doar 2 ecuații). Rezultă că TFD_{2^L} poate fi evaluat cu ajutorul a $(L-1)$ TFD₂.

Exemplu

$N=8=2^3$

Scheza de calcul

Segment 1-2: $x[0], x[4]$

Segment 3-4: $x[2], x[6]$

Segment 5-6: $x[1], x[5]$

Segment 7-8: $x[3], x[7]$

$X[0] = x[0] + x[4]$

$X[1] = x[0] - x[4]$

$X[2] = x[2] + x[6]$

$X[3] = x[2] - x[6]$

$X[4] = x[1] + x[5]$

$X[5] = x[1] - x[5]$

$X[6] = x[3] + x[7]$

$X[7] = x[3] - x[7]$

3 trepte de calcul paralel

Algoritmul FFT bazat pe segmentarea în timp

Prima variantă de calcul

În general, dacă $N=2^L$, valoarea TFD_{2^L} se poate obține cu ajutorul unei scheme de calcul astfel de trepte de calcul paralel.

Scheza de calcul

Seamnalul inițial, v_0 , obținut prin rearranjarea ecuațiilor lui x , este transformat succesiv în semnale intermediare $v_1, v_2, \dots, v_{L-1}$, semnalul final v_L fiind corespunzător cu TFD_{2^L}.

Fiecare semnal are $N=2^L$ ecuații: $v_l = [v_l[1], v_l[2], \dots, v_l[N]]^T, \forall l \in 0, L$

Practic, semnalul inițial este suficient succesor până când se obține rezultatul final.

v_0 este rearranjarea grupată a TFD₂.

Amplasarea ecuațiilor lui X în v_2 :

$v_2[k] = X[k-1], \forall k \in 1, 2^L$

Nu este necesară rearranjarea ecuațiilor.

Algoritmul FFT bazat pe segmentarea în timp

Prima variantă de calcul

Cum se poate realiza rearranjarea ecuațiilor lui x în v_0 ?

Pentru înțelegerea binară a indusului inițial.

Memory address (decimal) (binary): 0, 1, 2, 3, 4, 5, 6, 7

Memory: $x(0), x(1), x(2), x(3), x(4), x(5), x(6), x(7)$

Data decimation 1: $x(0), x(2), x(4), x(6)$

Data decimation 2: $x(0), x(4), x(2), x(6)$

Natural order

Bit-reversed order

Reprezentare binară a indusului inițial (pe L biți)

Observații și analiza de indusare pe nivelul indusurilor.

$(n_2 n_1 n_0) \rightarrow (n_0 n_2 n_1) \rightarrow (n_0 n_1 n_2)$

Reprezentare binară a indusului de argument (pe L biți)

Algoritmul FFT bazat pe segmentarea în timp

Prima variantă de calcul

Relațiile de calcul dintre ecuațiile lui v_{H+1} și v_L :

$v_{H+1}[2^{H+1}k+m] = v_L[2^{H+1}k+m] + w_{2^{H+1}}^{m-1} v_L[2^{H+1}k+2^H+m]$

$v_{H+1}[2^{H+1}k+2^H+m] = v_L[2^{H+1}k+m] + w_{2^{H+1}}^{2^H-m-1} v_L[2^{H+1}k+2^H+m]$

$\forall l \in 0, L-1$

$\forall m \in 1, 2^l$

$\forall k \in 0, 2^{L-l-1}-1$

Scheza generală de calcul

$v_l[2^{L-l}k+m]$

$v_{l+1}[2^{L-l}k+m]$

$w_{2^{L-l}}^{m-1}$

$w_{2^{L-l}}^{2^H-m-1}$

Doar înmulțiri și două adunări complexe!

Număr de operații

DFT: $O_L[N] = [4N^2]_{\bullet} + [2N(2N-1)]_{+} \sim 4N^2$

IFFT: $O_L[N] = [4N \log_2 N]_{\bullet} + [4N \log_2 N]_{+} \sim 4N \log_2 N$

Algoritmul FFT bazat pe segmentarea în timp

A doua variantă (eficientă) de calcul

Relații de calcul echivalente:

$v_{H+1}[2^{H+1}k+m] = v_L[2^{H+1}k+m] + w_{2^{H+1}}^{m-1} v_L[2^{H+1}k+2^H+m]$

$v_{H+1}[2^{H+1}k+2^H+m] = v_L[2^{H+1}k+m] + w_{2^{H+1}}^{2^H-m-1} v_L[2^{H+1}k+2^H+m]$

$\forall l \in 0, L-1$

$\forall m \in 1, 2^l$

$\forall k \in 0, 2^{L-l-1}-1$

Anulați: $w_{2^{L-l}}^{2^H-m-1} = -w_{2^{L-l}}^{m-1}$

Scheza generală echivalentă de calcul

$v_l[2^{L-l}k+m]$

$v_{l+1}[2^{L-l}k+m]$

$w_{2^{L-l}}^{m-1}$

-1

Doar o înmulțire și două adunări complexe!

Număr de operații

DFT: $O_L[N] = [4N^2]_{\bullet} + [2N(2N-1)]_{+} \sim 4N^2$

IFFT: $O_L[N] = [2N \log_2 N]_{\bullet} + [2N \log_2 N]_{+} \sim 2N \log_2 N$

Exemplu: $O_L[N] = 2^L$, $O_L[N] = 10 \times 2^L$

Număr de operații

$$O_{FFT-t}[N] = [2N \log_2 N]_{\bullet} + [2N \log_2 N]_{+} \sim 2N \log_2 N$$

① ③ Algoritmi fundamentali de tip FFT



Fast Fourier Transform

Algoritmul FFT-t

Principul segmentării în timp

$$X[k] = \sum_{n=0}^{N-1} x[n] w_N^{nk}, \quad \forall k \in \overline{0, N-1}$$

Segment cu eșantioane de ordin par

Segment cu eșantioane de ordin impar

- Exprimare echivalentă a TFD, pentru $N = 2M$

$$X[k] = \sum_{m=0}^{M-1} x[2m] w_{2M}^{2mk} + \sum_{m=0}^{M-1} x[2m+1] w_{2M}^{(2m+1)k} \quad \forall k \in \overline{0, N-1}$$

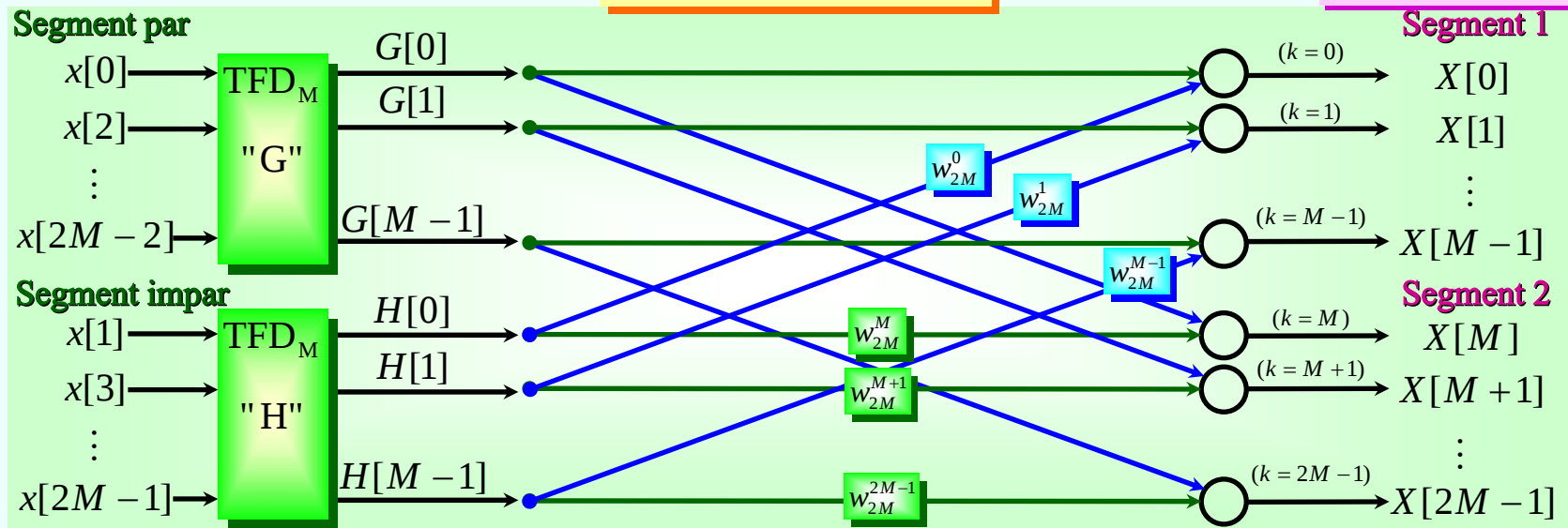
În calculul TFD_{2M} se poate utiliza o pereche de TFD_M aplicate segmentelor par și impar ale semnalului original.

$$X[k] = \underbrace{\sum_{m=0}^{M-1} x[2m] w_M^{mk}}_{G[k]} + w_{2M}^k \underbrace{\sum_{m=0}^{M-1} x[2m+1] w_M^{mk}}_{H[k]} \quad \forall k \in \overline{0, N-1}$$

$w_{2M}^{2mk} = w_M^{mk}$

Schema de calcul

Periodicitatea lui w_M^{mk}



① ③ Algoritmi fundamentali de tip FFT

Algoritmul FFT-t

Principul segmentării în timp

- Dacă $N = 4M$, semnalul poate fi partajat în 4 segmente (fiecare segment din perechea anterioară este împărțită într-o nouă pereche de segmente, mai scurte). Rezultă că TFD_{4M} poate fi evaluată cu ajutorul a 4 TFD_M .
- În general, dacă $N = 2^L$, semnalul poate fi partajat în 2^{L-1} segmente (fiecare segment avînd doar 2 eșantioane). Rezultă că TFD_N poate fi evaluată cu ajutorul a $(L-1)$ TFD_2 .

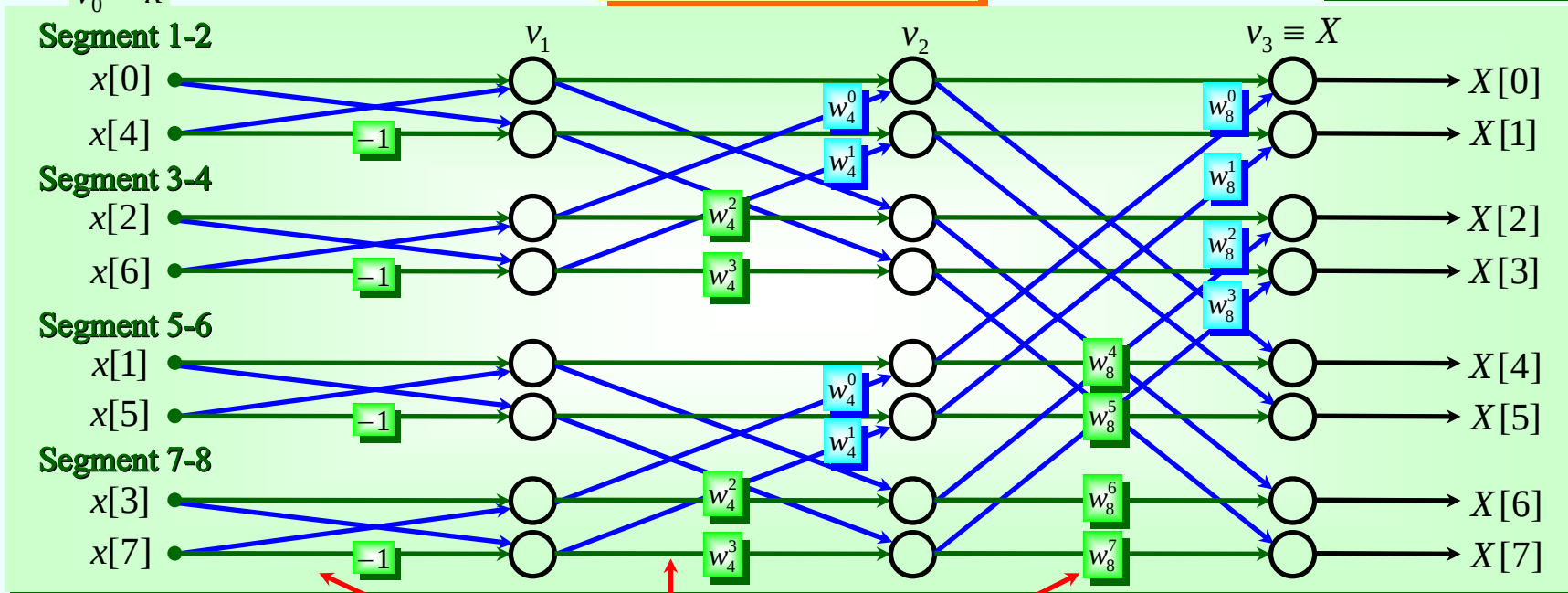
Exemplu

$$N = 8 = 2^3$$

$v_0 \sim x$

Schema de calcul

$$\begin{cases} X[0] = x[0] + x[1] \\ X[1] = x[0] - x[1] \end{cases}$$



3 trepte de calcul paralel

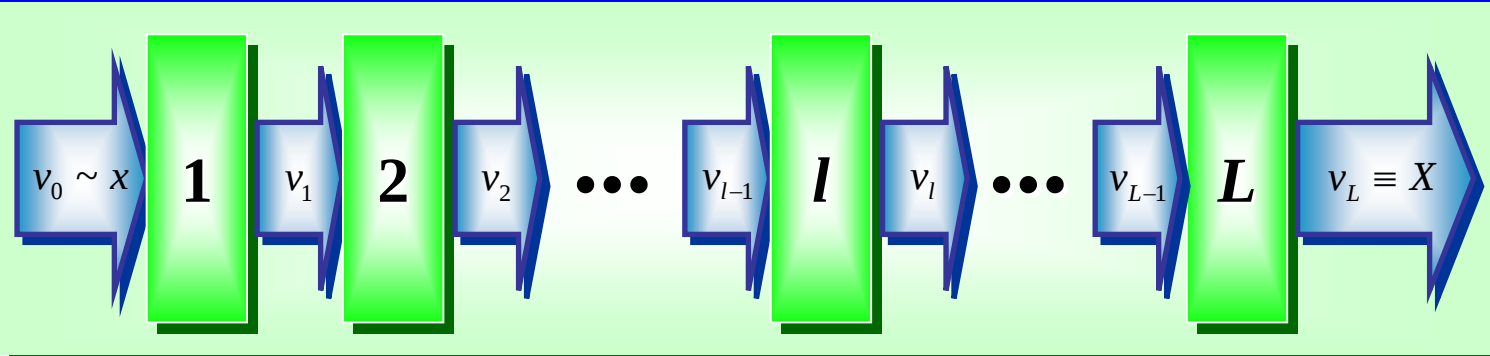
①③ Algoritmi fundamentali de tip FFT



Algoritmul FFT-t

Prima variantă de calcul

- În general, dacă $N = 2^L$, valorile TFD_N se pot obține cu ajutorul unei scheme de calcul având L trepte de calcul paralel.



Semnalul inițial, v_0 , obținut prin rearanjarea eșantioanelor lui x , este transformat succesiv în semnalele intermediare $v_1, v_2, \dots, v_{L-1}$, semnalul final v_L fiind coincident cu TFD_N .

☞ Fiecare semnal are $N = 2^L$ eșantioane.

$$v_l \equiv [v_l[1] \ v_l[2] \ \dots \ v_l[N]]^T, \quad \forall l \in \overline{0, L}$$

- Practic, semnalul inițial este rafinat succesiv pînă cînd se obține rezultatul final.
- v_0 este versiunea grosieră a TFD_N .
- Aranjarea eșantioanelor lui X în v_L :

$$v_L[k] = X[k-1], \quad \forall k \in \overline{1, 2^L}$$

☞ Nu este necesară rearanjarea eșantioanelor.

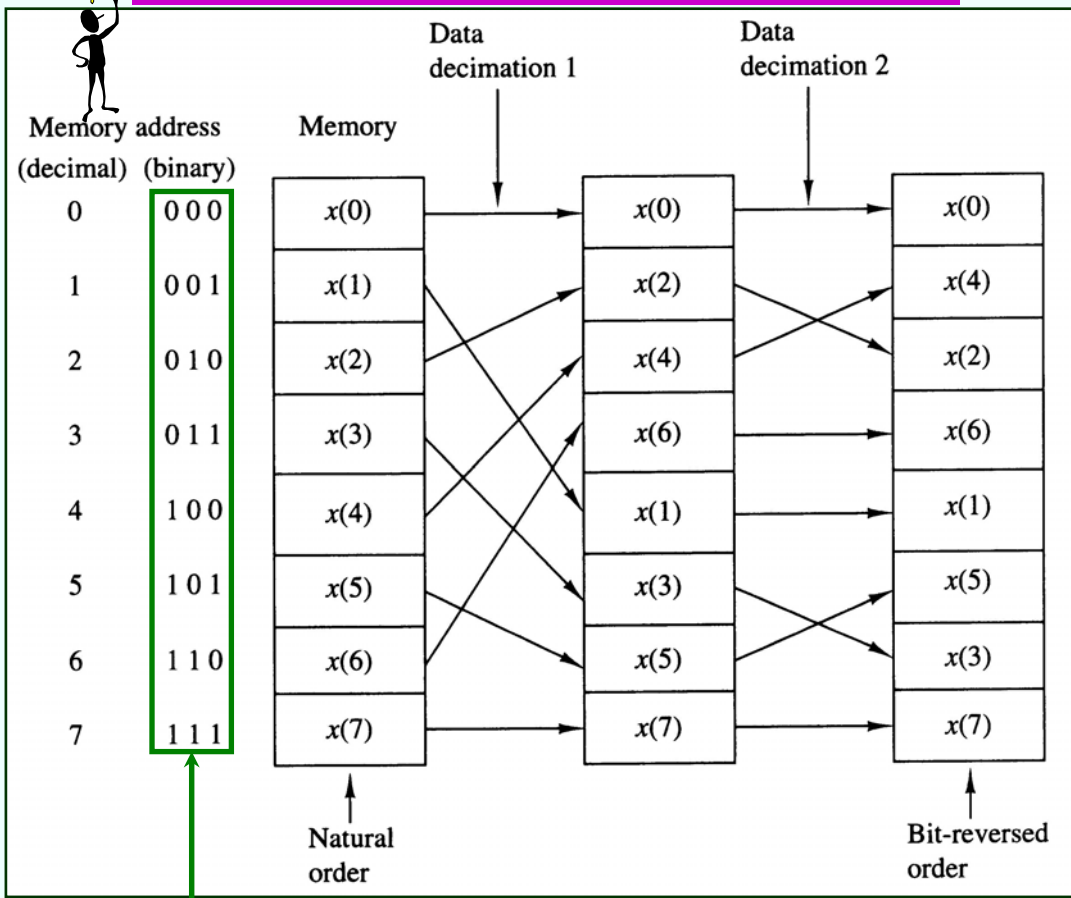
① ③ Algoritmi fundamentali de tip FFT

Algoritmul FFT-t

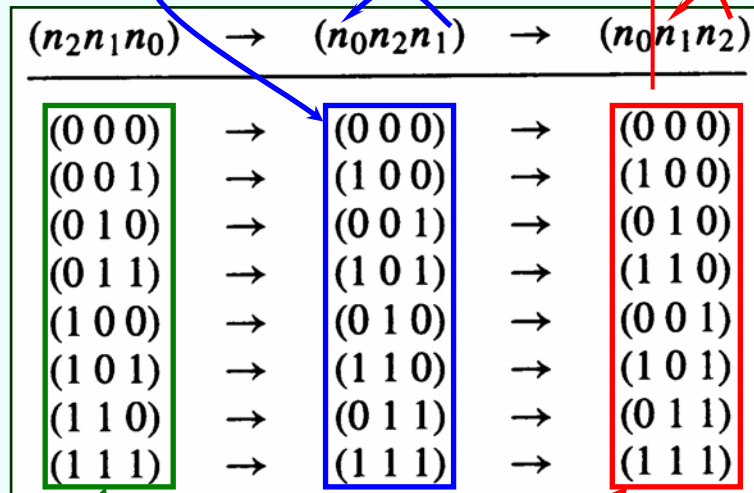
Prima variantă de calcul

Cum se poate realiza aranjarea eşantioanelor lui x în v_0 ?

Prin inversarea binară a indexului inițial.



Observați și maniera de indexare pe nivelul intermediar.



Reprezentare binară a indexului inițial (pe L biți)

Reprezentare binară a indexului de segment (tot pe L biți)

① ③ Algoritmi fundamentali de tip FFT



Algoritmul FFT-t

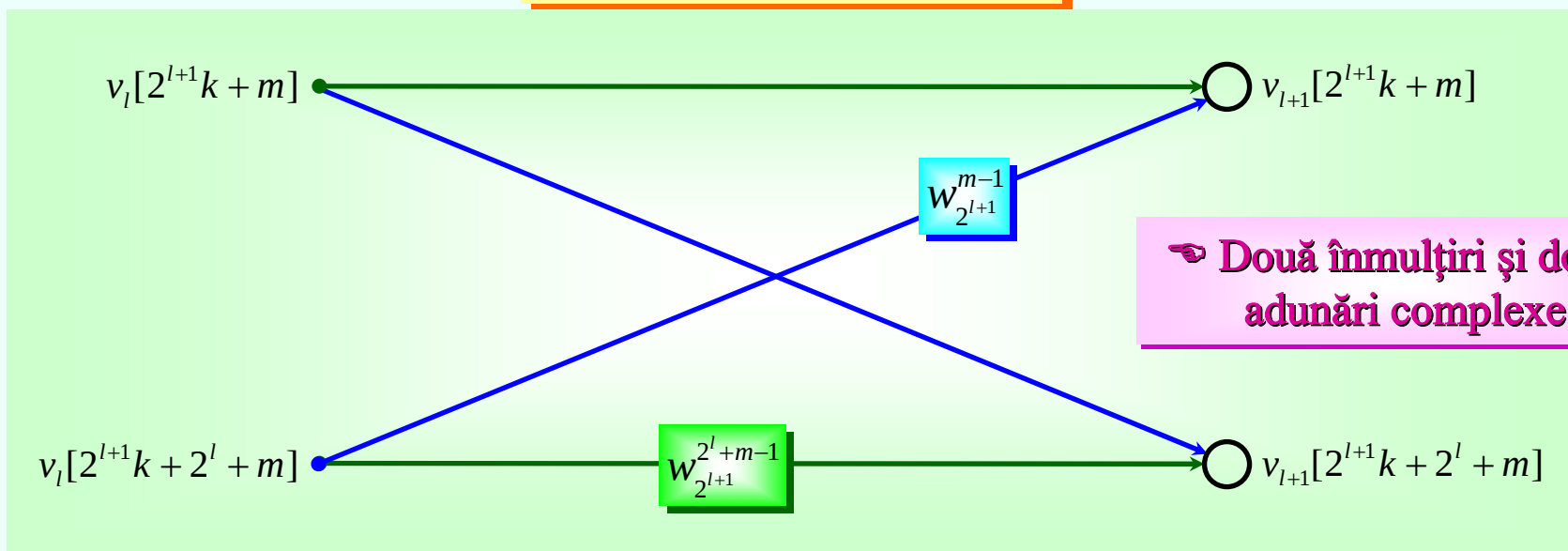
Prima variantă de calcul

- Relațiile de calcul dintre eșantioanele lui v_{l+1} și v_l :

$$\begin{cases} v_{l+1}[2^{l+1}k + m] = v_l[2^{l+1}k + m] + w_{2^{l+1}}^{m-1} v_l[2^{l+1}k + 2^l + m] \\ v_{l+1}[2^{l+1}k + 2^l + m] = v_l[2^{l+1}k + m] + w_{2^{l+1}}^{2^l + m - 1} v_l[2^{l+1}k + 2^l + m] \end{cases}$$

$$\begin{cases} \forall l \in \overline{0, L-1} \\ \forall m \in \overline{1, 2^l} \\ \forall k \in \overline{0, 2^{L-l-1} - 1} \end{cases}$$

Schema generică de calcul



DFT :

$$\mathcal{O}_0[N] = [4N^2]_{\bullet} + [2N(2N-1)]_{+} \sim 4N^2$$

FFT-t :

$$\mathcal{O}_2[N] = [4N \log_2 N]_{\bullet} + [4N \log_2 N]_{+} \sim 4N \log_2 N$$



Număr de operații

① ③ Algoritmi fundamentali de tip FFT

Algoritmul FFT-t

A doua variantă (eficientă) de calcul

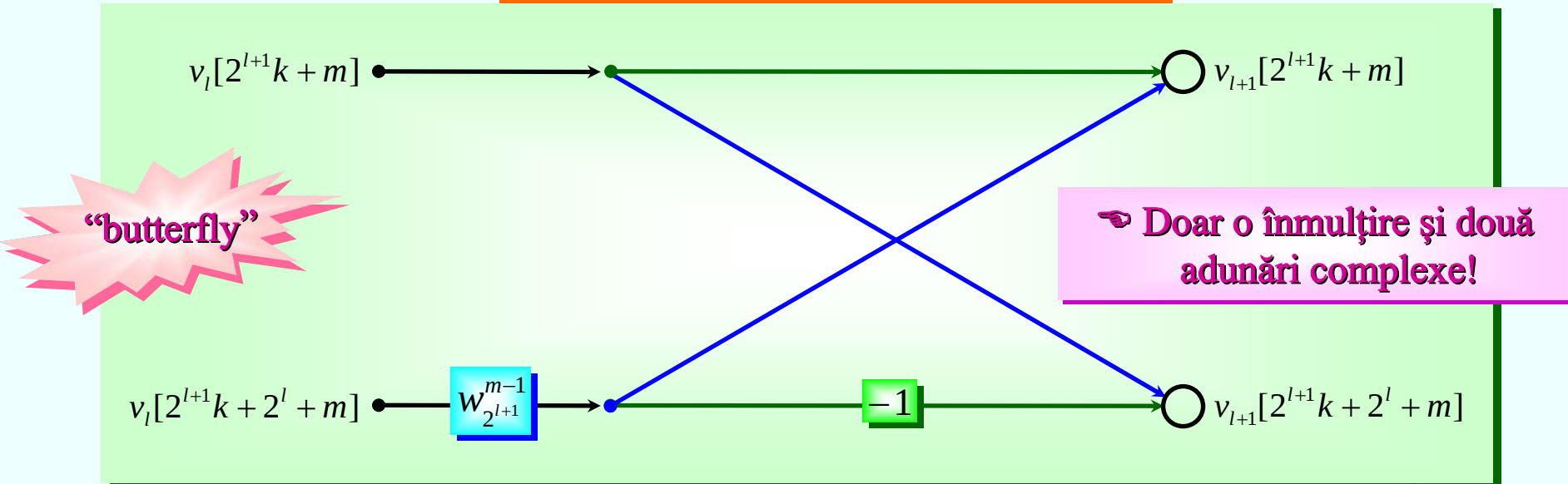
• Relații de calcul echivalente:

$$\begin{cases} v_{l+1}[2^{l+1}k + m] = v_l[2^{l+1}k + m] + w_{2^{l+1}}^{m-1} v_l[2^{l+1}k + 2^l + m] \\ v_{l+1}[2^{l+1}k + 2^l + m] = v_l[2^{l+1}k + m] - w_{2^{l+1}}^{m-1} v_l[2^{l+1}k + 2^l + m] \end{cases}$$

$$\begin{cases} \forall l \in \overline{0, L-1} \\ \forall m \in \overline{1, 2^l} \\ \forall k \in \overline{0, 2^{L-l-1} - 1} \end{cases}$$

Antisimetrie: $w_{2M}^{M+m} = -w_{2M}^m$

Schema generică echivalentă de calcul



Număr de operații

DFT :

$$\mathcal{O}_0[N] = [4N^2]_{\bullet} + [2N(2N-1)]_{+} \sim 4N^2$$

FFT-t :

$$\mathcal{O}_2[N] = [2N \log_2 N]_{\bullet} + [2N \log_2 N]_{+} \sim 2N \log_2 N$$

Exemplu

$$N = 1024 = 2^{10}$$

$$\mathcal{O}_0[N] \sim 2^{22}$$

$$\mathcal{O}_2[N] \sim 10 \times 2^{11}$$



164

