

# Procesarea semnalelor

## Transformata Discretă Cosinus

Paul Irofti

Universitatea din Bucureşti

Facultatea de Matematică și Informatică

Departamentul de Informatică

Email: [paul.irofti@fmi.unibuc.ro](mailto:paul.irofti@fmi.unibuc.ro)

# Discretizare și eșantionare

Continuu:

$$x(t) = \sin(2\pi f_0 t) \quad (1)$$

Discret:

$$x(n) = \sin(2\pi f_0 n t_s) = \sin(2\pi(f_0 + k f_s)n t_s) \quad (2)$$

unde

- ▶  $f_0$  – frecvență (Hz) măsoară numărul de oscilații într-o secundă
- ▶  $n$  – eșantionul, indexul în sirul de timpi  $0, 1, 2, \dots$
- ▶  $t_s$  – perioada de eșantionare; constantă (ex. la fiecare secundă)
- ▶  $nt_s$  – orizontul de timp (s)
- ▶  $f_0 n t_s$  – numărul de oscilații măsurat
- ▶  $2\pi f_0 n t$  – unghiul măsurat în radiani (vezi note de curs)
- ▶  $f_s$  – frecvență de eșantionare (Hz)
- ▶  $f_0 + k f_s$  – frecvență de aliere,  $\forall k \in \mathbb{N}$

# Transformata Fourier Discretă (DFT)

## Definiție

*Transformata Fourier a unui semnal discret (aperiodic):*

$$\begin{aligned} X(m) &= \sum_{n=0}^{N-1} x(n)e^{-j2\pi mn/N} \\ &= \sum_{n=0}^{N-1} x(n) [\cos(2\pi mn/N) - j \sin(2\pi mn/N)] \end{aligned} \tag{3}$$

- ▶  $X(m)$  – componenta  $m$  DFT (ex.  $X(0), X(1), X(2), \dots$ )
- ▶  $m$  – indicele componentei DFT în domeniul frecvenței ( $m = 0, 1, \dots, N - 1$ )
- ▶  $x(n)$  – eșantioanele în timp (ex.  $x(0), x(1), x(2), \dots$ )
- ▶  $n$  – indicele eșantioanelor în domeniul timpului ( $n = 0, 1, \dots, N - 1$ )
- ▶  $N$  – numărul eșantioanelor în timp la intrare și numărul componentelor în frecvență la ieșire

## Generalizare DFT

DFT aparține unei clase generale de transformări ortogonale de lungime finită de tipul:

$$X(m) = \sum_{n=0}^{N-1} x(n)\Phi_m^H(n) \quad (4)$$

$$x(n) = \frac{1}{N} \sum_{m=0}^{N-1} X(m)\Phi_m^H(n) \quad (5)$$

unde secvențele, vectorii,  $\Phi_m(n)$  formează o bază ortogonală:

$$\Phi_i\Phi_j^H = \frac{1}{N} \sum_{n=0}^{N-1} \Phi_i(n)\Phi_j^H(n) = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases} \quad (6)$$

Unde  $H$  este operatorul de conjugare și transpunere.

## Baza ortogonală DFT

În cazul DFT baza ortogonală este

$$W_N = e^{-j2\pi/N} \quad (7)$$

Putând rescrive DFT și IDFT:

$$X(m) = \sum_{n=0}^{N-1} x(n) W_N^{mn} \quad (8)$$

$$x(n) = \frac{1}{N} \sum_{m=0}^{N-1} X(m) W_N^{-mn} \quad (9)$$

Notația diferă un pic față de formularea generală, dar este cea întâlnită în literatură.

## Baza ortogonală reală ( $\Phi_m \in \mathbb{R}$ )

În cazul DFT baza ortogonală este alcătuită din numere complexe:

$$W_N = e^{-j2\pi/N} = \cos(2\pi/N) + j \sin(2\pi/N) \quad (10)$$

dar multe semnale au doar componente reale și sunt prelucrate pe calculatoare numerice ce suportă nativ numere reale.

→ nevoie unei baze reale a.î. când  $x(n) \in \mathbb{R}$  să fie și  $X(m) \in \mathbb{R}$

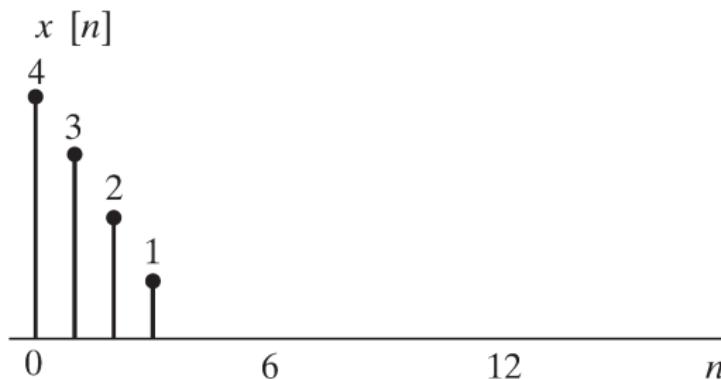
Apar mai multe transformate: Haar, Hadamard, DCT.

Cea mai apropiată de DFT este DCT cu multiple aplicații în practică (ex. JPEG, MP3 etc.)

## Transformata cosinus discretă (DCT)

DCT folosește cosinus pentru a forma baza vectorială  $\Phi_m$ .

Fie  $x(n)$  un semnal cu  $N = 4$  eșantioane

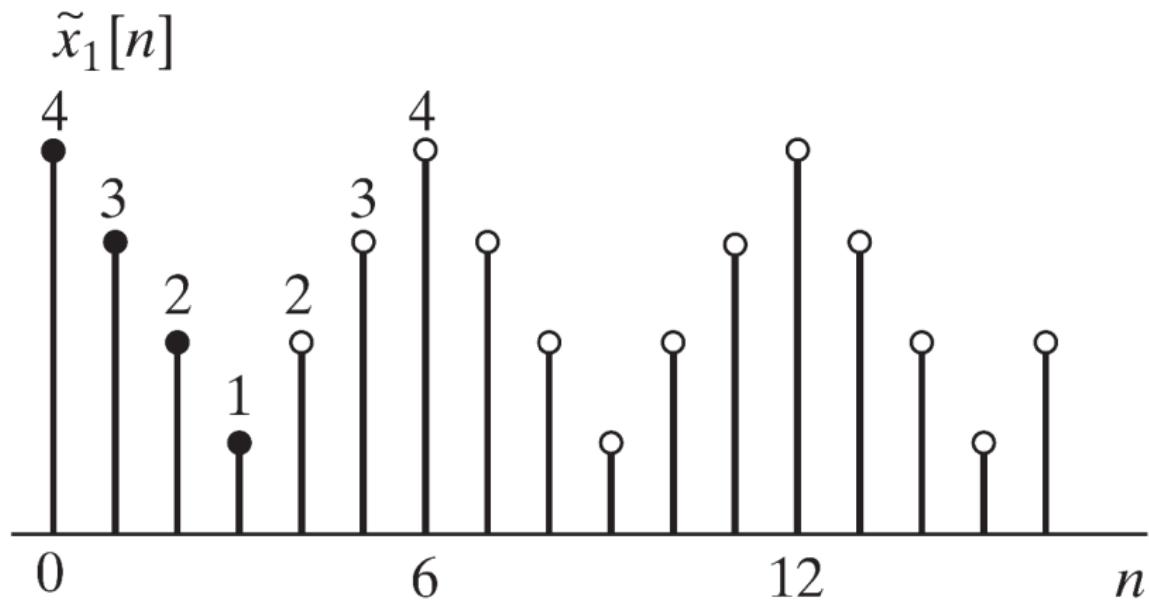


Cosinus este periodică și pară,  $\cos(x) = \cos(-x)$ , deci extensia unui semnal  $x(n)$  în afara intervalului  $n = \overline{0 : N - 1}$  trebuie să fie periodică și pară.

Există mai multe moduri de a extinde un semnal ceea ce duce la mai multe tipuri de transformată DCT.

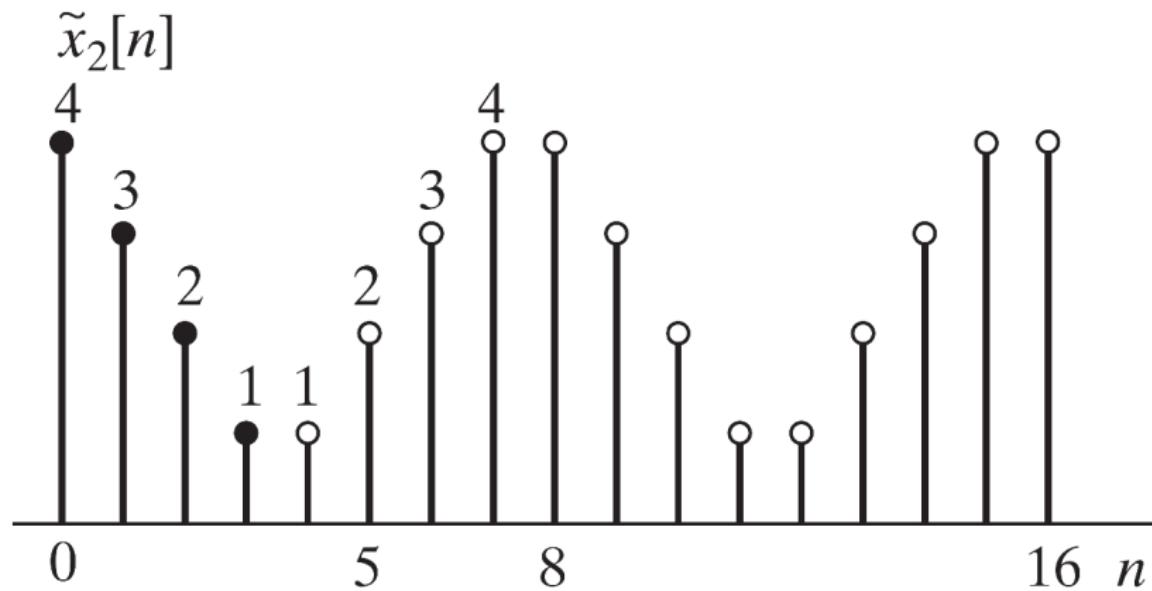
## DCT-I (Type-1)

Extensia la un semnal  $\tilde{x}_1(n)$  periodic și par de 17 eșantioane.



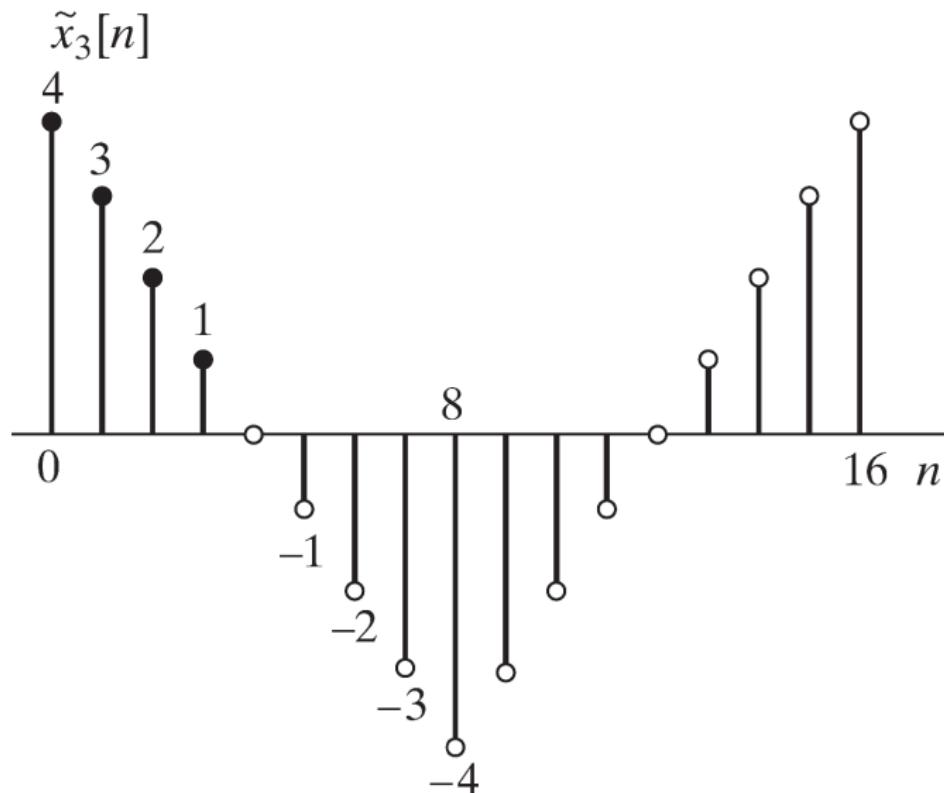
Perioadă:  $2N - 2 = 6$  și simetrie pară la  $n = \{0, N - 1\}$

## DCT-II (Type-2)



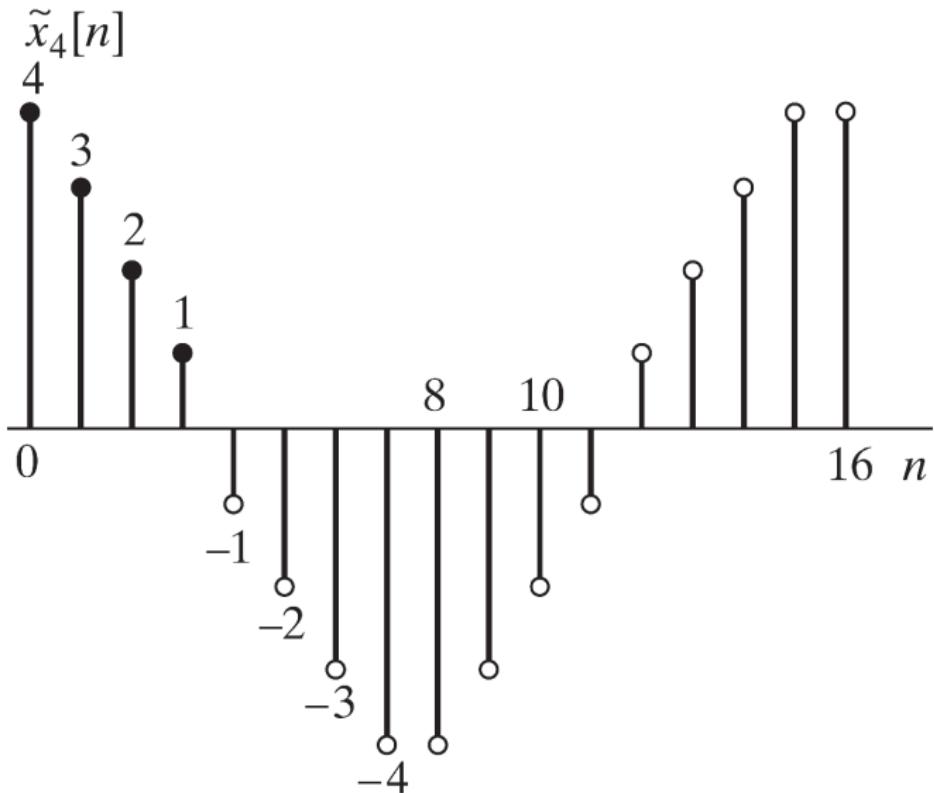
Perioadă:  $2N = 8$  și simetrie pară la  $n = \{-\frac{1}{2}, \frac{7}{2}\}$

## DCT-III (Type-3)



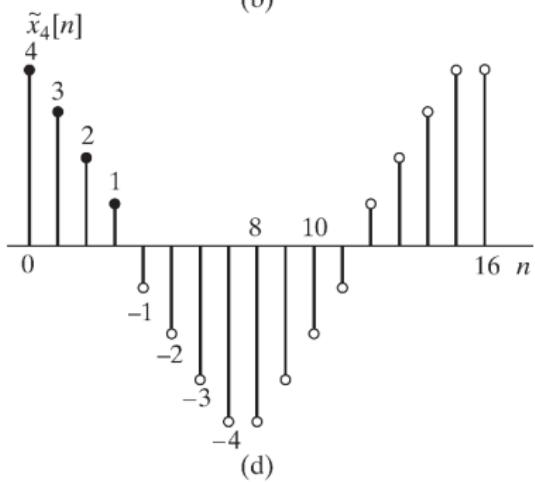
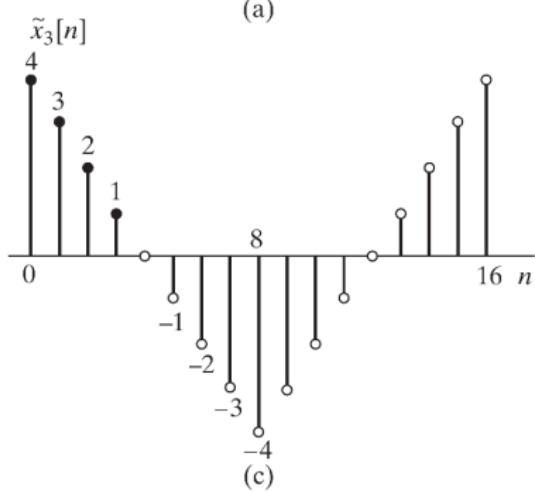
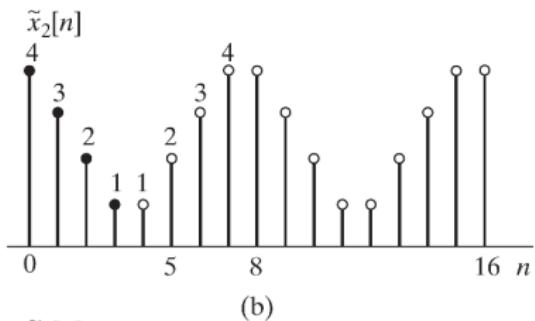
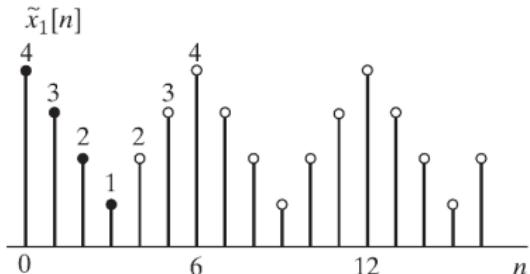
Perioadă:  $4N = 16$  și simetrie pară la  $n = \{0, 8\}$

## DCT-IV (Type-4)



Perioadă:  $4N = 16$  și simetrie pară la  $n = \{-\frac{1}{2}, \frac{15}{2}\}$

# Varintele DCT



Variantele sunt copii deplasate de  $N$  eșantioane  $\pm x(n)$  și  $\pm x(-n)$ .

## Calcul DCT-I

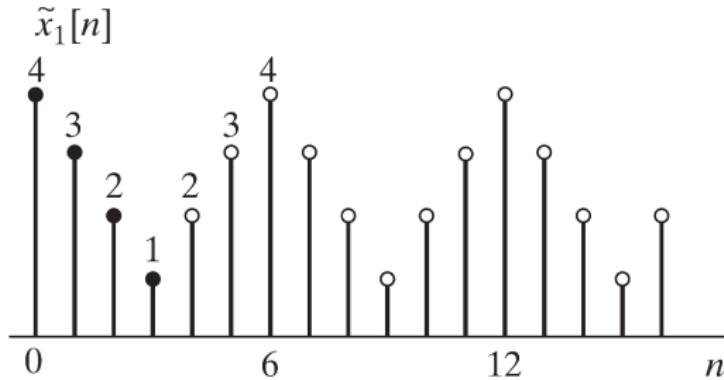
Secvența de perioadă  $2N-2$  este produsă prin extinderea semnalului original:

$$\tilde{x}_1(n) = x_\alpha(n \bmod 2N - 2) + x_\alpha((-n) \bmod 2N - 2) \quad (11)$$

unde  $x_\alpha(n) = \alpha(n)x(n)$  cu  $\begin{cases} \frac{1}{2}, & n = \{0, N-1\} \\ 1, & n = \overline{1 : N-2} \end{cases}$ .

Semnalul rezultat este identic cu originalul în intervalul  $1 : N-1$  și este par cu puncte de simetrie la  $n = \{0, N-1, 2N-2, \dots\}$

## Calcul extindere DCT-I



$$\tilde{x}_1(0) = \frac{1}{2}x(0) + \frac{1}{2}x(-0) = x(0); \quad \tilde{x}_1(4) = x(4) + x(2) = x(2);$$

$$\tilde{x}_1(1) = x(1) + \underbrace{x(5)}_{-1 \bmod 6} = x(1); \quad \tilde{x}_1(5) = x(5) + x(1) = x(1);$$

$$\tilde{x}_1(2) = x(2) + x(4) = x(2); \quad \tilde{x}_1(6) = \frac{1}{2}x(0) + \frac{1}{2}x(0) = x(0);$$

$$\tilde{x}_1(3) = \frac{1}{2}x(3) + \frac{1}{2}x(3) = x(3); \quad \tilde{x}_1(7) = x(1) + x(5) = x(1);$$

# Transformata DCT-I

Simetrie pară periodică în punctele  $n = \{0, N - 1, 2N - 2, \dots\}$ .

## Definiție

Transformata DCT-I este definită de perechea:

$$X_1(m) = 2 \sum_{n=0}^{N-1} \alpha(n)x(n) \cos \left( \frac{\pi mn}{N-1} \right) \quad (12)$$

$$x_1(n) = \frac{1}{N-1} \sum_{m=0}^{N-1} \alpha(m)X_1(m) \cos \left( \frac{\pi mn}{N-1} \right) \quad (13)$$

## Transformata DCT-II

Relația de extindere a semnalului este dată de:

$$\tilde{x}_2(n) = x_\alpha(n \bmod 2N) + x_\alpha((-n - 1) \bmod 2N) \quad (14)$$

Simetrie pară periodică în punctele  $n = \{-\frac{1}{2}, N - \frac{1}{2}, 2N - \frac{1}{2}, \dots\}$ .

### Definiție

Transformata DCT-II este definită de perechea:

$$X_2(m) = 2 \sum_{n=0}^{N-1} x(n) \cos \left( \frac{\pi m(2n - 1)}{2N} \right) \quad (15)$$

$$x_2(n) = \frac{1}{N} \sum_{m=0}^{N-1} \beta(m) X_2(m) \cos \left( \frac{\pi m(2n + 1)}{2N} \right) \quad (16)$$

cu  $\beta(0) = \frac{1}{2}$  și  $\beta(m) = 1$  în rest.

## Transformata DCT-II unitară

În multe aplicații avem nevoie de o transformată unitară (ce distribuie factorii de-a lungul vectorilor din bază).

### Definiție

*Transformata DCT-II unitară este definită de perechea:*

$$X_2(m) = \sqrt{\frac{2}{N}} \tilde{\beta}(m) \sum_{n=0}^{N-1} x(n) \cos\left(\frac{\pi m(2n+1)}{2N}\right) \quad (17)$$

$$x_2(n) = \sqrt{\frac{2}{N}} \sum_{m=0}^{N-1} \tilde{\beta}(m) X_2(m) \cos\left(\frac{\pi m(2n+1)}{2N}\right) \quad (18)$$

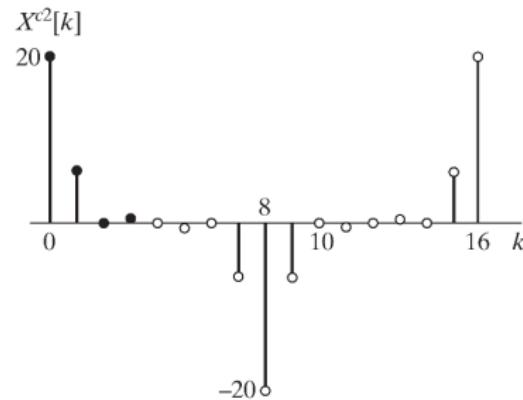
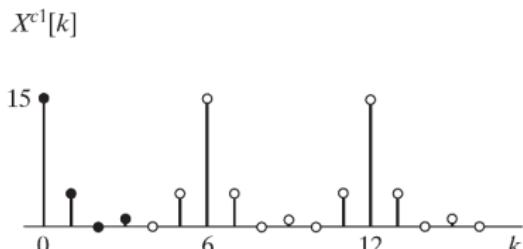
cu  $\tilde{\beta}(0) = \frac{1}{\sqrt{2}}$  și  $\tilde{\beta}(m) = 1$  în rest.

Unitar:  $\sum_{n=0}^{N-1} x(n)^2 = \sum_{m=0}^{N-1} X(m)^2$  (conservă norma).

## Exemplu: transformata DCT-I și DCT-II

În partea dreaptă avem transformata DCT-I și DCT-II pentru semnalul cu  $N = 4$  eșantioane.

- ▶ periodicitate după  $N - 1$
- ▶ simetria nu respectă tipul DCT
- ▶ simetrie identică pentru  $X_1$
- ▶ dar diferită pentru  $X_2$
- ▶  $X_2$  are simetria lui  $\tilde{x}_3$  (DCT-III)
- ▶ perioada lui  $\tilde{x}_2$  este  $2N$
- ▶ perioada lui  $X_2$  este  $4N!$



# Teorema lui Parseval

## Teoremă

Dacă  $x(n) \xrightleftharpoons[IFT]{FT} X(e^{j\omega})$  atunci:

$$E = \sum_{n=-\infty}^{\infty} |x(n)|^2 = \frac{1}{2\pi} \int_{-\pi}^{\pi} |X(e^{j\omega})|^2 d\omega \quad (19)$$

unde funcția  $|X(e^{j\omega})|^2$  se numește și densitatea energiei spectrului pentru că determină distribuția energiei în domeniul frecvenței.

# Compactarea energiei

DCT este folosită în multe aplicații ce implică compresia datelor datorită proprietății sale de compactare a energiei.

## Remarcă

*DCT-II a unui semnal are adesea coeficienții adunați la indicii mici  $X(m)$  ai transformatei față de DFT care îi are distribuții de-a lungul tuturor indicilor.*

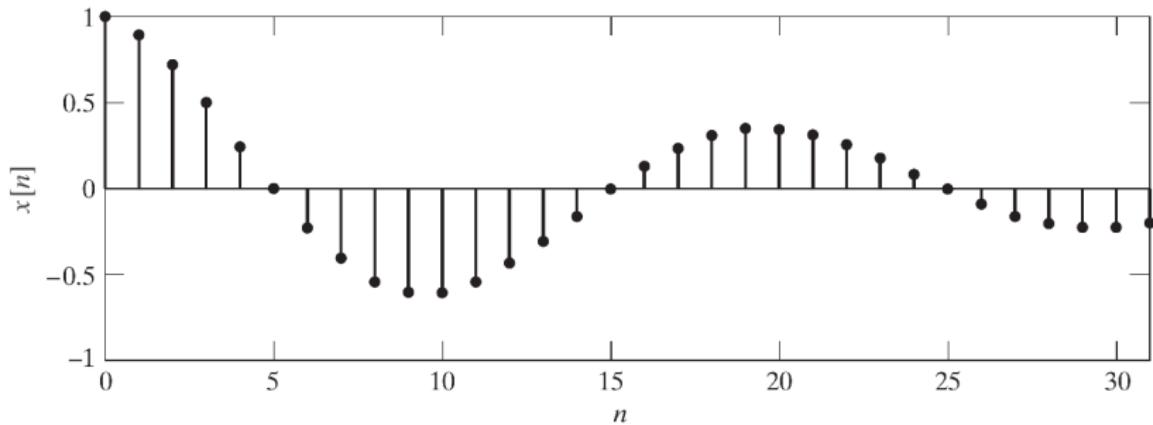
Remarca devine importantă în contextul teoremei lui Parseval:

$$\text{DCT-I: } \sum_{n=0}^{N-1} \alpha(n) |x(n)|^2 = \frac{1}{2N-2} \sum_{n=0}^{N-1} \alpha(m) |X(m)|^2 \quad (20)$$

$$\text{DCT-II: } \sum_{n=0}^{N-1} |x(n)|^2 = \frac{1}{N} \sum_{n=0}^{N-1} \beta(m) |X(m)|^2 \quad (21)$$

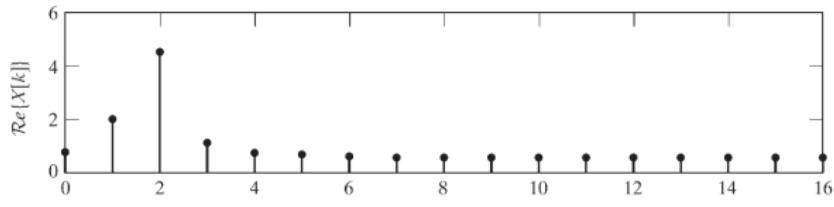
## Exemplu: compactarea energiei

Fie semnalul  $x(n) = a^n \cos(\omega_0 n + \varphi)$ .

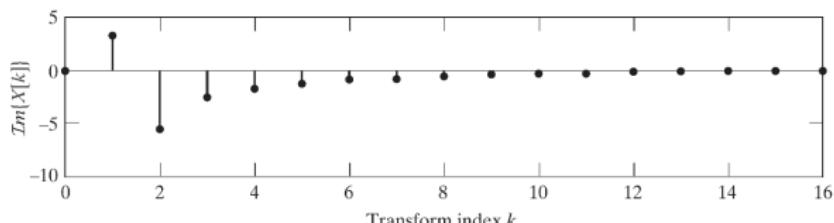


în figură cu parametrizarea  $a = 0,9$ ,  $f_0 = 0,1\pi$  și  $N = 32$ .

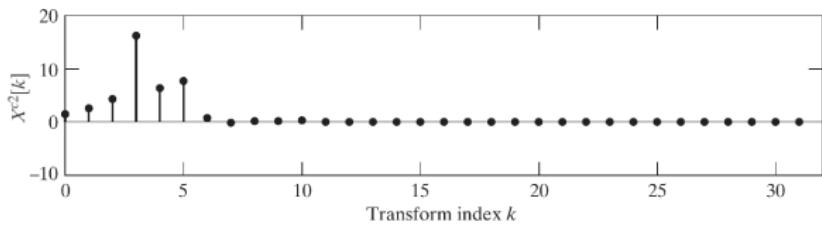
## Exemplu: compactarea energiei DFT vs. DCT



(a)



(b)



(c)

DFT și DCT-II pentru semnalul  $x(n)$  din figura anterioară.

## Exemplu: trunchiere DFT

Trunchierea în frecvență a DFT:

$$x^{\text{dft}}(n) = \frac{1}{N} \sum_{m=0}^{N-1} T_k(m) X(m) e^{j2\pi mn/N} \quad (22)$$

se efectuează printr-un operator de *hard-thresholding*:

$$T_k(m) = \begin{cases} 1, & 0 \leq m \leq (N - 1 - k)/2 \\ 0, & (N + 1 - k)/2 \leq m \leq (N - 1 + k)/2 \\ 1, & (N + 1 + k)/2 \leq m \leq N - 1 \end{cases} \quad (23)$$

- ▶ pentru  $k = 1$ , termenul  $X(N/2)$  este anulat.
- ▶ pentru  $k = 3$ , termenii  $X(N/2), X(N/2 - 1)$  și termenul conjugat  $X(N/2 + 1)$  sunt anulați
- ▶ s.a.m.d. pentru  $k = \{1, 3, 5, \dots, N - 1\}$ .

## Exemplu: mean squared error (MSE)

Trunchierea în frecvență a DCT-II este mai simplă:

$$x^{\text{dct}}(n) = \frac{1}{N} \sum_{m=0}^{N-1-k} \beta(m) X(m) \cos\left(\frac{\pi m(2n+1)}{2N}\right) \quad (24)$$

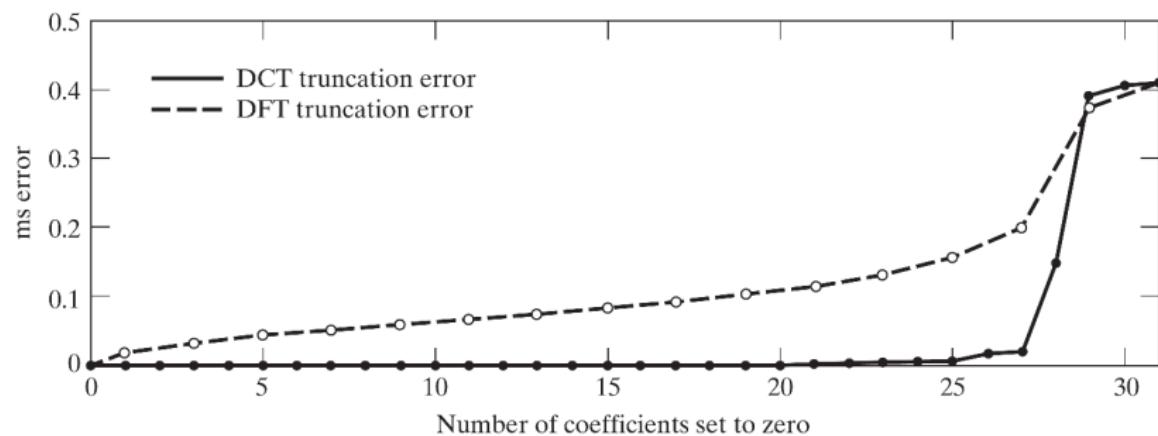
Comparăm cele două aproximări cu ajutorul medie pătratică a erorii:

$$E^{\text{dft}}(k) = \frac{1}{N} \sum_{n=0}^{N-1} |x(n) - x_k^{\text{dft}}(n)|^2 \quad (25)$$

$$E^{\text{dct}}(k) = \frac{1}{N} \sum_{n=0}^{N-1} |x(n) - x_k^{\text{dct}}(n)|^2 \quad (26)$$

## Exemplu: Rezultat DFT vs. DCT-II

Trunchierea în funcție de numărul de coeficienți anulați  $k$ :



# Cuantizare

Adesea împreună cu trunchierea sunt folosite tehnici de cuantizare.

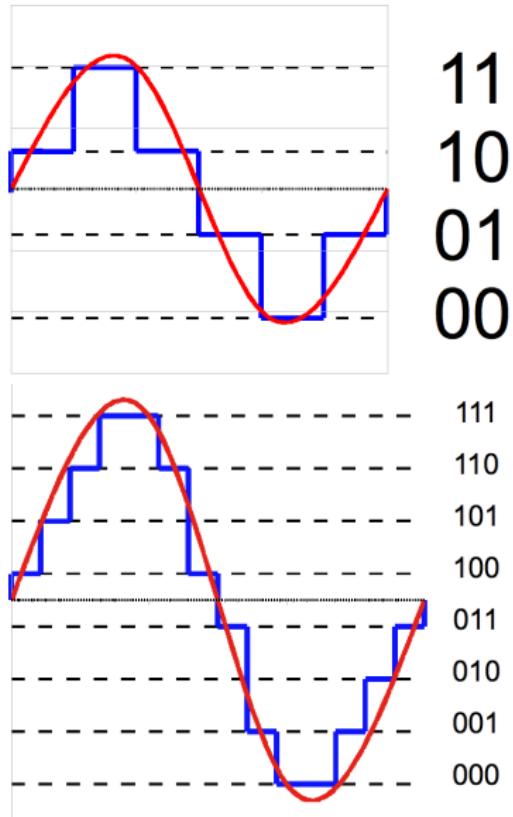
Cuantizarea este similară cu discretizarea:

- ▶ alegem anumiți pași de cuantizare  $\Delta$
- ▶  $\Delta$  este similar cu bin-urile de la DFT
- ▶ către aceste cuante aproximăm valorile  $x(n)$  ale semnalului dat

$$Q(n) = \Delta \left\lfloor \frac{x(n)}{\Delta} + \frac{1}{2} \right\rfloor \quad (27)$$

unde  $\Delta = 1$  pentru rotunjirea către cel mai apropiat întreg.

## Exemplu: cuantizare pe 2-biți și 3-biți



Sursă: [https://en.wikipedia.org/wiki/Quantization\\_\(signal\\_processing\)](https://en.wikipedia.org/wiki/Quantization_(signal_processing))

# JPEG

Ipotezele JPEG bazate pe modul de funcționare al ochilor umani:

- ▶ ochii sunt mai sensibili la luminozitatea culorii decât la paleta cromatică
- ▶ ochii nu observă conținutul de frecvență înaltă din imagini

Algoritmul de compresie JPEG are patru etape:

1. transformarea imaginii din pixeli RGB în Y'CbCr
2. aplicarea 2D-DCT pe blocuri disincte de 8x8 pixeli din imagine
3. cuantizarea în frecvență cu  $Q$  dat de standardul JPEG
4. optional compresia rezultatului cu coduri Huffman

# Algoritmul JPEG fără compresie Huffman

Fie imaginea  $I \in \mathbb{R}^{h \times w \times 3}$  păstrată drept un tensor tridimensional cu luminozitatea Y', cromatică albastră Cb, și cromatică roșie Cr.

1. cuantizarea (*down-sampling*) paletei cromatice:

$$I_{Cb} = Q_{Cb} \left\lfloor \frac{I_{Cb}}{Q_{Cb}} + \frac{1}{2} \right\rfloor; \quad I_{Cr} = Q_{Cr} \left\lfloor \frac{I_{Cr}}{Q_{Cr}} + \frac{1}{2} \right\rfloor \quad (28)$$

2. pentru fiecare dimensiune  $d \in \{Y', Cb, Cr\}$

2.1 pentru fiecare bloc  $x$  de  $8 \times 8$  pixeli

2.1.1  $X = DCT(x)$

2.1.2  $X = Q_{JPEG} \left\lfloor \frac{X}{Q_{JPEG}} + \frac{1}{2} \right\rfloor$

2.1.3  $x = IDCT(X)$

unde pașii de cuantizare  $Q_{JPEG}$  sunt dați de către standardul JPEG.