EENG 479 : Digital Signal Processing (DSP)

Lecture #2:

2.1 Discrete time signal

Prof. Mohab A. Mangoud
Professor of Wireless Communications (Networks, IoT and AI)
University of Bahrain, College of Engineering
Department of Electrical and Electronics Engineering
P.O.Box 32038- Kingdom of Bahrain
mmangoud@uob.edu.bh
http://mangoud.com

Lecture # 2 outlines

2.1 Discrete time signal

2.1.1 Time domain representation √

2.1.2 Operation on sequences

- ex 2.1,2,3,4
- sampling rate alteration
- classification of sequences ex 2.5,6,7

2.2 Typical sequences and sequence representation

2.2.1 Some basic sequences ex 2.8, 9,10,

- 2.2.2 sequence generation using matlab (prog 2.2 and 2.3)
- 2.2.3 Representation of an arbitrary sequence

2.1 Discrete time signals

2.1.1 Time domain representation

Length of a discrete time signal

- Finite length
- Infinite length
- Appending with zeros (zero padding)
- Right sided sequence = causal sequence
- Left sided sequence = anti causal sequence
- Two-sided sequence

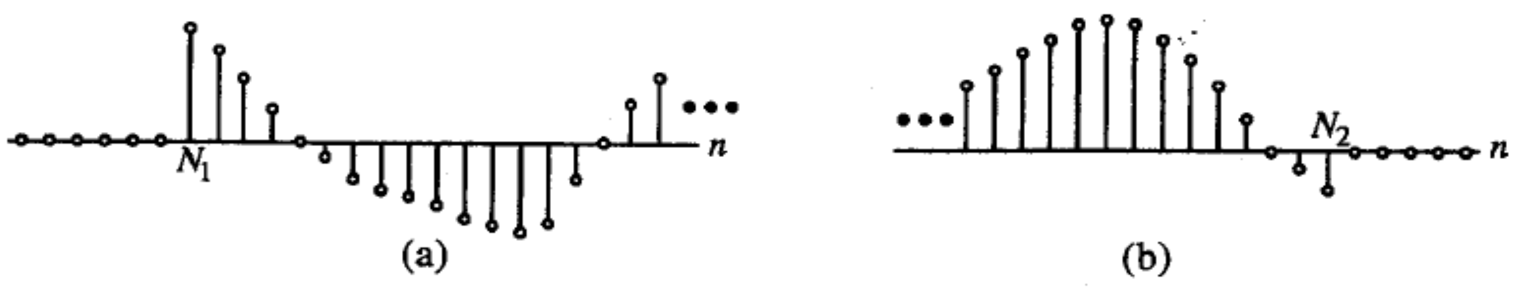


Figure 2.4: (a) A right-sided sequence and (b) a left-sided sequence.

The discrete-time signal may be a *finite-length* or an *infinite-length sequence*. A finite-length (also called *finite-duration* or *finite-extent*) sequence is defined only for a finite time interval:

$$N_1 \leq n \leq N_2, \tag{2.5}$$

where $-\infty < N_1$ and $N_2 < \infty$ with $N_2 \geq N_1$. The *length or duration N* of the above finite-length sequence is

$$N = N_2 - N_1 + 1. \tag{2.6}$$

A length-N discrete-time sequence consists of N samples and is often referred to as an N-point sequence. A finite-length sequence can also be considered as an infinite-length sequence by assigning zero values to samples whose arguments are outside the above range. The process of lengthening a sequence by adding zero-valued samples is called *appending with zeros* or *zero-padding*.

There are three types of infinite-length sequences. A *right-sided sequence* $x[n]$ has zero-valued samples for $n < N_1$; that is,

$$x[n] = 0 \qquad \text{for } n < N_1, \tag{2.7}$$

where N_1 is a finite integer that can be positive or negative. If $N_1 \geq 0$, a right-sided sequence is usually called a *causal sequence*.[2] Likewise, a *left-sided sequence* $x[n]$ has zero-valued samples for $n > N_2$; that is,

$$x[n] = 0 \qquad \text{for } n > N_2, \tag{2.8}$$

where N_2 is a finite integer that can be positive or negative. If $N_2 \leq 0$, a left-sided sequence is usually called an *anticausal sequence*. A general *two-sided sequence* is defined for both positive and negative values of n. Figure 2.4 illustrates the above two types of one-sided sequences.

Size of discrete time signal

L1-Norm : the mean absolute value

L2-norm : root mean squared (rms) value

L∞-norm : the peak absolute value

The norm of the finite length sequence can be computed using the M file *norm* in MATLAB

$$\| x \|_p = \left(\sum_{n=-\infty}^{\infty} |x[n]|^p \right)^{1/p},$$

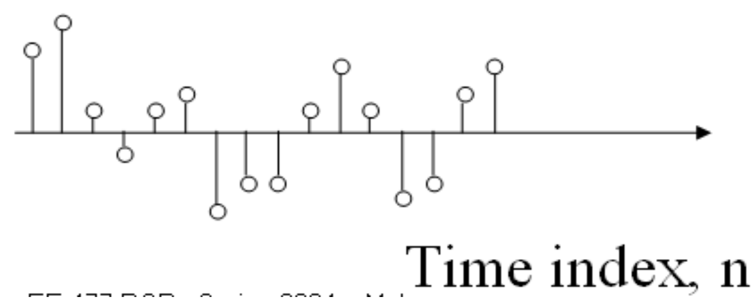
The norm provides an estimate of the size of a signal.

Sampling a Continuous Signal

- Obtain a sequence of signal *samples* using a periodic instantaneous sampler:

$$x[n] = x(nT_s)$$

- Often plot discrete signals as dots or "lollypops":



2.2.1 Some Basic Sequences

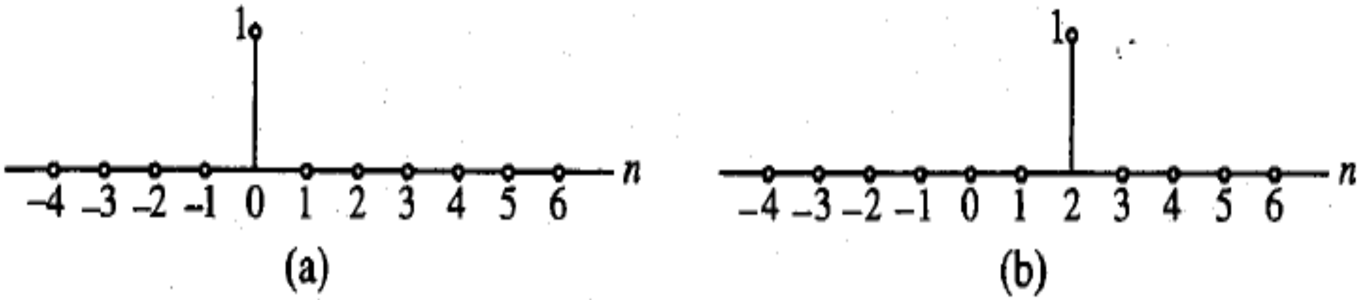


Figure 2.14: (a) The unit sample sequence $\{\delta[n]\}$ and (b) the shifted unit sample sequence $\{\delta[n-2]\}$.

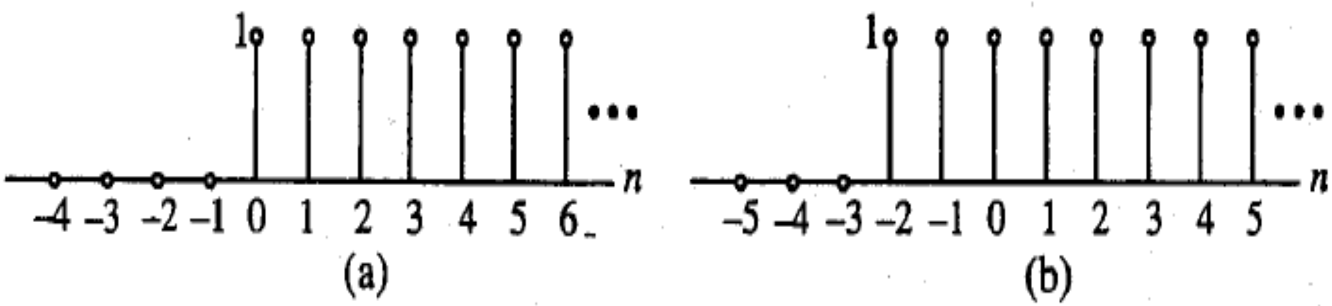


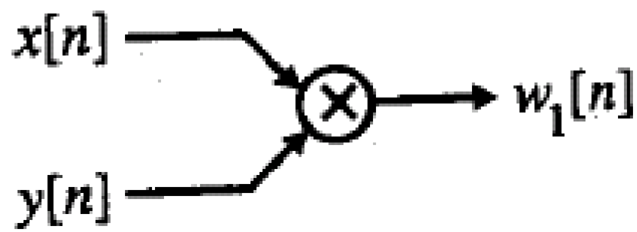
Figure 2.15: (a) The unit step sequence $\{\mu[n]\}$ and the shifted unit step sequence $\{\mu[n+2]\}$.

Elementary Operations

Let $x[n]$ and $y[n]$ be two known sequences. By forming the *product* of the sample values of these two sequences at each instant, we form a new sequence $w_1[n]$:

$$w_1[n] = x[n] \cdot y[n]. \tag{2.12}$$

In some applications, the product operation is also known as *modulation*. The device implementing the modulation operation is called a *modulator*, and its schematic representation is shown in Figure 2.5(a).



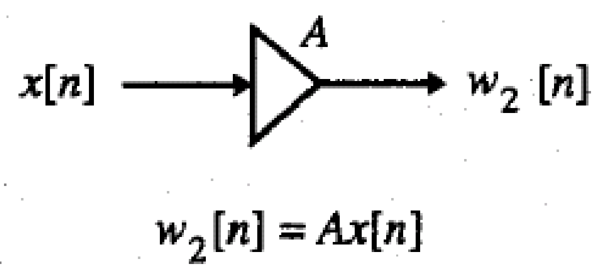
$$w_1[n] = x[n]y[n]$$

An application of the product operation is in forming a finite-length sequence from an infinite-length sequence by multiplying the latter with a finite-length sequence called *a window sequence.* This process of forming the finite-length sequence is usually called *windowing,* which plays an important role in the design of certain types of digital filters (Section 10.2). Another application of the product operation is illustrated in Example 2.10.

The second basic operation is the scalar *multiplication,* whereby a new sequence is generated by multiplying each sample of a sequence $x[n]$ by a scalar A:

$$w_2[n] = Ax[n]. \tag{2.13}$$

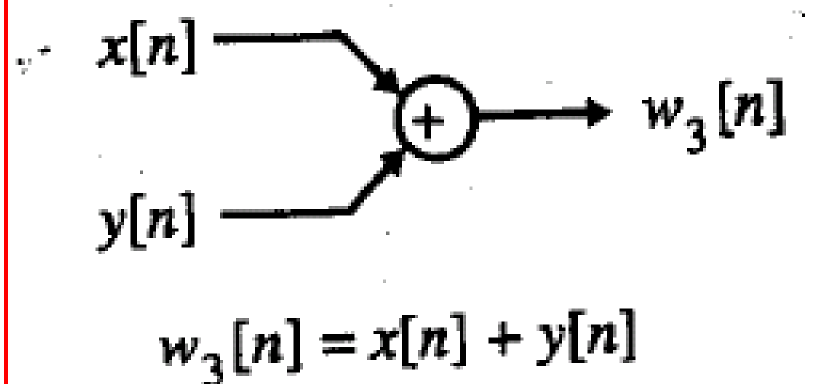
The device implementing the multiplication operation is called a *multiplier,* and its schematic representation is shown in Figure 2.5(b).



The third basic operation is the *addition* by which a new sequence $w_2[n]$ is obtained by adding the sample values of two sequences $x[n]$ and $y[n]$:

$$w_3[n] = x[n] + y[n]. \tag{2.14}$$

The device implementing the addition operation is called an *adder*, and its schematic representation is shown in Figure 2.5(c). By inverting the signs of all samples of the sequence $y[n]$, an adder can also be used to implement the *subtraction* operation.



A very simple application of the addition operation is in improving the quality of measured data that has been corrupted by an additive random noise. In many cases, the actual uncorrupted data vector $\mathbf{s}$ remains essentially the same from one measurement to the next, while the additive noise vector is random and not reproducible. Let $\mathbf{d}_i$ denote the noise vector corrupting the i-th measurement of the uncorrupted data vector $\mathbf{s}$:

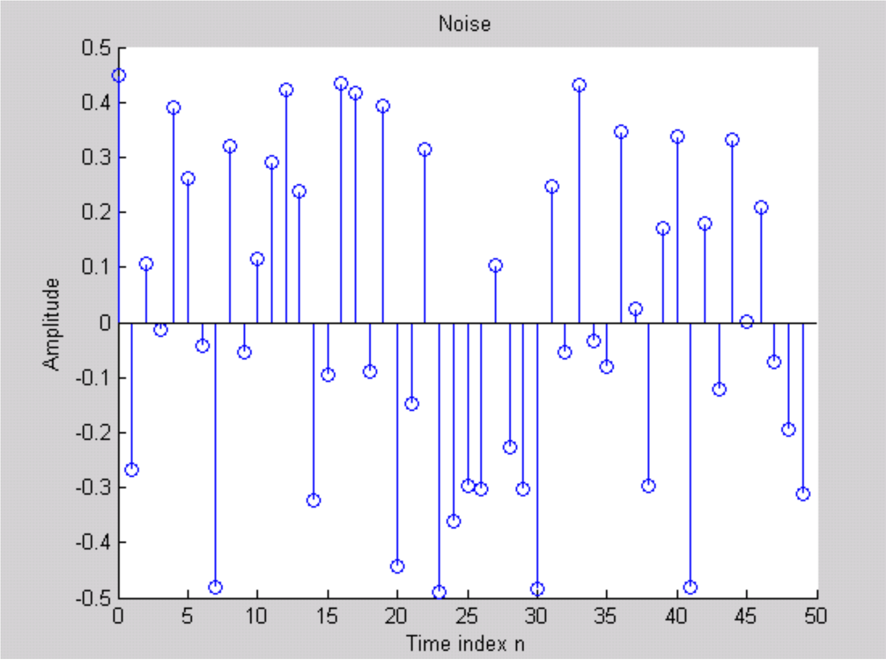
$$\mathbf{x}_i = \mathbf{s} + \mathbf{d}_i .$$

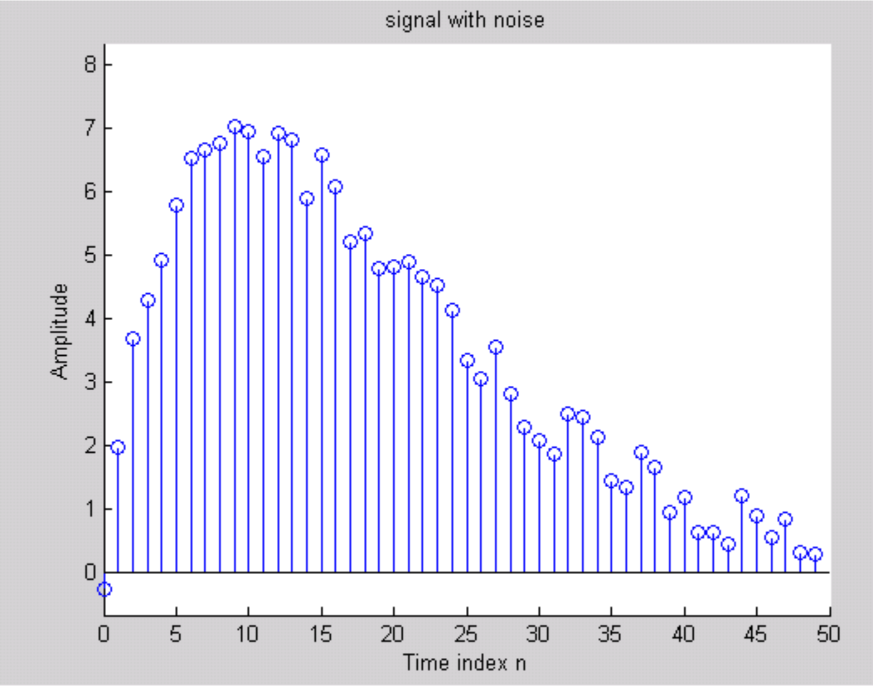
The average data vector, called the *ensemble average*, obtained after K measurements is then given by

$$\mathbf{x}_{\text{ave}} = \frac{1}{K}\sum_{i=1}^{K}(\mathbf{x}_i) = \frac{1}{K}\sum_{i=1}^{K}(\mathbf{s} + \mathbf{d}_i) = \mathbf{s} + \frac{1}{K}\left(\sum_{i=1}^{K}\mathbf{d}_i\right).$$

For a very large value of K, $\mathbf{x}_{\text{ave}}$ is usually a reasonable replica of the desired data vector $\mathbf{s}$, as the samples of the average of the summed noise vector $\frac{1}{K}(\sum_{i=1}^{K}\mathbf{d}_i)$ become very small due to the randomness of the noise. Example 2.1 illustrates ensemble averaging.

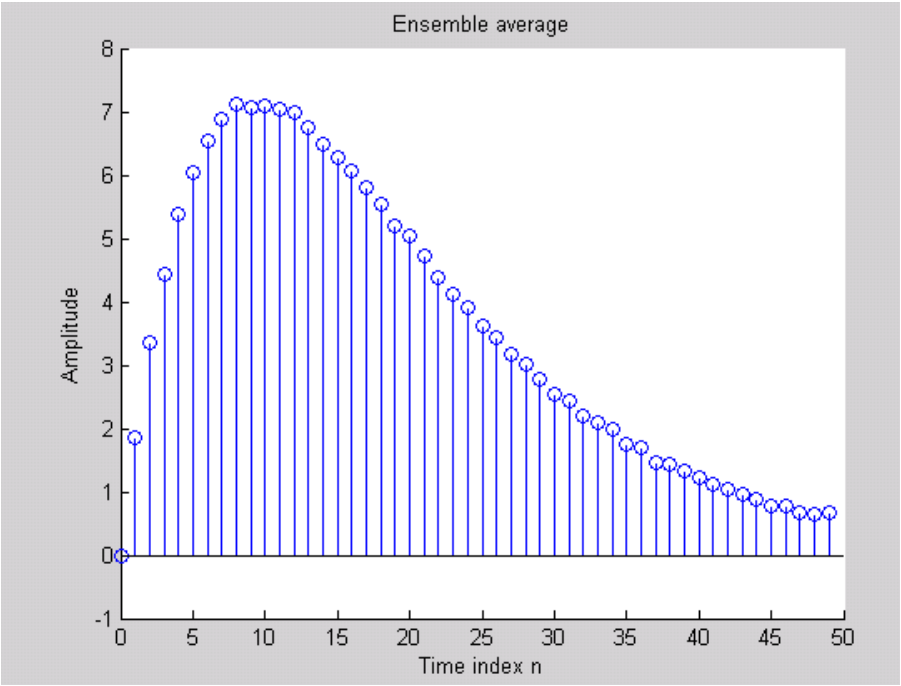
- % Program 2_1
- % Generation of the ensemble average
- %
- R = 50;
- m = 0:R-1;
- s = 2*m.*(0.9.^m);
- % Generate the uncorrupted signal
- d = rand(R,1)-0.5;
- % Generate the random noise
- x1 = s+d';
- stem(m,d);
- xlabel('Time index n');
- ylabel('Amplitude');
- title('Noise');
- Pause
- stem(m,x1);
- xlabel('Time index n');
- ylabel('Amplitude');
- title(‘Signal with Noise');
- pause





$$\mathbf{x}_{\text{ave}} = \frac{1}{K}\sum_{i=1}^{K}(\mathbf{x}_i)$$

```
for n = 1:50;
    d = rand(R,1)-0.5;
    x = s + d';
    x1 = x1 + x;
end
x1 = x1/50;
stem(m,x1);
xlabel('Time index n');
ylabel('Amplitude');
title('Ensemble average');
```



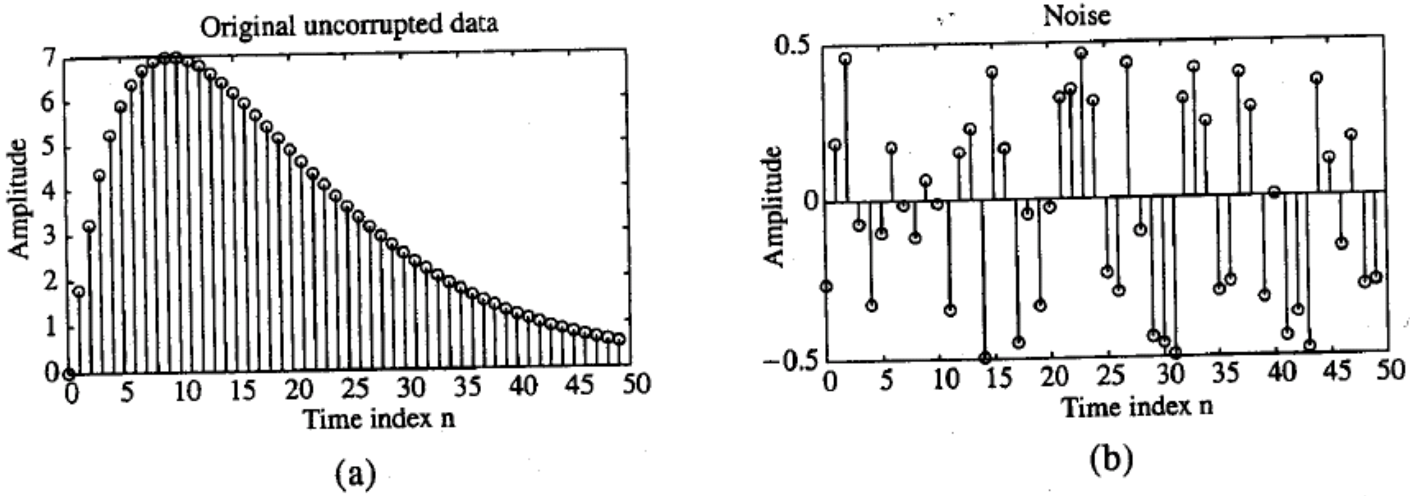


Figure 2.6: (a) The original uncorrupted sequence $s[n]$ and (b) the noise sequence $d_i[n]$.

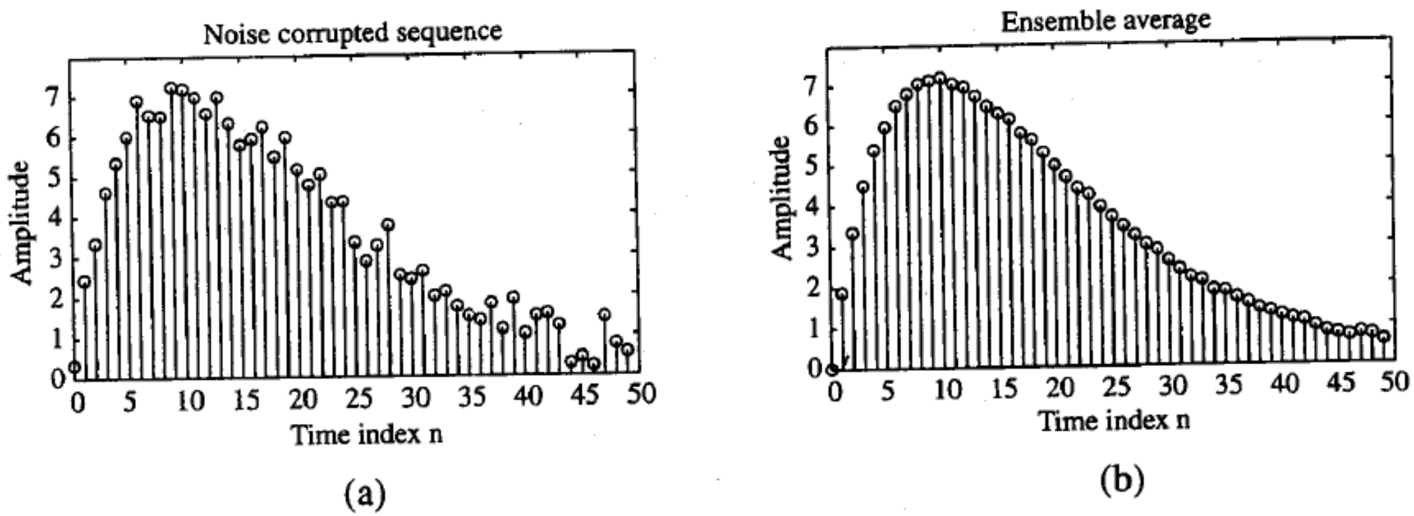


Figure 2.7: (a) A sample of the corrupted sequence and (b) the ensemble average after 50 measurements.

An application of ensemble averaging is in the power spectrum estimation of a random signal, discussed in Section 15.4.1.

The *time-shifting operation* illustrated below in Eq. (2.16) shows the relation between $x[n]$ and its time-shifted version $w_4[n]$:

$$w_4[n] = x[n - N], \tag{2.16}$$

where N is an integer. If $N > 0$, it is a *delaying* operation, and if $N < 0$, it is an *advancing* operation. For $N = 1$, we have the input–output relation

$$w_4[n] = x[n - 1],$$

and the device implementing the delay operation by one sample period is called a *unit delay*. In terms of z-transform, introduced later in Chapter 6, the above relation can be rewritten as

$$W_4(z) = z^{-1}X(z),$$

where $W_4(z)$ and $X(z)$ are, respectively, the z-transforms of the output sequence $w_4[n]$ and the input sequence $x[n]$. It is a usual practice to represent schematically the unit delay operation using the symbol z^{-1}, as shown in Figure 2.5(d).

The opposite of the unit delay operation is the unit advance operation defined by

$$w_5[n] = x[n+1],$$

which in terms of the z-transform is given by

$$W_5(z) = z\,X(z).$$

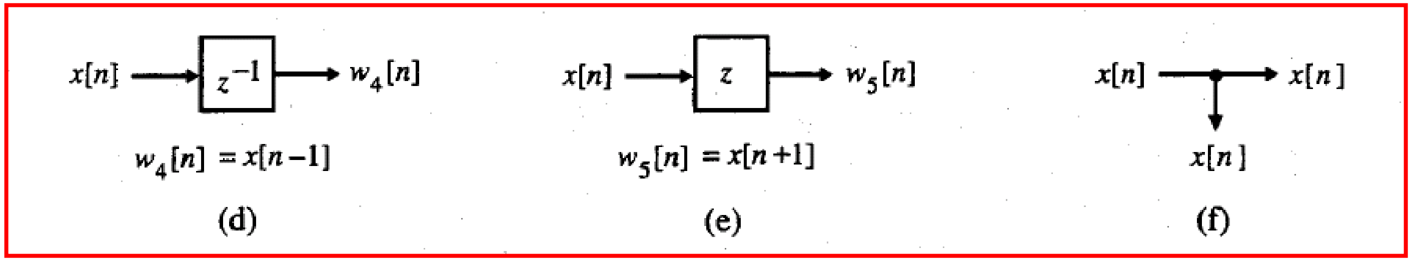
As a result, the *unit advance* operation is commonly represented schematically using the symbol z, as shown in Figure 2.5(e).

The *time-reversal* operation, also called the *folding operation*, is another useful scheme to develop a new sequence. An example is

$$w_6[n] = x[-n], \tag{2.17}$$

which is the time-reversed version of the sequence $x[n]$.

In Figure 2.5(f), we show *a pick-off node*, which is used to feed a sequence to different parts of a discrete-time system.



EXAMPLE 2.2 Basic Operations on Sequences of Equal Lengths

Consider the following two sequences of length 5 defined for $0 \le n \le 4$:

$$c[n] = \{3.2, \quad 41, \quad 36, \quad -9.5, \quad 0\},$$
$$d[n] = \{1.7, \quad -0.5, \quad 0, \quad 0.8, \quad 1\}.$$

Several new sequences of length 5 generated from the above sequences are given by

$$w_1[n] = c[n] \cdot d[n] = \{5.44, \quad -20.5, \quad 0, \quad -7.6, \quad 0\},$$
$$w_2[n] = c[n] + d[n] = \{4.9, \quad 40.5, \quad 36, \quad -8.7, \quad 1\},$$
$$w_3[n] = \tfrac{7}{2}c[n] = \{11.2, \quad 143.5, \quad 126, \quad -33.25, \quad 0\}.$$

As indicated by Example 2.2, operations on two or more sequences to generate a new sequence can be carried out if all sequences are of the same length and defined for the same range of the time index n. However, if the sequences are not of equal length, all sequences can be made to have the same range of the time index n by appending zero-valued samples to the sequence(s) of smaller lengths. This process is illustrated in Example 2.3.

EXAMPLE 2.3 Basic Operations on Sequences of Unequal Lengths

Consider a sequence $\{g[n]\}$ of length 3 defined for $0 \leq n \leq 2$ given by

$$\{g[n]\} = \{-21, \quad 1.5, \quad 3\}.$$

It is clear that we cannot develop another sequence by operating on this sequence and any one of the length-5 sequences of Example 2.2. However, it is possible to treat $\{g[n]\}$ as a sequence of length 5 and defined for $0 \leq n \leq 4$ by appending it with two zero-valued samples:

$$\{g_e[n]\} = \{-21, \quad 1.5, \quad 3, \quad 0, \quad 0\}.$$

Examples of new sequences generated from $\{g_e[n]\}$ and $c[n]$ of the previous example are indicated below:

$$\{w_4[n]\} = \{c[n] \cdot g_e[n]\} = \{-67.2, \quad 61.5, \quad 108, \quad 0, \quad 0\},$$
$$\{w_5[n]\} = \{c[n] + g_e[n]\} = \{-17.8, \quad 42.5, \quad 39, \quad -9.5, \quad 0\}.$$

Combination of Elementary Operations

In most applications, combinations of the above elementary operations are used. Illustrations of such combinations are given in Example 2.4.

EXAMPLE 2.4 Illustration of Combination of Basic Operations on Sequences

We next analyze the discrete-time system of Figure 2.8. Observe first that the two left-most delay blocks generate the sequences $x[n-1]$ and $x[n-2]$, whereas the two right-most delay blocks develop the sequences $y[n-1]$ and $y[n-2]$. These delayed sequences, along with $x[n]$, are then applied to the five multipliers, labeled b_0, b_1, b_2, a_1, and a_2, developing the sequences $b_0x[n]$, $b_1x[n-1]$, $b_2x[n-2]$, $a_1y[n-1]$, and $a_2y[n-2]$, which are then added to yield $y[n]$:

$$y[n] = b_0x[n] + b_1x[n-1] + b_2x[n-2] + a_1y[n-1] + a_2y[n-2]. \quad (2.18)$$

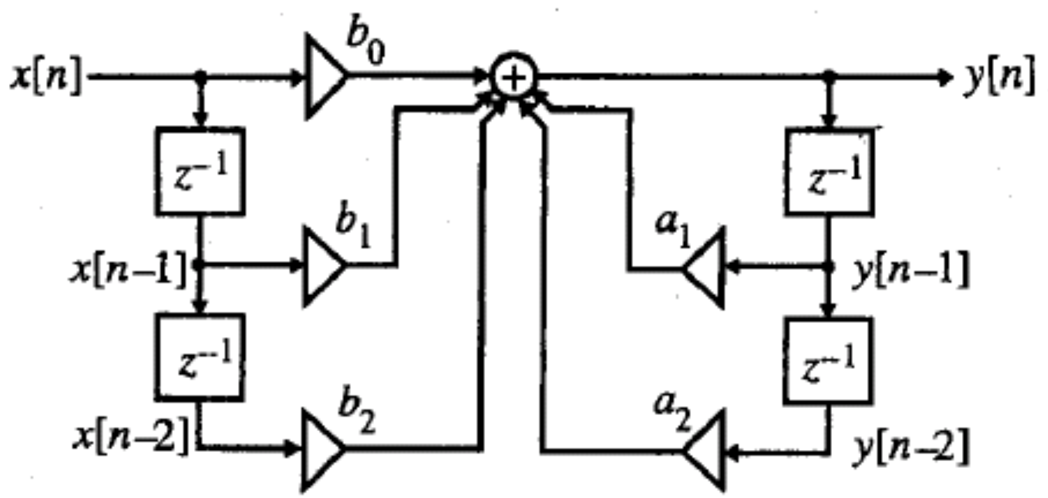


Figure 2.8: Discrete-time system of Example 2.4.

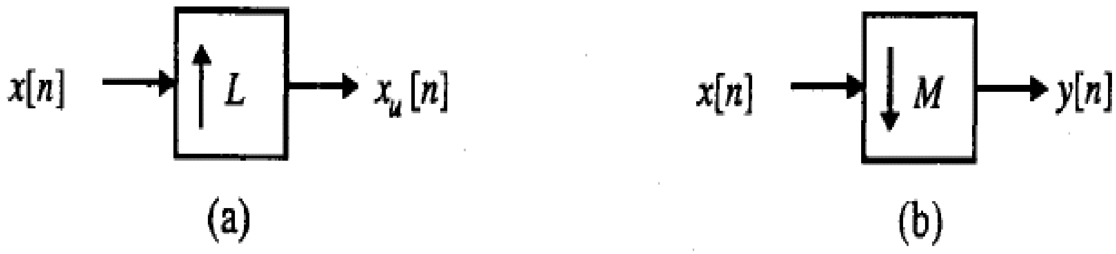


Figure 2.9: Representation of basic sampling rate alteration devices: (a) up-sampler and (b) down-sampler.

Sampling Rate Alteration

Another quite useful operation is the sampling rate alteration that is employed to generate a new sequence with a sampling rate higher or lower than that of a given sequence. Thus, if $x[n]$ is a sequence with a sampling rate of F_T Hz and it is used to generate another sequence $y[n]$ with a desired sampling rate of F'_T Hz, then the *sampling rate alteration ratio* is given by

$$\frac{F'_T}{F_T} = R. \tag{2.19}$$

If $R > 1$, the process is called *interpolation* and results in a sequence with a higher sampling rate. The discrete-time system implementing the interpolation process is called an *interpolator*. On the other hand, if $R < 1$, the sampling rate is decreased by a process called *decimation*. The discrete-time system implementing the decimation process is called a *decimator*..

The basic operations employed in the sampling rate alteration process are called *up-sampling* and *down-sampling*. These operations play important roles in multirate discrete-time systems and are considered in Chapters 13 and 14.

In up-sampling by an integer factor $L > 1$, $L - 1$ equidistant zero-valued samples are inserted by the up-sampler between each two consecutive samples of the input sequence $x[n]$ to develop an output sequence $x_u[n]$ according to the relation

$$x_u[n] = \begin{cases} x[n/L], & n = 0, \pm L, \pm 2L, \ldots, \\ 0, & \text{otherwise.} \end{cases} \tag{2.20}$$

The sampling rate of $x_u[n]$ is L times larger than that of the original sequence $x[n]$.

Classification Based on Symmetry

A sequence $x[n]$ is called a *conjugate-symmetric sequence* if $x[n] = x^*[-n]$. A real conjugate-symmetric sequence is called an *even sequence*. A sequence $x[n]$ is called a *conjugate-antisymmetric sequence* if $x[n] = -x^*[-n]$. A real conjugate-antisymmetric sequence is called an *odd sequence*. For a conjugate-antisymmetric sequence $x[n]$, the sample value at $n = 0$ must be purely imaginary. Consequently, for an odd sequence $x[0] = 0$. Examples of even and odd sequences are shown in Figure 2.12.

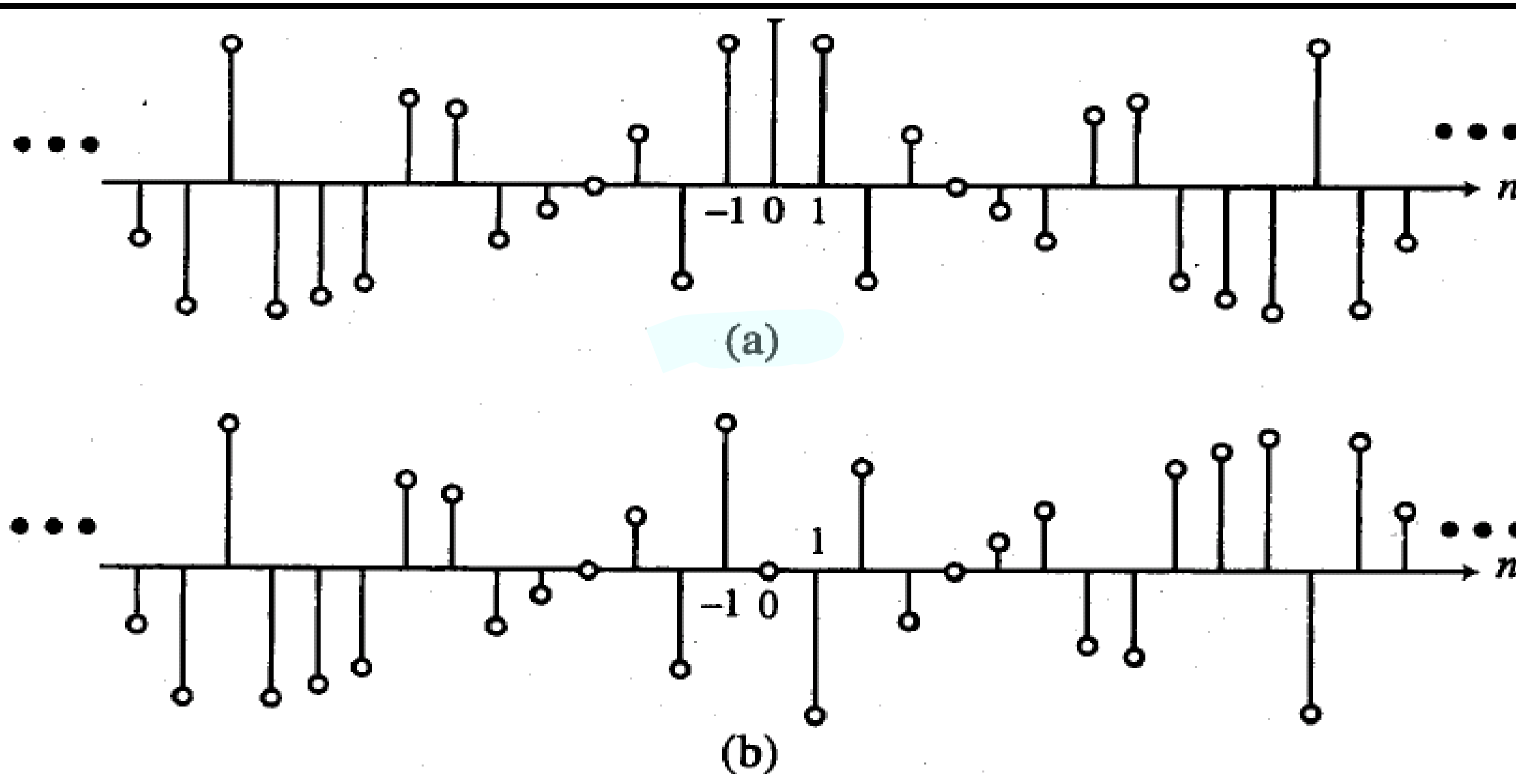


Figure 2.12: (a) An even sequence and (b) an odd sequence.

odd sequence $x[0] = 0$. Examples of even and odd s...

Any complex sequence $x[n]$ can be expressed as a sum of its conjugate-symmetric part $x_{cs}[n]$ and its conjugate-antisymmetric part $x_{ca}[n]$:

$$x[n] = x_{cs}[n] + x_{ca}[n], \tag{2.22}$$

where

$$x_{cs}[n] = \tfrac{1}{2}\left(x[n] + x^*[-n]\right), \tag{2.23a}$$

$$x_{ca}[n] = \tfrac{1}{2}\left(x[n] - x^*[-n]\right). \tag{2.23b}$$

As indicated by Eqs. (2.23a) and (2.23b), the computation of the conjugate-symmetric and conjugate-antisymmetric parts of a sequence involves conjugation, time-reversal, addition, and multiplication operations. Because of the time-reversal operation, the decomposition of a finite-length sequence into a sum of a conjugate-symmetric sequence and a conjugate-antisymmetric sequence is possible, if the parent sequence is of odd length defined for a symmetric interval, $-M \le 0 \le M$.

Consider the finite length sequence

$\{g[n]\}=\{0,\ \ 1+j4,\ \ -2+j3,\ \ \ 4-j2,\ \ -5-j6,\ \ -j2,\ \ 3\}$

Find its conjugate symmetric part and its conjugate antisymmetric part

$$\{g^*[n]\} = \{0,\ 1-j4,\ -2-j3,\ \underset{\uparrow}{4+j2},\ -5+j6,\ j2,\ 3\},$$

whose time-reversed version is then given by

$$\{g^*[-n]\} = \{3,\ j2,\ -5+j6,\ \underset{\uparrow}{4+j2},\ -2-j3,\ 1-j4,\ 0\}.$$

Using Eq. (2.23a), we thus arrive at

$$\{g_{cs}[n]\} = \{1.5,\ 0.5+j3,\ -3.5+j4.5,\ \underset{\uparrow}{4},\ -3.5-j4.5,\ 0.5-j3,\ 1.5\}.$$

Likewise, using Eq. (2.23b), we get

$$\{g_{ca}[n]\} = \{-1.5,\ 0.5+j,\ 1.5-j1.5,\ \underset{\uparrow}{-j2},\ -1.5-j1.5,\ -0.5+j,\ 1.5\}.$$

It can be easily verified that $g_{cs}[n] = g^*_{cs}[-n]$ and $g_{ca}[n] = -g^*_{ca}[-n]$.

Periodic and Aperiodic Signals

A sequence $\tilde{x}[n]$ satisfying

$$\tilde{x}[n] = \tilde{x}[n + kN] \qquad \text{for all } n \tag{2.26}$$

is called a *periodic sequence* with a *period* N, where N is a positive integer and k is any integer. An example of a periodic sequence that has a period $N = 7$ samples is shown in Figure 2.13. A sequence is called an *aperiodic sequence* if it is not periodic. To distinguish a periodic sequence from an aperiodic sequence, we shall denote the former with a "~" on top. The *fundamental period* N_f of a periodic signal is the smallest value of N for which Eq. (2.26) holds.

Sum of two or more periodic sequences is also a periodic sequence. If $\tilde{x}_a[n]$ and $\tilde{x}_b[n]$ are two periodic sequences with fundamental periods N_a and N_b, respectively, then the sequence $\tilde{y}[n] = \tilde{x}_a[n] + \tilde{x}_b[n]$ is a periodic sequence with a fundamental period N given by

$$N = \frac{N_a N_b}{GCD(N_a, N_b)}, \tag{2.27}$$

where $GCD(N_a, N_b)$ is the greatest common divisor of N_a and N_b.

Energy and Power Signals

The total *energy* of a sequence $x[n]$ is defined by

$$\mathcal{E}_x = \sum_{n=-\infty}^{\infty} |x[n]|^2 . \qquad (2.28)$$

An infinite-length sequence with finite sample values may or may not have finite energy, as illustrated in Example 2.6.

The *average power* of an aperiodic sequence $x[n]$ is defined by

$$\mathcal{P}_x = \lim_{K\to\infty} \frac{1}{2K+1} \sum_{n=-K}^{K} x[n]^2. \qquad (2.31)$$

The average power of a sequence can be related to its energy by defining its energy over a finite interval $-K \leq n \leq K$ as

$$\mathcal{E}_{x,K} = \sum_{n=-K}^{K} |x[n]|^2. \tag{2.32}$$

Then,

$$\mathcal{P}_x = \lim_{K\to\infty} \frac{1}{2K+1} \mathcal{E}_{x,K}. \tag{2.33}$$

The average power of a periodic sequence $\tilde{x}[n]$ with a period N is given by

$$\mathcal{P}_x = \frac{1}{N} \sum_{n=0}^{N-1} |\tilde{x}[n]|^2. \tag{2.34}$$

The average power of an infinite-length sequence may be finite or infinite.

An infinite energy signal with finite average power is called a *power signal*. Likewise, a finite energy signal with zero average power is called an *energy signal*. An example of a power signal is a periodic sequence that has a finite average power but infinite energy. An example of an energy signal is a finite-length sequence which has finite energy but zero average power.

Other Types of Classification

A sequence $x[n]$ is said to be *bounded* if each of its samples is of magnitude less than or equal to a finite positive number B_x, that is,

$$|x[n]| \le B_x < \infty. \tag{2.35}$$

The periodic sequence of Figure 2.13 is a bounded sequence with a bound $B_x = 2$.

A sequence $x[n]$ is said to be *absolutely summable* if

$$\sum_{n=-\infty}^{\infty} |x[n]| < \infty. \tag{2.36}$$

A sequence is said to be *square-summable* if

$$\sum_{n=-\infty}^{\infty} |x[n]|^2 < \infty. \tag{2.37}$$

A square-summable sequence therefore has finite energy and is an energy signal if it also has zero power.

An example of a sequence that is square-summable but not absolutely-summable is

$$x_a[n] = \frac{\sin \omega_c n}{\pi n}, \qquad -\infty < n < \infty.$$

Examples of sequences that are neither absolutely-summable nor square-summable are

$$x_b[n] = \sin \omega_c n, \qquad -\infty < n < \infty,$$
$$x_c[n] = K, \qquad -\infty < n < \infty,$$

where K is a constant.

The sequence $\tilde{y}[n]$ obtained by adding an absolutely summable sequence $x[n]$ with its replicas shifted by integer multiples of N,

$$\tilde{y}[n] = \sum_{k=-\infty}^{\infty} x[n + kN], \tag{2.38}$$

where N is a positive integer, is a periodic sequence with a period N. The periodic sequence $\tilde{y}[n]$ is called an *N-periodic extension* of $x[n]$.