The Discrete Fourier Transform

Multimedial Signal Processing 1st Module

Politecnico di Milano –

Polo regionale di Como

Particulars

- Eliana Frigerio
  - efrigerio@elet.polimi.it
  - Phone: 02 2399 9653
  - Send to me an email in order to organize conferences and for any question

Summary:

- Ex1: Discrete Fourier Transform computation
- Ex2: DFT's property:
  - Linearity Circular folding
  - Symmetry properties for real sequences
  - Conjugation Circular shift
  - Circular convolution Multiplication
  - Parseval's relation
- Ex3: Linear convolution
- Ex4: Block convolution

Exercise 1 (1/16)

- THEOREM: If x(n) is time-limited, then N equispaced samples of the DTFT X($e^{j\omega_k}$) can uniquely reconstruct X($e^{j\omega}$) for all the frequencies.
- These N samples around the unit circle are calle the Dicrete Fourier transform coefficients:

$$DFT_k(x) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}$$

Exercise 1 (2/16)

- . $$DFT_k(x) = X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}$$
  - k corresponds to $\omega_k \stackrel{\Delta}{=} \frac{2\pi}{N} k \quad k = 0,1,\ldots,N$ which are (N+1) equispaced frequencies between [0, 2π]
  - DFT is a complex signal
- . $$IDFT_n(X) = x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j2\pi kn/N}$$

Exercise 1 (3/16)

- Goal: compute the DFT in accordance with its definition:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N}$$

- N.B.: Pay attention to the fact that Matlab indexes start from 1 and not 0 as in DFT definition

Exercise 1 (4/16)

- **function [X,w]=dft_for(x,N)**
- w = [0:1:N-1]*2*pi/N;   % defined in radiant
- X = zeros(1,length(w));
- n = [0:1:N-1]
- f_index=1;
- for omega = w
-     t_index=1;
-     for t = n
-         X(f_index)=X(f_index)+ x(t_index)*exp(-1i*(t)*(omega));
-         t_index=t_index+1;
-     end
-     f_index=f_index+1;
- end

Exercise 1 (5/16)

- Exercise1: Given the sequence x(n)=[1 1 1 1]
- Compute and plot its DFT:
  - x = [1,1,1,1];
  - N = length(x);
  - [X, w] = dft(x,N);
  - stem(w*N/(2*pi),abs(X);

% we divided the w array by 2*pi and multiplied by N before plotting so that the frequeny axes are in the sample unit k.

Exercise 1 (6/16)

- The DFT computation can be implemented as a matrix-vector multiplication operation:

- Rearranging as column vectors: $\overrightarrow{X} = W\vec{x}$

  - Where: $W = e^{-j\left(2\pi/N\right)kn} \qquad n,k = 0,1,\ldots N$

  - rearranging also k and n as row vectors

$$W = \left[e^{-j\left(2\pi/N\right)\overrightarrow{k^T}\vec{n}} \right]_{NxN}$$

Exercise 1 (7/16)

Working with row vectors:

$$\overrightarrow{X^T} = \overrightarrow{x^T} W^T = \overrightarrow{x^T}\left[e^{-j\left(2\pi/N\right)\overrightarrow{n^T}\vec{k}} \right]$$

- Pseudocode:
- n = [0:1:N-1];   % row vector for n
- k = [0:1:N-1];   % row vecor for k
- WN = exp(-j*2*pi/N);   % Wn factor
- nk = n'*k;   % creates a N by N matrix of nk values
- WNnk = WN .^ nk;   % DFT matrix
- Xk = xn * WNnk;

Exercise 1 (8/16)

- The functions $s_k = \left(W_N^k\right)^n = e^{j2\pi kn/N}$, in the definition, are the sinusoidail basis set of the DFT. These are complex functions that make a whole number of periods in N samples.

- Ex1d: See the real part of each of these for different values of k.

Exercise 1 (9/16)

- N.B.: k±mN refers to the same sinusoid for all integer m.
- Choose N=4:
  - s_0 has 0 periods in N samples
  - s_1 has 1 period in N samples
  - s_2 has 2 periods in N samples
  - $s_3 = s_{3-4} = s_{-1}$ has 1 period in N samples

Exercise 1 (10/16)

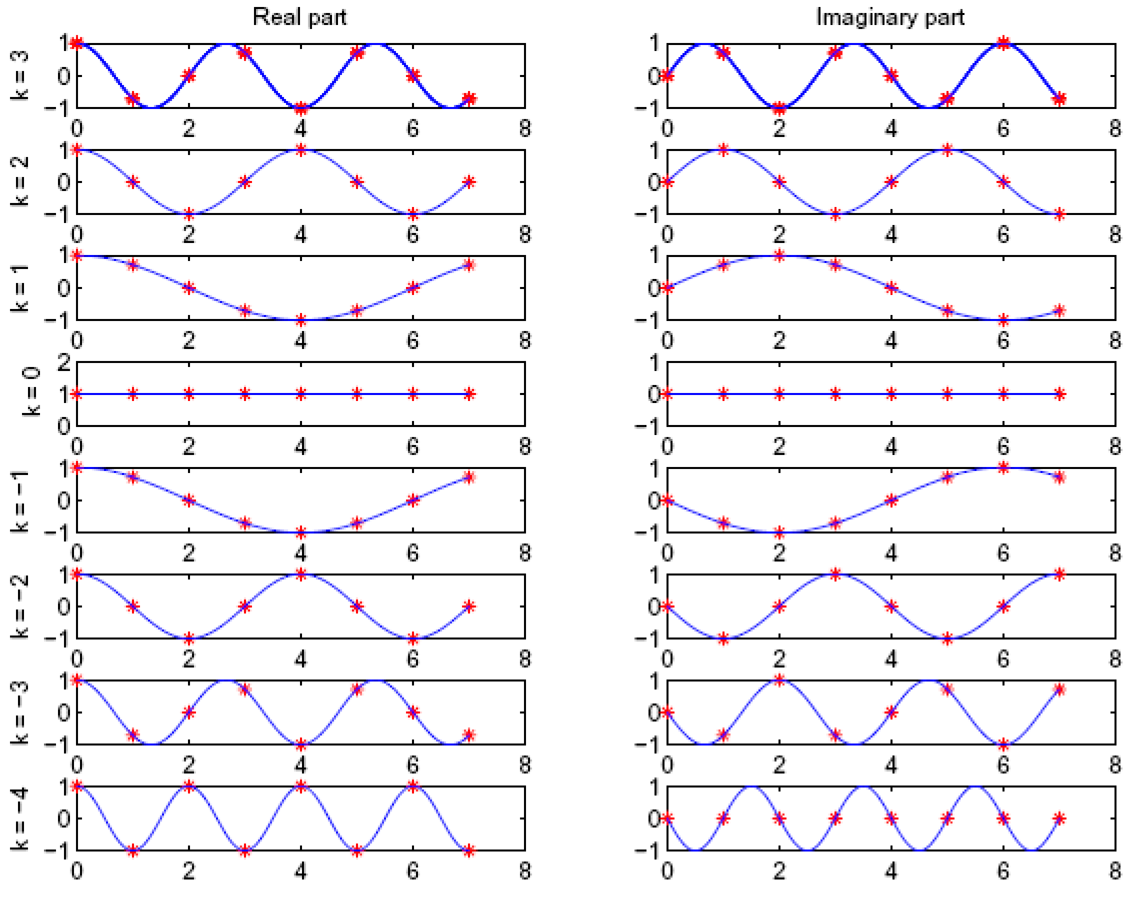


Fig. 1.1. DFT sinusoids, for $N = 8$

Exercise 1 (11/16)

- Given the sequence x(n)=[1 1 1 1]
- Compute and plot its DFT using the matrix implementation:

  - x = [1,1,1,1];
  - N = length(x);
  - [Xk] = dft_matrix(x,N)
  - k = [0:1:N-1]
  - figure, subplot(2,1,1);stem(k,abs(Xk));

Exercise 1 (12/16)

- Matlab provides the function “X=fft(x)” that computes the FFT of “x” and stores it in “X”.
- EX1a:Compare the results of the dft and the fft functions.
- Ex1a:Compute the anti-transform and compare it with the original signal.

Exercise 1 (13/16)

- Ex1c:The frequency domain can be defined as:
  - Index of the sample k=[0 : N-1]
  - Frequency $f = \frac{Fs \cdot k}{N}$ Hz [0 : Fs)
  - Normalized frequency $f_{norm} = f \cdot T = \frac{f}{Fs} = \frac{k}{N}$ [0 : 1)
  - Radian frequency $\omega = 2\pi \frac{k}{N} = 2\pi f_{norm}$ rad/sample [0 : 2п)

Exercise 1 (14/16)

EXAMPLE 5.6 Let $x(n)$ be a 4-point sequence:

$$x(n) = \begin{cases} 1, & 0 \le n \le 3 \\ 0, & \text{otherwise} \end{cases}$$

a. Compute the discrete-time Fourier transform $X(e^{j\omega})$ and plot its magnitude and phase.

b. Compute the 4-point DFT of $x(n)$.

Exercise 1 (15/16)

□ **EXAMPLE 5.7** How can we obtain other samples of the DTFT $X(e^{j\omega})$?

Solution It is clear that we should sample at dense (or finer) frequencies; that is, we should increase N. Suppose we take twice the number of points, or $N = 8$ instead of 4. This we can achieve by treating $x(n)$ as an 8-point sequence by *appending* 4 zeros.

$$x(n) = \{\underset{\uparrow}{1}, 1, 1, 1, 0, 0, 0, 0\}$$

N.B.: The zero-padding gives us a high-density spectrum, but not a high resolution spectrum because non new information is added to the signal.

Exercise 1 (16/16)

EXAMPLE 5.8 To illustrate the difference between the high-density spectrum and the high-resolution spectrum, consider the sequence

$$x(n) = \cos(0.48\pi n) + \cos(0.52\pi n)$$

We want to determine its spectrum based on the finite number of samples.

- a. Plot the DFT of x(n) for $0 \le n \le 100$. High resolution spectrum
- b. Plot the DFT of x(n) for $0 \le n \le 10$.
- c. Plot the 100 samples of the DFT of x(n) for $0 \le n \le 10$ (zero padding). High density spectrum.

Summary:

- Ex2: DFT's property:
  - Linearity
  - Circular folding
  - Conjugation
  - Symmetry properties for real sequences
  - Circular shift
  - Circular convolution
  - Multiplication
  - Parseval's relation

Exercise 2 (1/24)

- PROPERTIES:
  - 1. Linearity:

$$DFT\{\alpha x_1(n)+\beta x_2(n)\}=\alpha\cdot DFT\{x_1(n)\}+\beta\cdot DFT\{x_2(n)\}$$

$$N_3=\max\{N_1,N_2\}$$

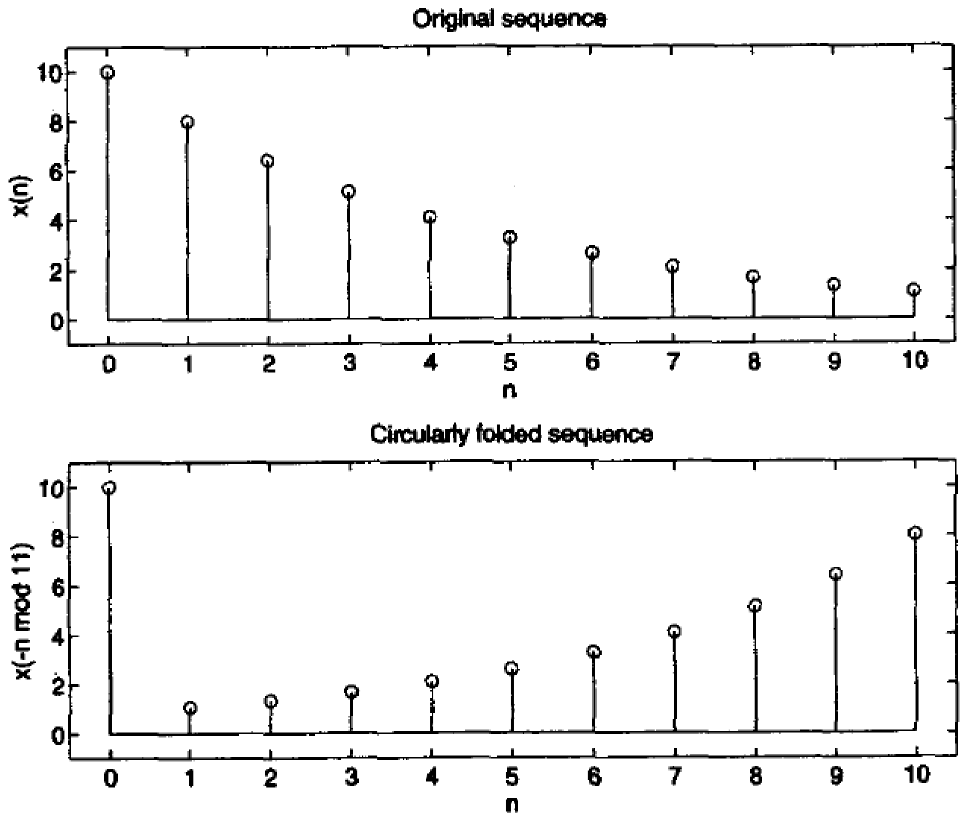
- EX2: Verify that DFT(ax+by)=aX+bY=Z and that ax+by = IDFT(Z).

Exercise 2: (2/24)

```
a = 0.2;                %Mix coefficients
b = 0.3;
x = ones(100,1);        %x sequence
y = triang(100);        %y sequence

z1 = a*x+b*y;           %Mix signal in temporal domain

X = dft(x);
Y = dft(y);

Z2 = a*X+b*Y;           %Mix signal in frequency domain
Z = dft(z1);
error=sum(abs(Z)-abs(Z2));

z2 = real(ifft(Z2));
error2 = sum(z1-z2);
```

Exercise 2

- PROPERTIES:



- 2. Circular folding:

$$DFT\{x((-n))_N\} = X((-k))_N$$

Where:

$$x((-n))_N = \begin{cases} x(0) & n = 0 \\ x(N-n) & 1 \le n \le N-1 \end{cases}$$

Exercise 2 (4/24)

EXAMPLE 5.9 Let $x(n) = 10\,(0.8)^n, \quad 0 \leq n \leq 10.$

a. Determine and plot $x\left((-n)\right)_{11}$.
b. Verify the circular folding property.

Matlab provides the function:

"k=mod(n,N)" where $k = n - floor\left(\frac{n}{N}\right).*N$

It could be used to obtain the circular foding:

x_fold = x(mod(-n,N+1)+1);

Exercise 2 (5/24) $k = n - floor\left(\frac{n}{N}\right).*N$

It could be used to obtain the circular folding:

x_fold = x(mod(-n,N)+1);

n = 0 1 2 3 4 5

N = length(n) = 6

-n = 0 -1 -2 -3 -4 -5

floor(-n/6) = 0 -1 -1 -1 -1 -1

floor(-n/6.*6) = 0 -6 -6 -6 -6 -6

-n – (floor(-n/6).*6) = 0 5 4 3 2 1

% index in Matlab start from 1

(mod(-n,N)+1) = 1 6 5 4 3 2

Exercise 2 (6/24)

- So:
  - n = 0 1 2 3 4 5 6 7 8 9 10
  - N = length(n)
  - x = 10*(0.8) .^ n

    = 10.0000 8.0000 6.4000 5.1200 4.0960 3.2768 2.6214 2.0972 1.6777 1.3422 1.0737

  - y = x(mod(-n,N)+1)

    = 10.0000 1.0737 1.3422 1.6777 2.0972 2.6214 3.2768 4.0960 5.1200 6.4000 8.0000

Exercise 2 (7/24)

- PROPERTIES:
  - 3. Conjugation: Circular folding in the frequency domain:

$$DFT\left\{x^*(n)\right\} = X^*\left((-k)\right)_N$$

  - Example 5.9 bis: Let x(n)=exp(j*π*3*n) n=0,...,N
    - Determine and plot $X^*\left((-k)\right)_N$
    - Determine and plot $DFT\left\{x^*(n)\right\}$
    - Verify the property

Exercise 2 (8/24)

- PROPERTIES:
- 4. Symmetry properties for real sequences: if x(n) is real, then $$X(k) = X^*\big((-k)\big)_N$$
- It implies that:

$$\mathrm{Re}\{X(k)\} = \mathrm{Re}\{X\big((-k)\big)_N\}$$

$$\mathrm{Im}\{X(k)\} = -\,\mathrm{Im}\{X\big((-k)\big)_N\}$$

$$|X(k)| = \left|X\big((-k)\big)_N\right|$$

$$\angle X(k) = -\angle X\big((-k)\big)_N$$

Exercise 2 (9/24)

Thanks to this symmetry property,

- one need to compute X(k) only for

$$k = 0,1,\ldots,\frac{N}{2} \quad if\ N\, is\, even$$

$$k = 0,1,\ldots,\frac{N-1}{2} \quad if\ N\, is\, odd$$

- The DFT coefficient at k=0 must be a real number.
- If N is even, the DFT coefficient at N/2 (Nyquist component) must be a real number.

Exercise 2 (10/24)

- Even and odd components: x(n) real:

$$x(n) = x_{ec}(n) + x_{oc}(n)$$

$$x_{ec}(n) = \frac{1}{2}\left[x(n) + x\left((-n)\right)_N\right] \quad x_{oc}(n) = \frac{1}{2}\left[x(n) - x\left((-n)\right)_N\right]$$

  - Then:

$$DFT\{x_{ec}(n)\} = \mathrm{Re}\{X(k)\} = \mathrm{Re}\{X\left((-k)\right)_N\}$$

$$DFT\{x_{oc}(n)\} = \mathrm{Im}\{X(k)\} = \mathrm{Im}\{X\left((-k)\right)_N\}$$

Exercise 2 (11/24)

EXAMPLE 5.10 Let $x(n) = 10\,(0.8)^n, \quad 0 \le n \le 10$ as in Example 5.9.

a. Decompose and plot the $x_{ec}(n)$ and $x_{oc}(n)$ components of $x(n)$.

- Design a function “circevod” that computes the even and odd components of the signal:
  - **[xec, xoc] = circevod(x)**

Exercise 2 (12/24)

- PROPERTIES:
  - 5. Circular shift of a sequence:

$$\tilde{x}(n-m) = x\left((n-m)\right)_N R_N(n)$$

  - Its DFT is given by:

$$DFT\left\{x\left((n-m)\right)_N R_N(n)\right\} = W_N^{km} X(k)$$

Exercise 2 (13/24)

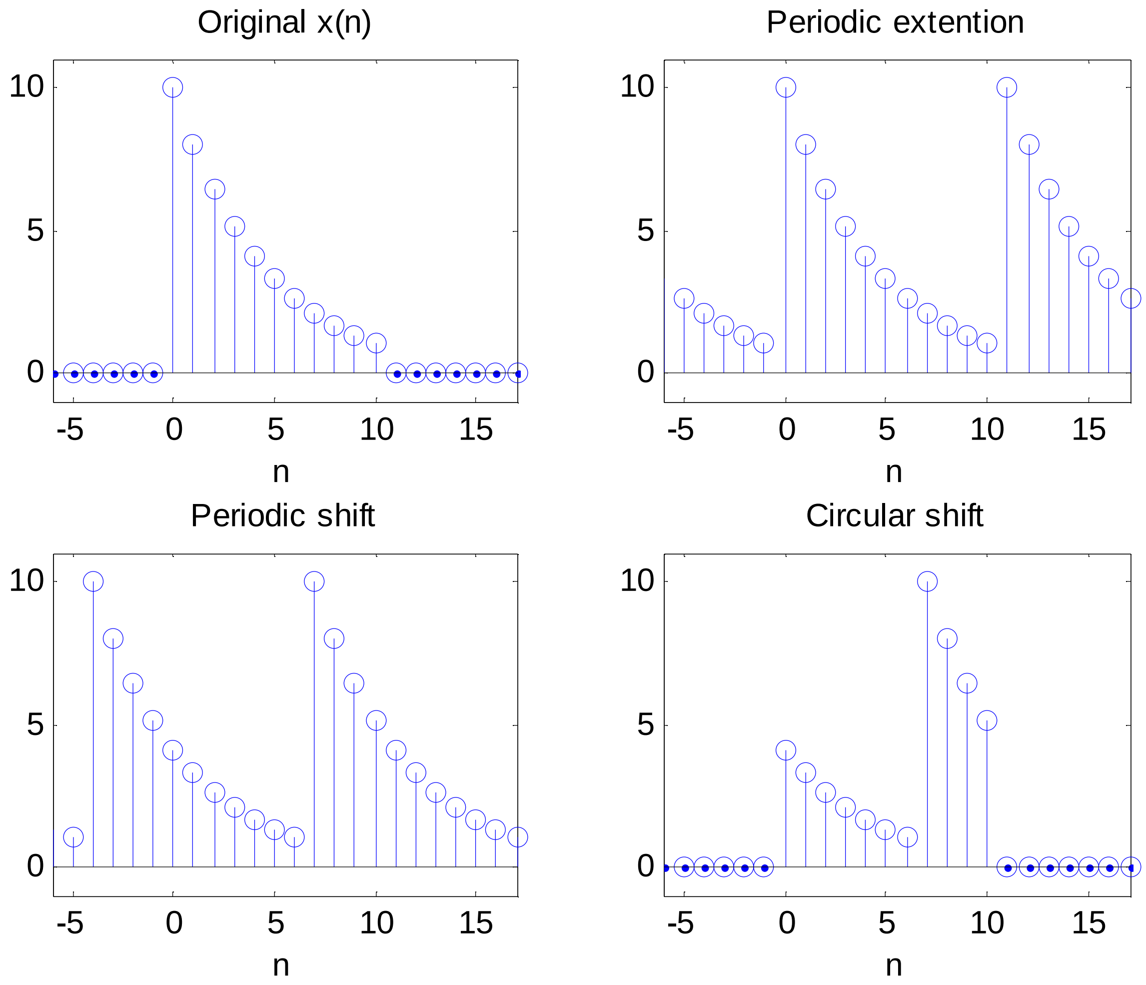
- To obtain a circular shift of a sequence we have to:
  - convert x(n) into its periodic extension $\tilde{x}$ with period N
  - shift it by m samples $x((n-m))_N$
  - convert into a N point sequence $x((n-m))_N R_N(n)$

EXAMPLE 5.11 Let $x(n) = 10\,(0.8)^n, \quad 0 \le n \le 10$ be an 11-point sequence.

a. Sketch $x\,((n+4))_{11}\,R_{11}(n)$, that is, a circular shift by 4 samples toward the left.

b. Sketch $x\,((n-3))_{15}\,R_{15}(n)$, that is, a circular shift by 3 samples toward the right, where $x(n)$ is assumed to be a 15-point sequence.

Exercise 2 (14/24)



Exercise 2 (15/24)

EXAMPLE 5.12 Given an 11-point sequence $x(n) = 10\,(0.8)^n$, $\quad 0 \le n \le 10$, determine and plot $x\left((n-6)\right)_{15}$.

- Design the function “circshift(x,m,N)” that computes the circular shift of x by m samples modulus N
  - x = [x zeros(1,N-length(x))];
  - n = [0:1:N-1];
  - n = mod(n-m,N);
  - y = x(n+1);

Exercise 2 (16/24)

- Ex.5.12: Verify the property.

$$DFT\{x((n-m))_N R_N(n)\} = W_N^{km} X(k)$$

$$W_N^{km} = e^{-j\left(2\pi/N\right)km}$$

Exercise 2 (17/24)

- PROPERTIES:
  - 6. Circular convolution:

$$x_1(n) \otimes_N x_2(n) = \sum_{m=0}^{N-1} x_1(m)\, x_2\left((n-m)\right)_N \qquad 0 \le n \le N-1$$

  - Its DFT is given by:

$$DFT\left\{x_1(n) \otimes_N x_2(n)\right\} = X_1(k)\, X_2(k)$$

Exercise 2 (18/24)

EXAMPLE 5.13 Let $x_1(n) = \{\underset{\uparrow}{1}, 2, 2\}$ and $x_2(n) = \{1, 2, 3, 4\}$. Compute the 4-point circular convolution $x_1(n)$ ④ $x_2(n)$.

- time domain:

$$x_1(n) \otimes_4 x_2(n) = \sum_{m=0}^{3} x_1(m)\ x_2\left((n-m)\right)_4 \qquad 0 \le n \le 3$$

- x1(n) = [1 2 2 0]
- x2(n) = [1 2 3 4]
- x2(-n) = [1 4 3 2]

for $n = 0$

$$\sum_{m=0}^{3} x_1(m) \cdot x_2((0-m))_5 = \sum_{m=0}^{3} [\{1,\ 2,\ 2,\ 0\} \cdot \{1,\ 4,\ 3,\ 2\}]$$

$$= \sum_{m=0}^{3} \{1,\ 8,\ 6,\ 0\} = 15$$

for $n = 1$

$$\sum_{m=0}^{3} x_1(m) \cdot x_2((1-m))_5 = \sum_{m=0}^{3} [\{1,\ 2,\ 2,\ 0\} \cdot \{2,\ 1,\ 4,\ 3\}]$$

$$= \sum_{m=0}^{3} \{2,\ 2,\ 8,\ 0\} = 12$$

for $n = 2$

$$\sum_{m=0}^{3} x_1(m) \cdot x_2\left((2-m)\right)_5 = \sum_{m=0}^{3} [\{1,\ 2,\ 2,\ 0\} \cdot \{3,\ 2,\ 1,\ 4\}]$$

$$= \sum_{m=0}^{3} \{3,\ 4,\ 2,\ 0\} = 9$$

for $n = 3$

$$\sum_{m=0}^{3} x_1(m) \cdot x_2\left((3-m)\right)_5 = \sum_{m=0}^{3} [\{1,\ 2,\ 2,\ 0\} \cdot \{4,\ 3,\ 2,\ 1\}]$$

$$= \sum_{m=0}^{3} \{4,\ 6,\ 4,\ 0\} = 14$$

Hence

$$x_1(n)\ (4)\ x_2(n) = \{15,\ 12,\ 9,\ 14\}$$

Exercise 2 (21/24)

- time domain:

$$x_1(n) \otimes_4 x_2(n) = \sum_{m=0}^{3} x_1(m)\ x_2\big((n-m)\big)_4 \quad 0 \le n \le 3$$

- design the function "**circonvt(x1,x2,N)**" that computes the circular convolution in the time domain:
- ...
- H(n,:) = cirshftt(x2,n-1,N); % each row of H contain the sequence $x_2\big((n-m)\big)_4$ for a different value of n
- ...

Exercise 2 (22/24)

- frequency domain:

$$DFT\{x_1(n) \otimes_N x_2(n)\} = X_1(k)\,X_2(k)$$

  - X1 = fft(x1,N);
  - X2 = fft(x2,N);
  - Y = X1.*X2;
  - y = ifft(Y,4)

Exercise 2 (23/24)

EXAMPLE 5.15 In this example we will study the effect of N on the circular convolution. Obviously, $N \geq 4$; otherwise there will be a time-domain aliasing for $x_2(n)$. We will use the same two sequences from Example 5.13.

a. Compute $x_1(n) \textcircled{5} x_2(n)$.

b. Compute $x_1(n) \textcircled{6} x_2(n)$.

c. Comment on the results.

- [15, 12, 9, 14]
- a. [9, 4, 9, 14, 14]
- b. [1, 4, 9, 14, 14, 8] = linear convolution

Exercise 2 (24/24)

- PROPERTIES:
- 7. Multiplication:

$$DFT\{x_1(n)\cdot x_2(n)\} = \frac{1}{N} X_1(k) \otimes_N X_2(k)$$

- 8. Parseval's relation: $E_x = \sum_{n=0}^{N-1} |x(n)|^2 = \frac{1}{N}\sum_{k=0}^{N-1} |X(k)|^2$

$$\frac{|X(k)|^2}{N} = energy \quad spectrum$$

Summary:

- Ex3: Linear convolution
- Ex4: Block convolution

Exercise 3 (1/3)

- Let $x_1(n)$ be a N_1-point sequence
- Let $x_2(n)$ be a N_2-point sequence
- Then $x_3(n) = x_1(n) * x_2(n)$ is an N-point sequence with $N=N_1+N_2-1$
- If we make both x_1 and x_2 N-point sequences, then the circular convolution is identical to the linear convolution

Exercise 3 (2/3)

EXAMPLE 5.16 Let $x_1(n)$ and $x_2(n)$ be the two 4-point sequences given below.

$$x_1(n) = \{1,\ 2,\ 2,\ 1\}, \quad x_2(n) = \{1,\ -1,\ -1,\ 1\}$$

a. Determine their linear convolution $x_3(n)$.
b. Compute the circular convolution $x_4(n)$ so that it is equal to $x_3(n)$.

- Matlab provides the function "conv(x,y)" to compute the linear convolution (zero-padded convolution) between the sequences "x" and "y".
  - x = [1 2 2 1];
  - y = [1 -1 -1 1];
  - z = conv(x,y);
  - Length(z)=Nx+Ny-1

Exercise 3 (3/3)

- Compute the convolution in the frequencies domain:
  - Zero padding:
    - Add at the end of x1, Lx2-1 zeros
    - Add at the end of x2, Lx1-1 zeros
  - Compute the DFTs of the new sequences X1 and X2.

  - Multiply the DFTs sample by sample: X3=X1*X2

  - Antitrasform: x3=real(ifft(X3));

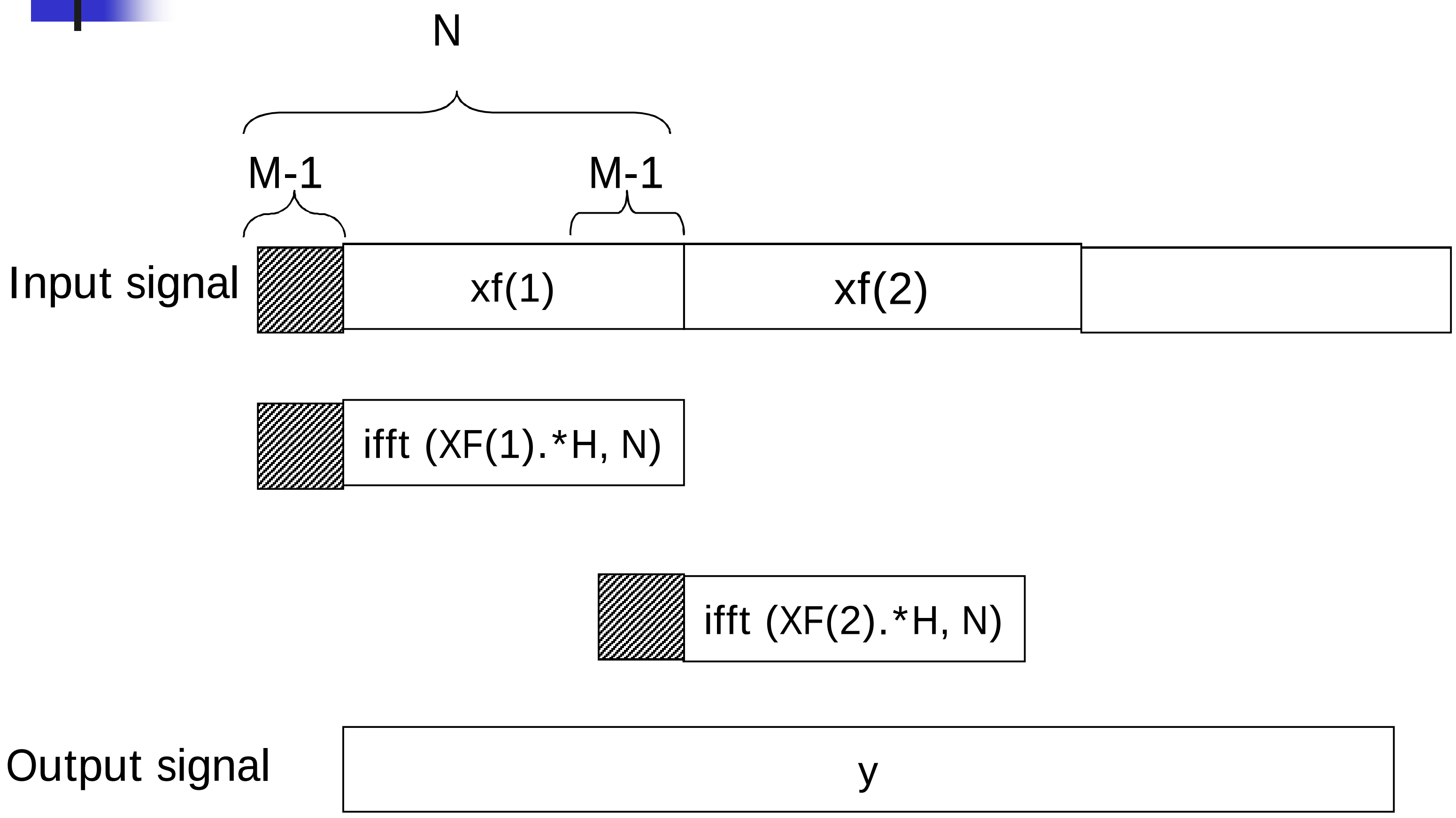
Exercise 4 (1/9)

- An error will be introduced when N is chosen less than the required value to perform a circular convolution that would be identical to the linear convolution.
- When $N=\max\{N_1,N_2\}$ is chosen for circular convolution, then the first M-1 samples are in error, where $M=\min\{N_1,N_2\}$.
- There are some situation where it will not be practical to perform the convolution of two signals using one DFT:
  - When l_x is extremely large (we can store the past $l_h - 1$ of the input signal x to calculate the next output. Unfortunately this procedure can be extremely time consuming when l_h is large).
  - In real time operation

Exercise 4 (2/9)

- We have to segment the infinite-length input sequence into smaller sections (blocks), process each section using the DFT, and finally assemble the output sequence from the output of each section → BLOCK CONVOLUTION.
- We can we can partition x(n) into sections, each overlapping with the previous one by exactly (M-1) samples, save the last (N-M+1) output samples, and finally concatenate these outputs into a sequence. To correct for the first (M-1) samples in the first output block, we set the first (M-1) samples in the first input block to zero → OVERLAP-SAVE method of block convolutions.

Exercise 4 (3/9)



Exercise 4 (4/9)

EXAMPLE 5.18 Let $x(n) = (n+1)$, $0 \leq n \leq 9$ and $h(n) = \{\underset{\uparrow}{1}, 0, -1\}$. Implement the overlap-save method using $N = 6$ to compute $y(n) = x(n) * h(n)$.

Since $M = 3$, we will have to overlap each section with the previous one by two samples. Now $x(n)$ is a 10-point sequence, and we will need $(M-1) = 2$ zeros in the beginning. Since $N = 6$, we will need 3 sections. Let the sections be

$$x_1(n) = \{0, 0, 1, 2, 3, 4\}$$

$$x_2(n) = \{3, 4, 5, 6, 7, 8\}$$

$$x_3(n) = \{7, 8, 9, 10, 0, 0\}$$

Exercise 4 (5/9)

Note that we have to pad $x_3(n)$ by two zeros since $x(n)$ runs out of values at $n = 9$. Now we will compute the 6-point circular convolution of each section with $h(n)$.

$$y_1 = x_1(n) \; ⑥ \; h(n) = \{-3, -4, 1, 2, 2, 2\}$$

$$y_2 = x_2(n) \; ⑥ \; h(n) = \{-4, -4, 2, 2, 2, 2\}$$

$$y_3 = x_3(n) \; ⑥ \; h(n) = \{7, 8, 2, 2, -9, -10\}$$

Noting that the first two samples are to be discarded, we assemble the output $y(n)$ as

$$y(n) = \{\underset{\uparrow}{1}, 2, 2, 2, 2, 2, 2, 2, 2, 2, -9, -10\}$$

Exercise 4 (6/9)

- Set the lengths:
  - Lx = length(x);   M = length(h);
  - M1 = M-1;
  - L = N-M1;   % number of useful samples for each block
  - h = [h zeros(1,N-M)];   % length N samples
  - x = [zeros(1,M1), x, zeros(1,N-1)]; % preappend (M-1) zeros and postappend zeros for not going out of values.
  - K = floor((Lenx+M1-1)/(L));   % # of blocks

Exercise 4 (7/9)

- N.B. Calling x(n) the zero padded sequence, the kth block is given by:

$$x_k(n) = x(m) \; kL \le m \le kL + N - 1$$

- Segment the input waveform x(n) into overlapping frames of length N.
  - for k=0:K
  - xk = x(k*L+1:k*L+N);
  - ...
  - end

Exercise 4 (8/9)

- For each input block:
    - for k=0:K
  - Extract the current input block of samples
  - Shift it in to the base time interval
    - xk = x(k*L+1:k*L+N);
  - Filter the signal in the frequency domain
    - Y(k+1,:) = circonvt(xk,h,N);
  - Y will be an K+1xN matrix
  - Discard the first M-1 samples for each row
    - Y = Y(M:N,:);
  - Put together all the row and asseble the output
    - Y = Y’; y = (Y(:))';

Exercise 4 (9/9)

- Compare the output obtained by means of the overlap&save algorithm (in time and frequency) with the one obtained filtering by means of the "conv" function:
  - n = 0:9; x = n+1; h = [1,0,-1]; N = 6;
  - y = ovrlpsav(x,h,N)
  - y1 = ovrlpsavf(x,h,N)
  - y2 = conv(x,h)