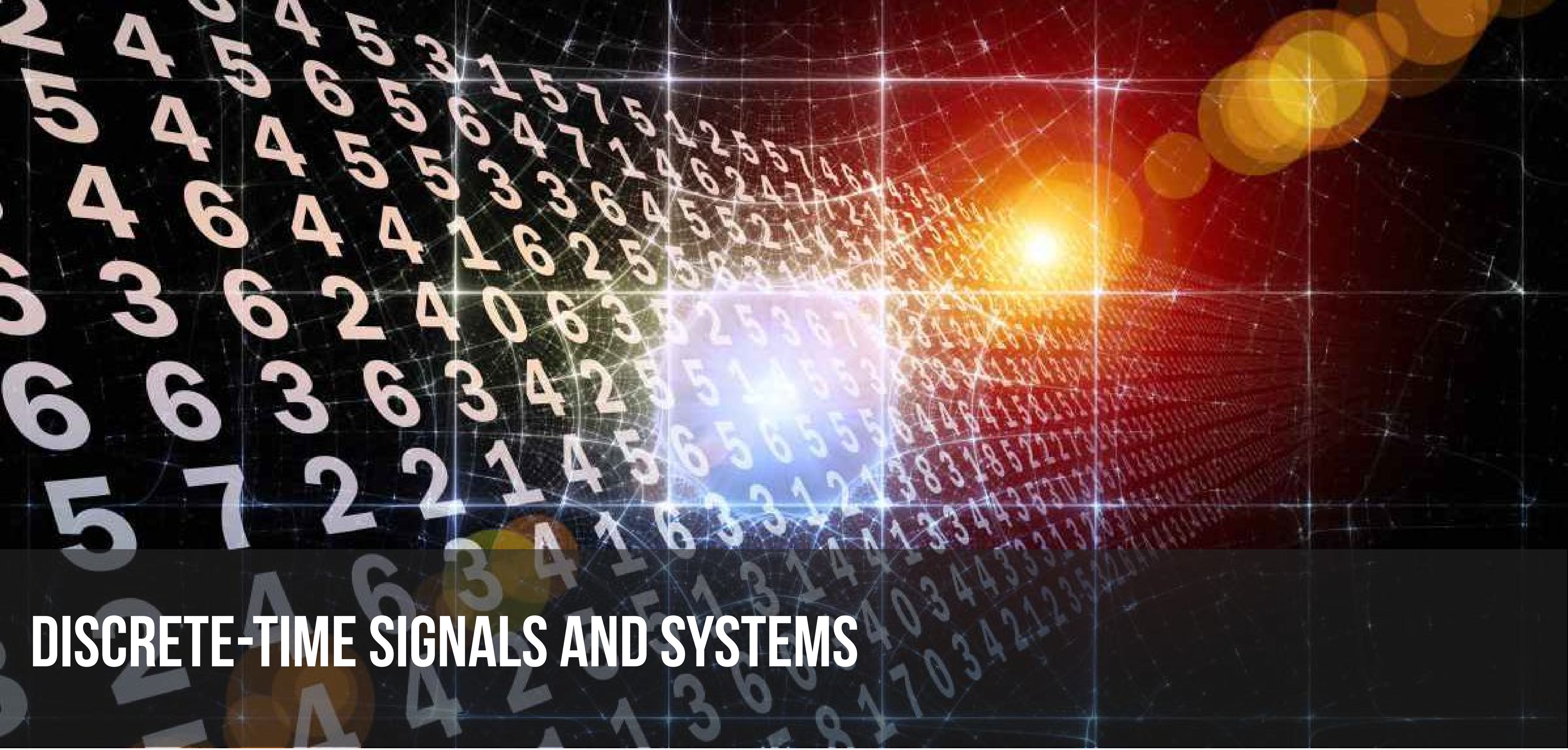
PENGOLAHAN SINYAL DIGITAL

Adhi Harmoko Saputro

UNIVERSITAS INDONESIA
Veritas, Probitas, Iustitia



DISCRETE-TIME SIGNALS AND SYSTEMS

Adhi Harmoko Saputro

UNIVERSITAS INDONESIA
Veritas, Probitas, Iustitia

OPERATIONS ON SEQUENCES

- **Signal addition**: $$\{x_1[n]\}+\{x_2[n]\}=\{x_1[n]+x_2[n]\}$$

```
function [y,n] = sigadd(x1,n1,x2,n2)
% implements y(n) = x1(n)+x2(n)
% -----------------------------
% [y,n] = sigadd(x1,n1,x2,n2)
% y = sum sequence over n, which includes n1 and n2
% x1 = first sequence over n1
% x2 = second sequence over n2 (n2 can be different from n1)
%
n = min(min(n1),min(n2)):max(max(n1),max(n2));
% duration of y(n)
y1 = zeros(1,length(n)); y2 = y1;
% initialization
y1(find((n>=min(n1))&(n<=max(n1))==1))=x1;
% x1 with duration of y
y2(find((n>=min(n2))&(n<=max(n2))==1))=x2;
% x2 with duration of y
y = y1+y2; % sequence addition
```

OPERATIONS ON SEQUENCES

- **Signal multiplication:** $$\{x_1[n]\}.\{x_2[n]\} = \{x_1[n]x_2[n]\}$$

```
function [y,n] = sigmult(x1,n1,x2,n2)
% implements y(n) = x1(n)*x2(n)
% -----------------------------
% [y,n] = sigmult(x1,n1,x2,n2)
% y = product sequence over n, which includes n1 and n2
% x1 = first sequence over n1
% x2 = second sequence over n2 (n2 can be different from n1)
n = min(min(n1),min(n2)):max(max(n1),max(n2));
% duration of y(n)
y1 = zeros(1,length(n)); y2 = y1; %
y1(find((n>=min(n1))&(n<=max(n1))==1))=x1;
% x1 with duration of y
y2(find((n>=min(n2))&(n<=max(n2))==1))=x2;
% x2 with duration of y
y = y1 .* y2; % sequence multiplication
```

OPERATIONS ON SEQUENCES

- **Scaling:**

$$\alpha\{x[n]\}=\{\alpha x[n]\}$$

OPERATIONS ON SEQUENCES

- **Shifting**: each sample of $x[n]$ is shifted by an amount k to obtain a shifted sequence $y[n]$.

$$y[n] = \{x[n-k]\}$$

  - If $m = n-k$, then $n = m+k$ and the above operation is given by

$$y[m+k] = \{x[m]\}$$

```
function [y,n] = sigshift(x,m,k)
% implements y(n) = x(n-k)
% -------------------------
% [y,n] = sigshift(x,m,k)
%
n = m+k; y = x;
```

OPERATIONS ON SEQUENCES

- **Folding**: In this operation each sample of *x*(*n*) is flipped around *n* = 0 to obtain a folded sequence *y*(*n*).

$$y[n]=\{x[-n]\}$$

```
function [y,n] = sigfold(x,n)
% implements y(n) = x(-n)
% -----------------------
% [y,n] = sigfold(x,n)
%
y = fliplr(x); n = -fliplr(n);
```

- **Sample summation**: This operation differs from signal addition operation. It adds all sample values of *x*(*n*) between n_1 and n_2.

$$\sum_{n=n_1}^{n_2} x[n]=x[n_1]+\ldots+x[n_2]$$

OPERATIONS ON SEQUENCES

- **Sample products**: This operation also differs from signal multiplication operation.
  - It multiplies all sample values of *x*(*n*) between n_1 and n_2.

$$\prod_{n_1}^{n_2} x[n] = x[n_1] \times \ldots \times x[n_2]$$

- **Signal energy**: The energy of a sequence *x*(*n*) is given by

$$\varepsilon_x = \sum_{-\infty}^{\infty} x[n]x^*[n] = \sum_{-\infty}^{\infty} \left| x[n] \right|^2$$

OPERATIONS ON SEQUENCES

- **Signal power**: The average power of a periodic sequence *x*[*n*] with fundamental period *N* is given by

$$P_x = \frac{1}{N}\sum_{0}^{N-1}\left|\tilde{x}\left[n\right]\right|^2$$

MATLAB EXAMPLE 1

- Generate and plot each of the following sequences over the indicated interval

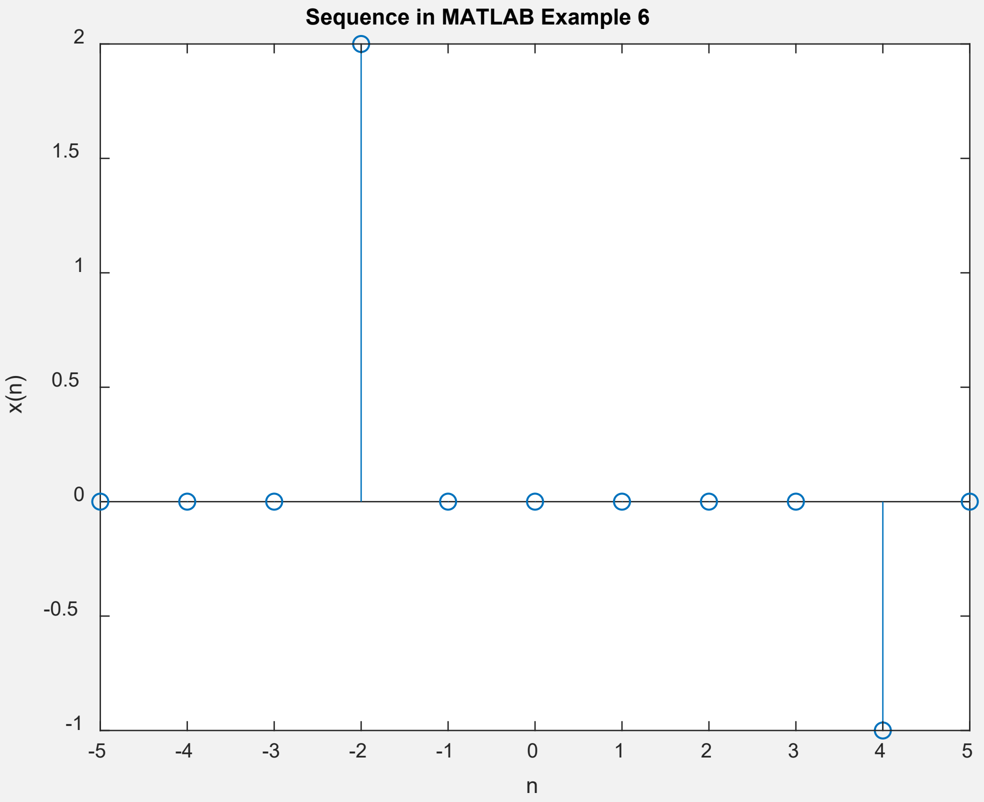
$$x(n) = 2\delta(n+2) - \delta(n-4),\ -5 \leq n \leq 5$$

```
n = -5:5;
x = 2*impseq(-2,-5,5) - impseq(4,-5,5);
stem(n,x); title('Sequence in MATLAB Example 1')
xlabel('n'); ylabel('x(n)');
```

MATLAB EXAMPLE 1

- Generate and plot each of the following sequences over the indicated interval

$$x(n) = 2\delta(n+2) - \delta(n-4),\ -5 \le n \le 5$$



MATLAB EXAMPLE 2

- Generate and plot each of the following sequences over the indicated interval

$$x(n) = n[u(n)-u(n-10)]+10e^{-0.3(n-10)}[u(n-10)-u(n-20)],$$
$$0 \le n \le 20$$

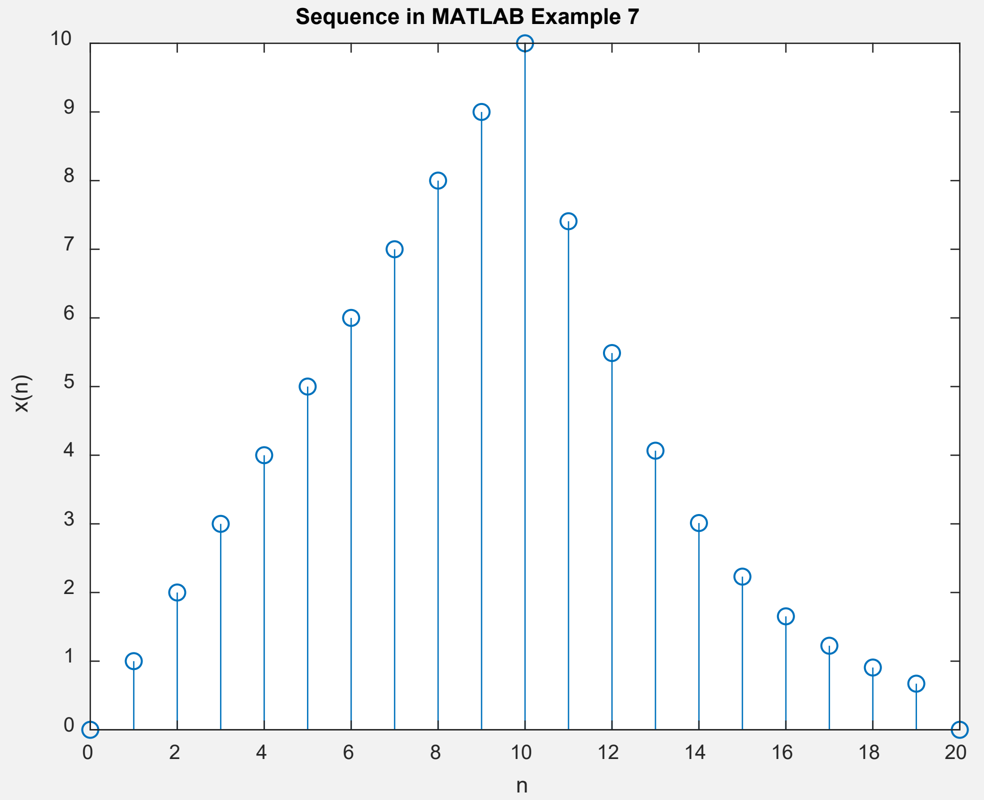
```
n = [0:20];
x1 = n.*(stepseq(0,0,20)-stepseq(10,0,20));
x2 = 10*exp(-0.3*(n-10)).*(stepseq(10,0,20)- ...
        stepseq(20,0,20));
x = x1+x2;
stem(n,x);
title('Sequence in MATLAB Example 2')
xlabel('n'); ylabel('x(n)');
```

MATLAB EXAMPLE 2

- Generate and plot each of the following sequences over the indicated interval

$$x(n) = n[u(n)-u(n-10)]+10e^{-0.3(n-10)}[u(n-10)-u(n-20)],$$

$$0 \leq n \leq 20$$



MATLAB EXAMPLE 3

- Generate and plot each of the following sequences over the indicated interval

$$x(n) = \cos(0.04\pi n) + 0.2w(n),\ 0 \le n \le 50$$

w(*n*) is a Gaussian random sequence with zero mean and unit variance

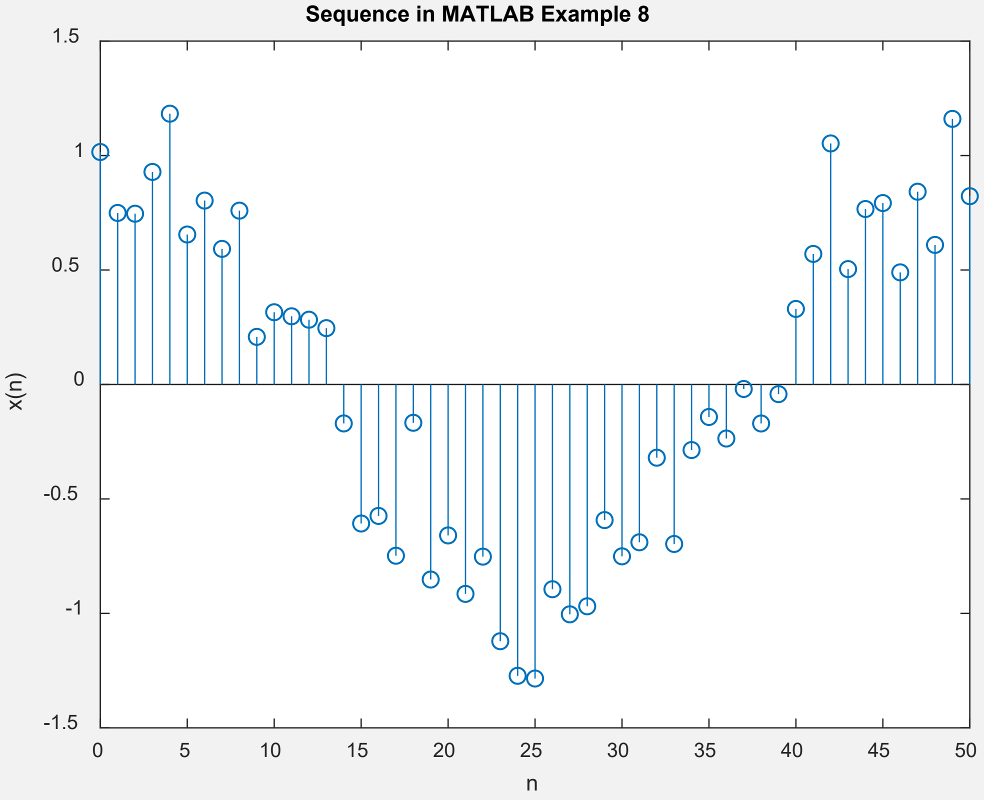
```
n = 0:50;
x = cos(0.04*pi*n)+0.2*randn(size(n));
stem(n,x);

title('Sequence in MATLAB Example 3')
xlabel('n'); ylabel('x(n)');
```

MATLAB EXAMPLE 3

- Generate and plot each of the following sequences over the indicated interval

$$x(n) = \cos(0.04\pi n) + 0.2w(n),\ 0 \leq n \leq 50$$

w(*n*) is a Gaussian random sequence with zero mean and unit variance



MATLAB EXAMPLE 4

- Generate and plot each of the following sequences over the indicated interval

$$\tilde{x}[n] = \{\ldots,5,4,3,2,1,\underset{\uparrow}{5},4,3,2,1,5,4,3,2,1,\ldots\}; \quad -10 \le n \le 9$$

- Note that over the given interval, the sequence $\tilde{x}(n)$ has four periods

```
n = -10:9;
x = [5,4,3,2,1];
xtilde = x' * ones(1,4);
xtilde = (xtilde(:))';
stem(n,xtilde);
title('Sequence in MATLAB Example 4')
xlabel('n'); ylabel('xtilde(n)');
```

MATLAB EXAMPLE 5

- Let

$$x[n] = \{1, 2, \underset{\uparrow}{3}, 4, 5, 6, 7, 6, 5, 4, 3, 2, 1\}$$

- Determine and plot the following sequences.
  - $x_1[n] = 2x(n-5) - 3x(n+4)$
  - $x_2[n] = x(3-n) + x(n)\,x(n-2)$

MATLAB EXAMPLE 5

$$x[n] = \{1, 2, \underset{\uparrow}{3}, 4, 5, 6, 7, 6, 5, 4, 3, 2, 1\}$$

- The sequence $x(n)$ is nonzero over $-2 \le n \le 10$.

```
n = -2:10; x = [1:7,6:-1:1];
```

MATLAB EXAMPLE 5A

- $x_1[n] = 2x[n-5] - 3x[n+4]$
- The first part is obtained by shifting $x[n]$ by 5 and the second part by shifting $x[n]$ by −4.
- This shifting and the addition can be easily done using the sigshift and the sigadd functions.

```
[x11,n11] = sigshift(x,n,5);
[x12,n12] = sigshift(x,n,-4);
[x1,n1] = sigadd(2*x11,n11,-3*x12,n12);
stem(n1,x1);
title('Sequence in MATLAB Example 5A')
xlabel('n'); ylabel('x1(n)');
```

MATLAB EXAMPLE 5B

- $x_2[n] = x[3 - n] + x[n]\, x[n - 2]$
- The first term can be written as $x[-[n - 3]]$. Hence it is obtained by first folding $x[n]$ and then shifting the result by 3.
- The second part is a multiplication of $x[n]$ and $x[n-2]$, both of which have the same length but different support (or sample positions).

MATLAB EXAMPLE 5B

- $x_2[n] = x[3 - n] + x[n]\, x[n - 2]$
- These operations can be easily done using the sigfold and the sigmult functions

```
[x21,n21] = sigfold(x,n);
[x21,n21] = sigshift(x21,n21,3);
[x22,n22] = sigshift(x,n,2);
[x22,n22] = sigmult(x,n,x22,n22);
[x2,n2] = sigadd(x21,n21,x22,n22);
stem(n2,x2);
title('Sequence in MATLAB Example 5B')
xlabel('n'); ylabel('x2(n)');
```

MATLAB EXAMPLE 6

- Generate the complex-valued signal

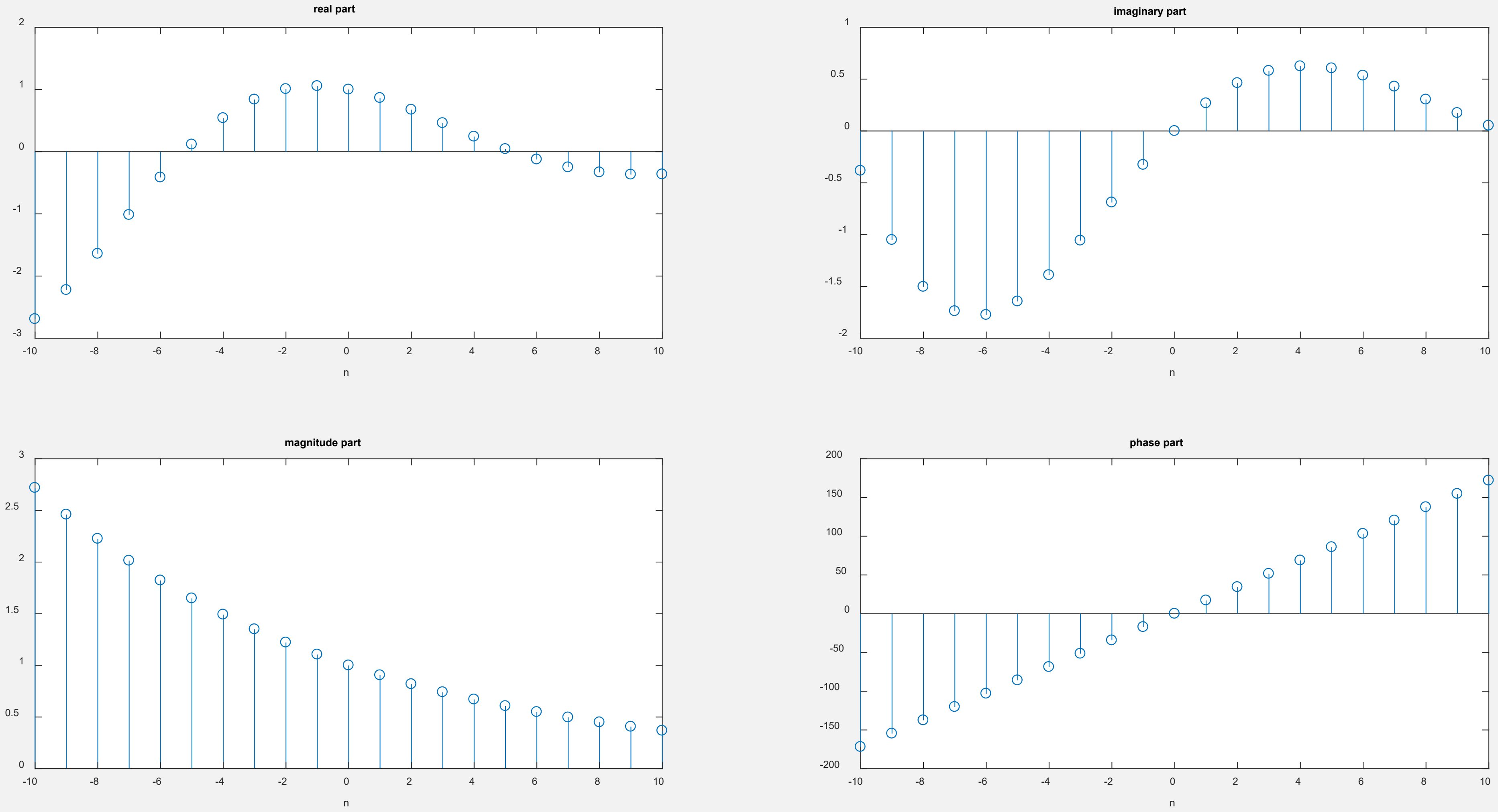
$$x[n] = e^{[-0.1+j0.3]n}, \ -10 \leq n \leq 10$$

- and plot its magnitude, phase, the real part, and the imaginary part in four separate subplots.

```
n = [-10:1:10]; alpha = -0.1+0.3j;
x = exp(alpha*n);
subplot(2,2,1); stem(n,real(x));
title('real part');xlabel('n')
subplot(2,2,2); stem(n,imag(x));
title('imaginary part');xlabel('n')
subplot(2,2,3); stem(n,abs(x));
title('magnitude part');xlabel('n')
subplot(2,2,4); stem(n,(180/pi)*angle(x));
title('phase part');xlabel('n')
```

MATLAB EXAMPLE 6

$$x[n] = e^{[-0.1+j0.3]n}, \ -10 \le n \le 10$$



DISCRETE-TIME SINUSOIDAL

- Basis for Fourier transform and in system theory as a basis for steady-state analysis

$$x[n] = A\cos(\omega_o n + \theta_o)$$

- Conveniently related to the continuous-time sinusoid

$$x_a(t) = A\cos(\Omega_o t + \theta_o)$$

PERIODICITY IN TIME

- the sinusoidal sequence is periodic if

$$x[n+N] = A\cos(\omega_o n + \omega_o N + \theta) = A\cos(\omega_o n + \theta_o) = x[n]$$

- This is possible if and only if $\omega_o N = 2\pi k$, where k is an integer

UNIT SAMPLE SYNTHESIS

- Any arbitrary sequence $x[n]$ can be synthesized as a weighted sum of delayed and scaled unit sample sequences, such as

$$x[n]=\sum_{k=-\infty}^{\infty} x[k]\delta[n-k]$$

EVEN AND ODD SYNTHESIS

- A real-valued sequence $x_e[n]$ is called even (symmetric) if

$$x_e[-n] = x_e[n]$$

- The cosine signal is an even signal. All other even signals can be obtained with

$$x[n] = a_o + \sum_{n=1}^{\infty} a_n \cos(\omega_o n)$$

  - where a_o is the average value of the signal

EVEN AND ODD SYNTHESIS

- A real-valued sequence $x_o[n]$ is called odd (antisymmetric) if

$$x_o[-n] = -x_o[n]$$

- The sine signal is an odd signal. All other odd signals can be obtained with:

$$x[n] = \sum_{n=1}^{\infty} b_n \sin(\omega_o n)$$

EVEN AND ODD SYNTHESIS

- Then any arbitrary real-valued sequence $x[n]$ can be decomposed into its even and odd components

$$x[n] = x_e[n] + x_o[n]$$

- where the even and odd parts are given by

$$x_e[n] = \frac{1}{2}\left[x[n] + x[-n]\right] \qquad x_o[n] = \frac{1}{2}\left[x[n] - x[-n]\right]$$

HARMONICS FOURIER

- Periodic signals can be decomposed into sums of sine and cosine signals, according with the following expression, which is a Fourier series:

$$x[n] = a_o + \sum_{n=1}^{\infty} a_n \cos(\omega_o n) + \sum_{n=1}^{\infty} b_n \sin(\omega_o n)$$

  - where ω_o is the frequency (rad/s) of the periodic signal

EVEN AND ODD SYNTHESIS

- MATLAB function to decompose a given sequence into its even and odd components

```
function [xe, xo, m] = evenodd(x,n)
% Real signal decomposition into even and odd parts
% ---------------------------------------------------
% [xe, xo, m] = evenodd(x,n)
%
if any(imag(x) ~= 0)
error('x is not a real sequence')
end
m = -fliplr(n);
m1 = min([m,n]); m2 = max([m,n]); m = m1:m2;
nm = n(1)-m(1); n1 = 1:length(n);
x1 = zeros(1,length(m)); x1(n1+nm) = x; x = x1;
xe = 0.5*(x + fliplr(x)); xo = 0.5*(x - fliplr(x));
```

MATLAB EXAMPLE 7

- Let $x[n] = u[n] - u[n-10]$.
  - Decompose $x[n]$ into even and odd components.

MATLAB EXAMPLE 7

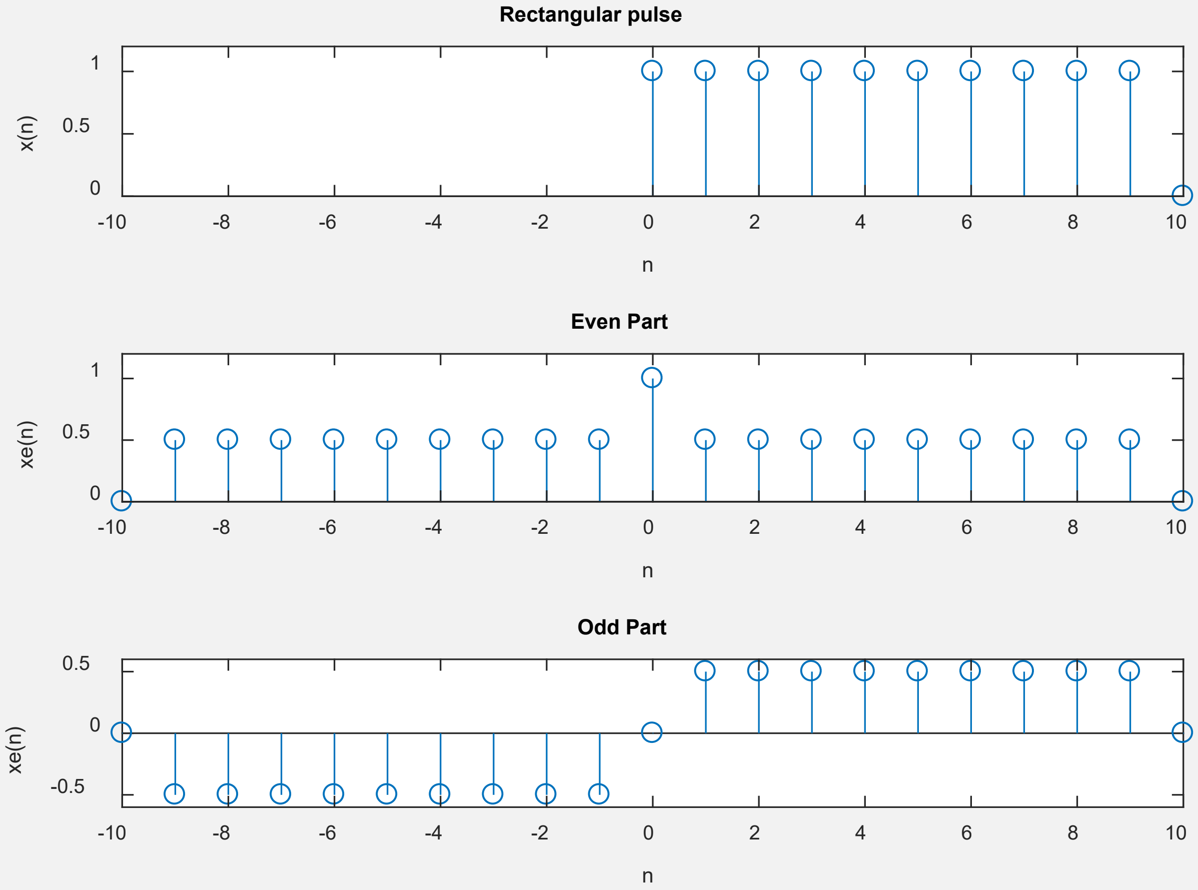
- Let $x[n] = u[n] - u[n-10]$.
  - Decompose $x[n]$ into even and odd components.
- The sequence $x[n]$, which is nonzero over $0 \le n \le 9$, is called a *rectangular pulse*

```
n = [0:10];
x = stepseq(0,0,10)-stepseq(10,0,10);
[xe,xo,m] = evenodd(x,n);
subplot(131); stem(n,x);
title(’Rectangular pulse’)
xlabel(’n’); ylabel(’x(n)’); axis([-10,10,0,1.2])
subplot(132); stem(m,xe); title(’Even Part’)
xlabel(’n’); ylabel(’xe(n)’);
axis([-10,10,0,1.2])
subplot(133); stem(m,xo); title(’Odd Part’)
xlabel(’n’); ylabel(’xe(n)’);
axis([-10,10,-0.6,0.6])
```

MATLAB EXAMPLE 7

- Let $x[n] = u[n] - u[n-10]$.
  - Decompose $x[n]$ into even and odd components.

- The sequence $x[n]$, which is nonzero over $0 \leq n \leq 9$, is called a *rectangular pulse*



THE GEOMETRIC SERIES

- A one-sided exponential sequence of the form

$$\{\ \alpha^n,\ n \geq 0\ \}$$

  - where α is an arbitrary constant

- The convergence and expression for the sum of this series are used in many applications

CORRELATIONS OF SEQUENCES

- Measure of the degree to which two sequences are similar
- the *crosscorrelation* of $x[n]$ and $y[n]$ is a sequence $r_{xy}[l]$ defined as

$$r_{xy}[\ell] = \sum_{n=-\infty}^{\infty} x[n]\,y[n-\ell]$$

TERIMA KASIH

Adhi Harmoko Saputro