ELCoM

Electrical | Computer | Mechatronics

www.ELCOM-HU.com

إرادة .. ثقة .. تغيير

Digital Signal Processing

Fourth Edition

John G. Proakis

Department of Electrical and Computer Engineering
Northeastern University
Boston, Massachusetts

Dimitris G. Manolakis

MIT Lincoln Laboratory
Lexington, Massachusetts

PEARSON
Prentice Hall

Upper Saddle River, New Jersey 07458

Contents

Preface

This book was developed based on our teaching of undergraduate- and graduate-level courses in digital signal processing over the past several years. In this book we present the fundamentals of discrete-time signals, systems, and modern digital processing as well as applications for students in electrical engineering, computer engineering, and computer science. The book is suitable for either a one-semester or a two-semester undergraduate-level course in discrete systems and digital signal processing. It is also intended for use in a one-semester first-year graduate-level course in digital signal processing.

It is assumed that the student has had undergraduate courses in advanced calculus (including ordinary differential equations) and linear systems for continuous-time signals, including an introduction to the Laplace transform. Although the Fourier series and Fourier transforms of periodic and aperiodic signals are described in Chapter 4, we expect that many students may have had this material in a prior course.

Balanced coverage of both theory and practical applications is provided. A large number of well-designed problems are provided to help the student in mastering the subject matter. A solutions manual is available for download for instructors only. Additionally, Microsoft PowerPoint slides of text figures are available for instructors on the publisher's website.

In the fourth edition of the book, we have added a new chapter on adaptive filters and have substantially modified and updated the chapters on multirate digital signal processing and on sampling and reconstruction of signals. We have also added new material on the discrete cosine transform.

In Chapter 1 we describe the operations involved in the analog-to-digital conversion of analog signals. The process of sampling a sinusoid is described in some detail and the problem of aliasing is explained. Signal quantization and digital-to-analog conversion are also described in general terms, but the analysis is presented in subsequent chapters.

Chapter 2 is devoted entirely to the characterization and analysis of linear time-invariant (shift-invariant) discrete-time systems and discrete-time signals in the time domain. The convolution sum is derived and systems are categorized according to the duration of their impulse response as a finite-duration impulse response (FIR) and as an infinite-duration impulse response (IIR). Linear time-invariant systems characterized by difference equations are presented and the solution of difference equations with initial conditions is obtained. The chapter concludes with a treatment of discrete-time correlation.

The z-transform is introduced in Chapter 3. Both the bilateral and the unilateral z-transforms are presented, and methods for determining the inverse z-transform are described. Use of the z-transform in the analysis of linear time-invariant systems is illustrated, and important properties of systems, such as causality and stability, are related to $z-$domain characteristics.

Chapter 4 treats the analysis of signals in the frequency domain. Fourier series and the Fourier transform are presented for both continuous-time and discrete-time signals.

In Chapter 5, linear time-invariant (LTI) discrete systems are characterized in the frequency domain by their frequency response function and their response to periodic and aperiodic signals is determined. A number of important types of discrete-time systems are described, including resonators, notch filters, comb filters, all-pass filters, and oscillators. The design of a number of simple FIR and IIR filters is also considered In addition, the student is introduced to the concepts of minimum-phase, mixed-phase, and maximum-phase systems and to the problem of deconvolution.

Chapter 6 provides a thorough treatment of sampling of continuous-time signals and the reconstruction of the signals from their samples Our coverage includes the sampling and reconstruction of bandpass signals, the sampling of discrete-time signals, and A/D and D/A conversion. The chapter concludes with the treatment of oversampling A/D and D/A converters.

The DFT, its properties and its applications, are the topics covered in Chapter 7. Two methods are described for using the DFT to perform linear filtering. The use of the DFT to perform frequency analysis of signals is also described. The final topic treated in this chapter is the discrete cosine transform.

Chapter 8 covers the efficient computation of the DFT. Included in this chapter are descriptions of radix-2, radix-4, and split-radix fast Fourier transform (FFT) algorithms, and applications of the FFT algorithms to the computation of convolution and correlation. The Goertzel algorithm and the chirp-z transform are introduced as two methods for computing the DFT using linear filtering.

Chapter 9 treats the realization of IIR and FIR systems. This treatment includes direct-form, cascade, parallel, lattice, and lattice-ladder realizations. The chapter also examines quantization effects in a digital implementation of FIR and IIR systems.

Techniques for design of digital FIR and IIR filters are presented in Chapter 10. The design techniques include both direct methods in discrete time and methods involving the conversion of analog filters into digital filters by various transformations.

Chapter 11 treats sampling-rate conversion and its applications to multirate digital signal processing. In addition to describing decimation and interpolation by integer and rational factors, we describe methods for sampling-rate conversion by an arbitrary factor and implementations by polyphase filter structures. This chapter also treats digital filter banks, two-channel quadrature mirror filters (QMF) and M-channel QMF banks.

Linear prediction and optimum linear (Wiener) filters are treated in Chapter 12. Also included in this chapter are descriptions of the Levinson–Durbin algorithm and Schur algorithm for solving the normal equations, as well as the AR lattice and ARMA lattice-ladder filters.

Chapter 13 treats single-channel adaptive filters based on the LMS algorithm and on recursive least squares (RLS) algorithms. Both direct form FIR and lattice RLS algorithms and filter structures are described.

Power spectrum estimation is the main topic of Chapter 14. Our coverage includes a description of nonparametric and model-based (parametric) methods. Also described are eigen-decomposition-based methods, including MUSIC and ESPRIT.

A one-semester senior-level course for students who have had prior exposure to discrete systems can use the material in Chapters 1 through 5 for a quick review and then proceed to cover Chapters 6 through 10.

In a first-year graduate-level course in digital signal processing, the first six chapters provide the student with a good review of discrete-time systems. The instructor can move quickly through most of this material and then cover Chapters 7 through 11, followed by selected topics from Chapters 12 through 14.

Many examples throughout the book and approximately 500 homework problems are included throughout the book. Answers to selected problems appear in the back of the book. Many of the homework problems can be solved numerically on a computer, using a software package such as MATLAB®. Available for use as a self-study companion to the textbook is a student manual: *Student Manual for Digital Signal Processing with MATLAB®*. MATLAB is incorporated as the basic software tool for this manual. The instructor may also wish to consider the use of other supplementary books that contain computer-based exercises, such as *Computer-Based Exercises for Signal Processing Using MATLAB* (Prentice Hall, 1994) by C. S. Burrus *et al.*

The authors are indebted to their many faculty colleagues who have provided valuable suggestions through reviews of previous editions of this book. These include W. E. Alexander, G. Arslan, Y. Bresler, J. Deller, F. DePiero, V. Ingle, J.S. Kang, C. Keller, H. Lev-Ari, L. Merakos, W. Mikhael, P. Monticciolo, C. Nikias, M. Schetzen, E. Serpedin, T. M. Sullivan, H. Trussell, S. Wilson, and M. Zoltowski. We are also indebted to R. Price for recommending the inclusion of split-radix FFT algorithms and related suggestions. Finally, we wish to acknowledge the suggestions and comments of many former graduate students, and especially those by A. L. Kok, J. Lin, E. Sozer, and S. Srinidhi, who assisted in the preparation of several illustrations and the solutions manual.

JOHN G. PROAKIS
DIMITRIS G. MANOLAKIS

CHAPTER

1

Introduction

Digital signal processing is an area of science and engineering that has developed rapidly over the past 40 years. This rapid development is a result of the significant advances in digital computer technology and integrated-circuit fabrication. The digital computers and associated digital hardware of four decades ago were relatively large and expensive and, as a consequence, their use was limited to general-purpose non-real-time (off-line) scientific computations and business applications. The rapid developments in integrated-circuit technology, starting with medium-scale integration (MSI) and progressing to large-scale integration (LSI), and now, very-large-scale integration (VLSI) of electronic circuits has spurred the development of powerful, smaller, faster, and cheaper digital computers and special-purpose digital hardware. These inexpensive and relatively fast digital circuits have made it possible to construct highly sophisticated digital systems capable of performing complex digital signal processing functions and tasks, which are usually too difficult and/or too expensive to be performed by analog circuitry or analog signal processing systems. Hence many of the signal processing tasks that were conventionally performed by analog means are realized today by less expensive and often more reliable digital hardware.

We do not wish to imply that digital signal processing is the proper solution for all signal processing problems. Indeed, for many signals with extremely wide bandwidths, real-time processing is a requirement. For such signals, analog or, perhaps, optical signal processing is the only possible solution. However, where digital circuits are available and have sufficient speed to perform the signal processing, they are usually preferable.

Not only do digital circuits yield cheaper and more reliable systems for signal processing, they have other advantages as well. In particular, digital processing hardware allows programmable operations. Through software, one can more eas-

ily modify the signal processing functions to be performed by the hardware. Thus digital hardware and associated software provide a greater degree of flexibility in system design. Also, there is often a higher order of precision achievable with digital hardware and software compared with analog circuits and analog signal processing systems. For all these reasons, there has been an explosive growth in digital signal processing theory and applications over the past three decades.

In this book our objective is to present an introduction of the basic analysis tools and techniques for digital processing of signals. We begin by introducing some of the necessary terminology and by describing the important operations associated with the process of converting an analog signal to digital form suitable for digital processing. As we shall see, digital processing of analog signals has some drawbacks. First, and foremost, conversion of an analog signal to digital form, accomplished by sampling the signal and quantizing the samples, results in a distortion that prevents us from reconstructing the original analog signal from the quantized samples. Control of the amount of this distortion is achieved by proper choice of the sampling rate and the precision in the quantization process. Second, there are finite precision effects that must be considered in the digital processing of the quantized samples. While these important issues are considered in some detail in this book, the emphasis is on the analysis and design of digital signal processing systems and computational techniques.

1.1 Signals, Systems, and Signal Processing

A *signal* is defined as any physical quantity that varies with time, space, or any other independent variable or variables. Mathematically, we describe a signal as a function of one or more independent variables. For example, the functions

$$\begin{aligned} s_1(t) &= 5t \\ s_2(t) &= 20t^2 \end{aligned} \tag{1.1.1}$$

describe two signals, one that varies linearly with the independent variable t (time) and a second that varies quadratically with t. As another example, consider the function

$$s(x, y) = 3x + 2xy + 10y^2 \tag{1.1.2}$$

This function describes a signal of two independent variables x and y that could represent the two spatial coordinates in a plane.

The signals described by (1.1.1) and (1.1.2) belong to a class of signals that are precisely defined by specifying the functional dependence on the independent variable. However, there are cases where such a functional relationship is unknown or too highly complicated to be of any practical use.

For example, a speech signal (see Fig. 1.1.1) cannot be described functionally by expressions such as (1.1.1). In general, a segment of speech may be represented to

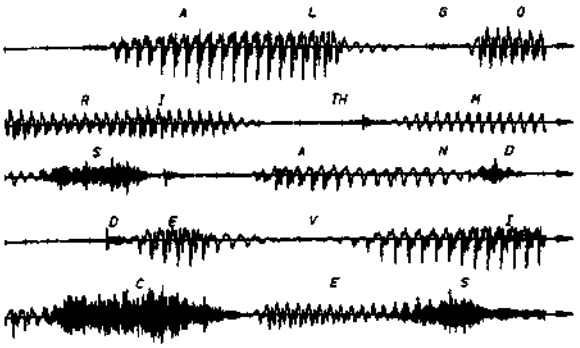


Figure 1.1.1
Example of a speech signal.

a high degree of accuracy as a sum of several sinusoids of different amplitudes and frequencies, that is, as

$$\sum_{i=1}^{N} A_i(t) \sin[2\pi F_i(t)t + \theta_i(t)] \tag{1.1.3}$$

where $\{A_i(t)\}$, $\{F_i(t)\}$, and $\{\theta_i(t)\}$ are the sets of (possibly time-varying) amplitudes, frequencies, and phases, respectively, of the sinusoids. In fact, one way to interpret the information content or message conveyed by any short time segment of the speech signal is to measure the amplitudes, frequencies, and phases contained in the short time segment of the signal.

Another example of a natural signal is an electrocardiogram (ECG). Such a signal provides a doctor with information about the condition of the patient's heart. Similarly, an electroencephalogram (EEG) signal provides information about the activity of the brain.

Speech, electrocardiogram, and electroencephalogram signals are examples of information-bearing signals that evolve as functions of a single independent variable, namely, time. An example of a signal that is a function of two independent variables is an image signal. The independent variables in this case are the spatial coordinates. These are but a few examples of the countless number of natural signals encountered in practice.

Associated with natural signals are the means by which such signals are generated. For example, speech signals are generated by forcing air through the vocal cords. Images are obtained by exposing a photographic film to a scene or an object. Thus signal generation is usually associated with a *system* that responds to a stimulus or force. In a speech signal, the system consists of the vocal cords and the vocal tract, also called the vocal cavity. The stimulus in combination with the system is called a *signal source*. Thus we have speech sources, images sources, and various other types of signal sources.

A *system* may also be defined as a physical device that performs an operation on a signal. For example, a filter used to reduce the noise and interference corrupting a desired information-bearing signal is called a system. In this case the filter performs some operation(s) on the signal, which has the effect of reducing (filtering) the noise and interference from the desired information-bearing signal.

When we pass a signal through a system, as in filtering, we say that we have processed the signal. In this case the processing of the signal involves filtering the noise and interference from the desired signal. In general, the system is characterized by the type of operation that it performs on the signal. For example, if the operation is linear, the system is called linear. If the operation on the signal is nonlinear, the system is said to be nonlinear, and so forth. Such operations are usually referred to as *signal processing*.

For our purposes, it is convenient to broaden the definition of a system to include not only physical devices, but also software realizations of operations on a signal. In digital processing of signals on a digital computer, the operations performed on a signal consist of a number of mathematical operations as specified by a software program. In this case, the program represents an implementation of the system in *software*. Thus we have a system that is realized on a digital computer by means of a sequence of mathematical operations; that is, we have a digital signal processing system realized in software. For example, a digital computer can be programmed to perform digital filtering. Alternatively, the digital processing on the signal may be performed by digital *hardware* (logic circuits) configured to perform the desired specified operations. In such a realization, we have a physical device that performs the specified operations. In a broader sense, a digital system can be implemented as a combination of digital hardware and software, each of which performs its own set of specified operations.

This book deals with the processing of signals by digital means, either in software or in hardware. Since many of the signals encountered in practice are analog, we will also consider the problem of converting an analog signal into a digital signal for processing. Thus we will be dealing primarily with digital systems. The operations performed by such a system can usually be specified mathematically. The method or set of rules for implementing the system by a program that performs the corresponding mathematical operations is called an *algorithm*. Usually, there are many ways or algorithms by which a system can be implemented, either in software or in hardware, to perform the desired operations and computations. In practice, we have an interest in devising algorithms that are computationally efficient, fast, and easily implemented. Thus a major topic in our study of digital signal processing is the discussion of efficient algorithms for performing such operations as filtering, correlation, and spectral analysis.

1.1.1 Basic Elements of a Digital Signal Processing System

Most of the signals encountered in science and engineering are analog in nature. That is, the signals are functions of a continuous variable, such as time or space, and usually take on values in a continuous range. Such signals may be processed directly by appropriate analog systems (such as filters, frequency analyzers, or frequency multipliers) for the purpose of changing their characteristics or extracting some desired information. In such a case we say that the signal has been processed directly in its analog form, as illustrated in Fig. 1.1.2. Both the input signal and the output signal are in analog form.

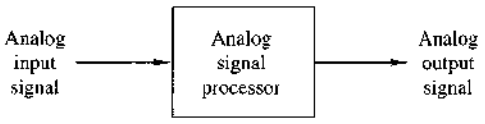


Figure 1.1.2
Analog signal processing.

Digital signal processing provides an alternative method for processing the analog signal, as illustrated in Fig. 1.1.3. To perform the processing digitally, there is a need for an interface between the analog signal and the digital processor. This interface is called an *analog-to-digital (A/D) converter*. The output of the A/D converter is a digital signal that is appropriate as an input to the digital processor.

The digital signal processor may be a large programmable digital computer or a small microprocessor programmed to perform the desired operations on the input signal. It may also be a hardwired digital processor configured to perform a specified set of operations on the input signal. Programmable machines provide the flexibility to change the signal processing operations through a change in the software, whereas hardwired machines are difficult to reconfigure. Consequently, programmable signal processors are in very common use. On the other hand, when signal processing operations are well defined, a hardwired implementation of the operations can be optimized, resulting in a cheaper signal processor and, usually, one that runs faster than its programmable counterpart. In applications where the digital output from the digital signal processor is to be given to the user in analog form, such as in speech communications, we must provide another interface from the digital domain to the analog domain. Such an interface is called a *digital-to-analog (D/A) converter*. Thus the signal is provided to the user in analog form, as illustrated in the block diagram of Fig. 1.1.3. However, there are other practical applications involving signal analysis, where the desired information is conveyed in digital form and no D/A converter is required. For example, in the digital processing of radar signals, the information extracted from the radar signal, such as the position of the aircraft and its speed, may simply be printed on paper. There is no need for a D/A converter in this case.

1.1.2 Advantages of Digital over Analog Signal Processing

There are many reasons why digital signal processing of an analog signal may be preferable to processing the signal directly in the analog domain, as mentioned briefly earlier. First, a digital programmable system allows flexibility in reconfiguring the digital signal processing operations simply by changing the program. Reconfigu-

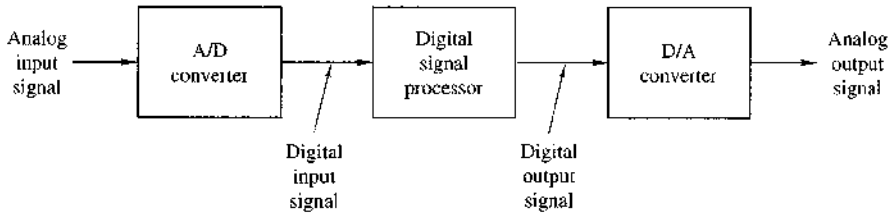


Figure 1.1.3 Block diagram of a digital signal processing system.

ration of an analog system usually implies a redesign of the hardware followed by testing and verification to see that it operates properly.

Accuracy considerations also play an important role in determining the form of the signal processor. Tolerances in analog circuit components make it extremely difficult for the system designer to control the accuracy of an analog signal processing system. On the other hand, a digital system provides much better control of accuracy requirements. Such requirements, in turn, result in specifying the accuracy requirements in the A/D converter and the digital signal processor, in terms of word length, floating-point versus fixed-point arithmetic, and similar factors.

Digital signals are easily stored on magnetic media (tape or disk) without deterioration or loss of signal fidelity beyond that introduced in the A/D conversion. As a consequence, the signals become transportable and can be processed off-line in a remote laboratory. The digital signal processing method also allows for the implementation of more sophisticated signal processing algorithms. It is usually very difficult to perform precise mathematical operations on signals in analog form but these same operations can be routinely implemented on a digital computer using software.

In some cases a digital implementation of the signal processing system is cheaper than its analog counterpart. The lower cost may be due to the fact that the digital hardware is cheaper, or perhaps it is a result of the flexibility for modifications provided by the digital implementation.

As a consequence of these advantages, digital signal processing has been applied in practical systems covering a broad range of disciplines. We cite, for example, the application of digital signal processing techniques in speech processing and signal transmission on telephone channels, in image processing and transmission, in seismology and geophysics, in oil exploration, in the detection of nuclear explosions, in the processing of signals received from outer space, and in a vast variety of other applications. Some of these applications are cited in subsequent chapters.

As already indicated, however, digital implementation has its limitations. One practical limitation is the speed of operation of A/D converters and digital signal processors. We shall see that signals having extremely wide bandwidths require fast-sampling-rate A/D converters and fast digital signal processors. Hence there are analog signals with large bandwidths for which a digital processing approach is beyond the state of the art of digital hardware.

1.2 Classification of Signals

The methods we use in processing a signal or in analyzing the response of a system to a signal depend heavily on the characteristic attributes of the specific signal. There are techniques that apply only to specific families of signals. Consequently, any investigation in signal processing should start with a classification of the signals involved in the specific application.

1.2.1 Multichannel and Multidimensional Signals

As explained in Section 1.1, a signal is described by a function of one or more independent variables. The value of the function (i.e., the dependent variable) can be

a real-valued scalar quantity, a complex-valued quantity, or perhaps a vector. For example, the signal

$$s_1(t) = A \sin 3\pi t$$

is a real-valued signal. However, the signal

$$s_2(t) = Ae^{j3\pi t} = A \cos 3\pi t + jA \sin 3\pi t$$

is complex valued.

In some applications, signals are generated by multiple sources or multiple sensors. Such signals, in turn, can be represented in vector form. Figure 1.2.1 shows the three components of a vector signal that represents the ground acceleration due to an earthquake. This acceleration is the result of three basic types of elastic waves. The primary (P) waves and the secondary (S) waves propagate within the body of

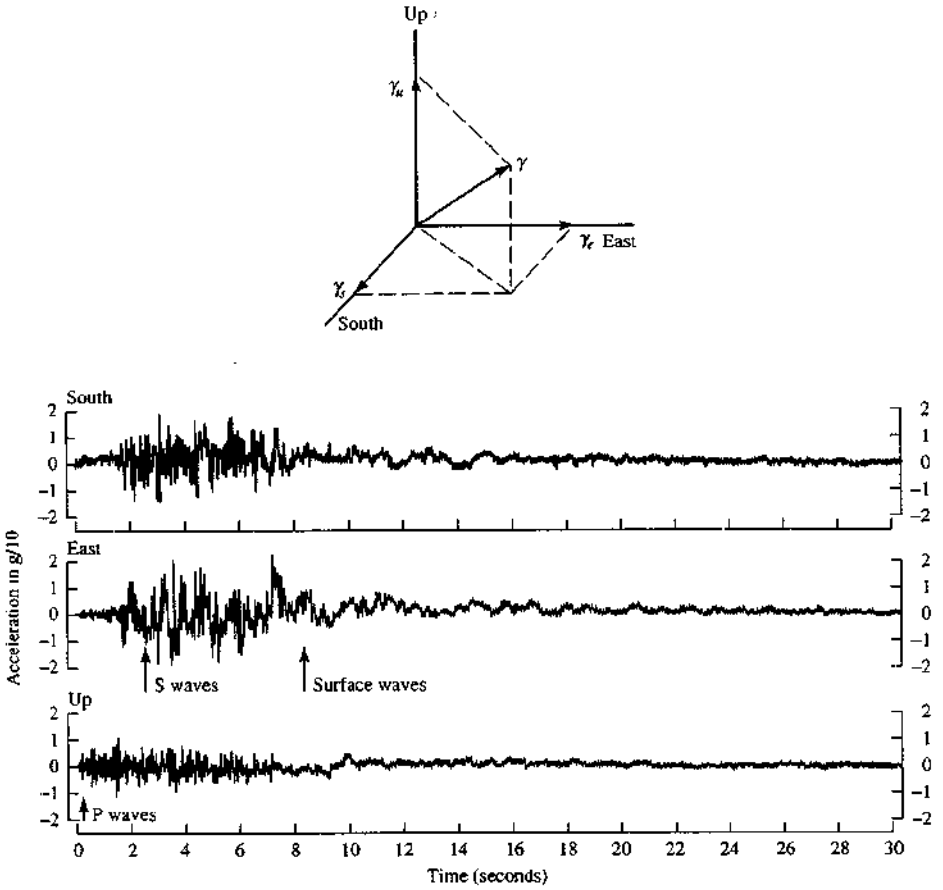


Figure 1.2.1 Three components of ground acceleration measured a few kilometers from the epicenter of an earthquake. (From *Earthquakes*, by B. A. Bold, ©1988 by W. H. Freeman and Company. Reprinted with permission of the publisher.)

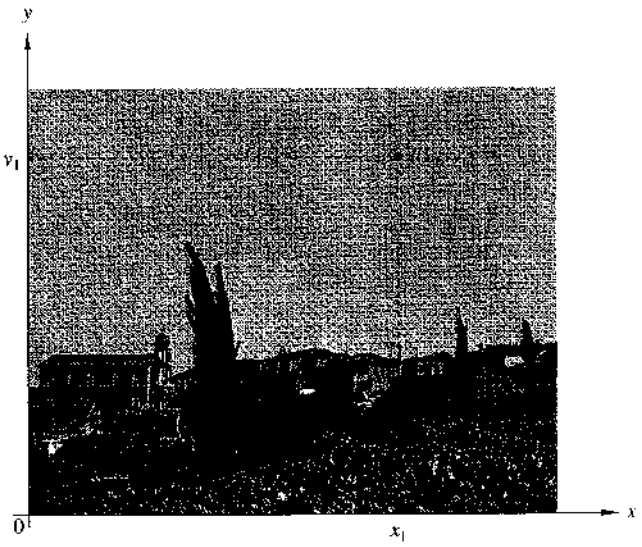


Figure 1.2.2
Example of a two-dimensional signal.

rock and are longitudinal and transversal, respectively. The third type of elastic wave is called the surface wave, because it propagates near the ground surface. If $s_k(t)$, $k = 1, 2, 3$, denotes the electrical signal from the kth sensor as a function of time, the set of $p = 3$ signals can be represented by a vector $\mathbf{S}_3(t)$, where

$$\mathbf{S}_3(t) = \begin{bmatrix} s_1(t) \\ s_2(t) \\ s_3(t) \end{bmatrix}$$

We refer to such a vector of signals as a *multichannel signal*. In electrocardiography, for example, 3-lead and 12-lead electrocardiograms (ECG) are often used in practice, which result in 3-channel and 12-channel signals.

Let us now turn our attention to the independent variable(s). If the signal is a function of a single independent variable, the signal is called a *one-dimensional* signal. On the other hand, a signal is called *M-dimensional* if its value is a function of M independent variables.

The picture shown in Fig. 1.2.2 is an example of a two-dimensional signal, since the intensity or brightness $I(x, y)$ at each point is a function of two independent variables. On the other hand, a black-and-white television picture may be represented as $I(x, y, t)$ since the brightness is a function of time. Hence the TV picture may be treated as a three-dimensional signal. In contrast, a color TV picture may be described by three intensity functions of the form $I_r(x, y, t)$, $I_g(x, y, t)$, and $I_b(x, y, t)$, corresponding to the brightness of the three principal colors (red, green, blue) as functions of time. Hence the color TV picture is a three-channel, three-dimensional signal, which can be represented by the vector

$$\mathbf{I}(x, y, t) = \begin{bmatrix} I_r(x, y, t) \\ I_g(x, y, t) \\ I_b(x, y, t) \end{bmatrix}$$

In this book we deal mainly with single-channel, one-dimensional real- or complex-valued signals and we refer to them simply as signals. In mathematical terms these signals are described by a function of a single independent variable. Although the independent variable need not be time, it is common practice to use t as the independent variable. In many cases the signal processing operations and algorithms developed in this text for one-dimensional, single-channel signals can be extended to multichannel and multidimensional signals.

1.2.2 Continuous-Time Versus Discrete-Time Signals

Signals can be further classified into four different categories depending on the characteristics of the time (independent) variable and the values they take. *Continuous-time signals* or *analog signals* are defined for every value of time and they take on values in the continuous interval (a, b), where a can be $-\infty$ and b can be ∞. Mathematically, these signals can be described by functions of a continuous variable. The speech waveform in Fig. 1.1.1 and the signals $x_1(t) = \cos \pi t$, $x_2(t) = e^{-|t|}$, $-\infty < t < \infty$ are examples of analog signals. *Discrete-time signals* are defined only at certain specific values of time. These time instants need not be equidistant, but in practice they are usually taken at equally spaced intervals for computational convenience and mathematical tractability. The signal $x(t_n) = e^{-|t_n|}$, $n = 0, \pm 1, \pm 2, \ldots$ provides an example of a discrete-time signal. If we use the index n of the discrete-time instants as the independent variable, the signal value becomes a function of an integer variable (i.e., a sequence of numbers). Thus a discrete-time signal can be represented mathematically by a sequence of real or complex numbers. To emphasize the discrete-time nature of a signal, we shall denote such a signal as $x(n)$ instead of $x(t)$. If the time instants t_n are equally spaced (i.e., $t_n = nT$), the notation $x(nT)$ is also used. For example, the sequence

$$x(n) = \begin{cases} 0.8^n, & \text{if } n \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{1.2.1}$$

is a discrete-time signal, which is represented graphically as in Fig. 1.2.3.

In applications, discrete-time signals may arise in two ways:

1. By selecting values of an analog signal at discrete-time instants. This process is called *sampling* and is discussed in more detail in Section 1.4. All measuring instruments that take measurements at a regular interval of time provide

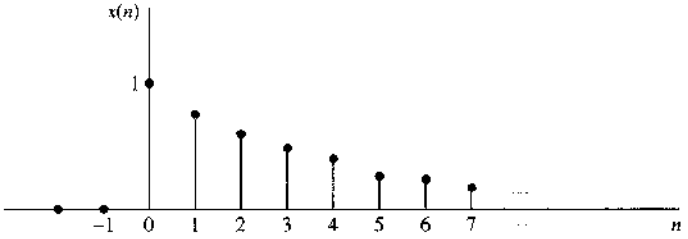


Figure 1.2.3 Graphical representation of the discrete time signal $x(n) = 0.8^n$ for $n > 0$ and $x(n) = 0$ for $n < 0$.

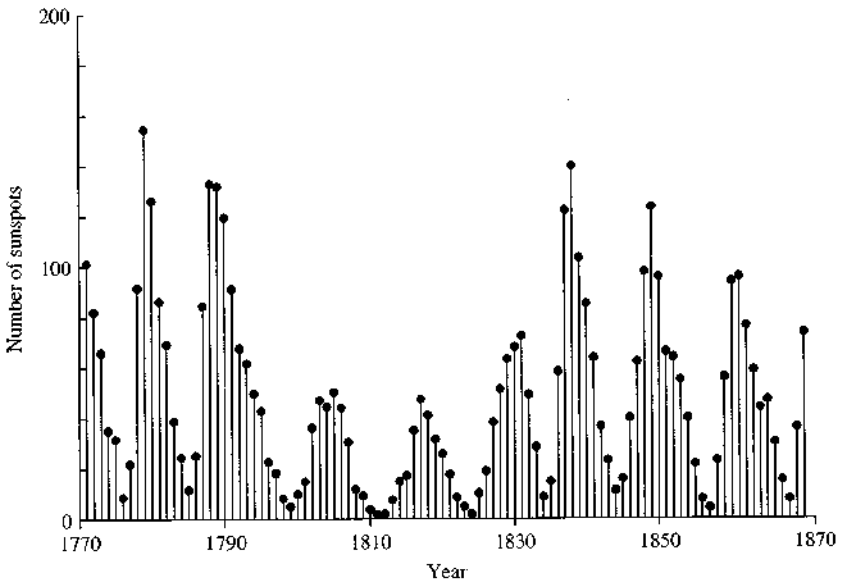


Figure 1.2.4 Wölfer annual sunspot numbers (1770–1869).

discrete-time signals. For example, the signal $x(n)$ in Fig. 1.2.3 can be obtained by sampling the analog signal $x(t) = 0.8^t$, $t \geq 0$ and $x(t) = 0$, $t < 0$ once every second.

2. By accumulating a variable over a period of time. For example, counting the number of cars using a given street every hour, or recording the value of gold every day, results in discrete-time signals. Figure 1.2.4 shows a graph of the Wölfer sunspot numbers. Each sample of this discrete-time signal provides the number of sunspots observed during an interval of 1 year.

1.2.3 Continuous-Valued Versus Discrete-Valued Signals

The values of a continuous-time or discrete-time signal can be continuous or discrete. If a signal takes on all possible values on a finite or an infinite range, it is said to be a continuous-valued signal. Alternatively, if the signal takes on values from a finite set of possible values, it is said to be a discrete-valued signal. Usually, these values are equidistant and hence can be expressed as an integer multiple of the distance between two successive values. A discrete-time signal having a set of discrete values is called a *digital signal*. Figure 1.2.5 shows a digital signal that takes on one of four possible values.

In order for a signal to be processed digitally, it must be discrete in time and its values must be discrete (i.e., it must be a digital signal). If the signal to be processed is in analog form, it is converted to a digital signal by sampling the analog signal at discrete instants in time, obtaining a discrete-time signal, and then by *quantizing* its values to a set of discrete values, as described later in the chapter. The process

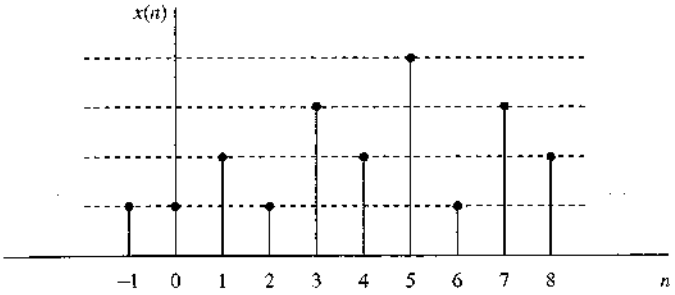


Figure 1.2.5 Digital signal with four different amplitude values.

of converting a continuous-valued signal into a discrete-valued signal, called *quantization*, is basically an approximation process. It may be accomplished simply by rounding or truncation. For example, if the allowable signal values in the digital signal are integers, say 0 through 15, the continuous-value signal is quantized into these integer values. Thus the signal value 8.58 will be approximated by the value 8 if the quantization process is performed by truncation or by 9 if the quantization process is performed by rounding to the nearest integer. An explanation of the analog-to-digital conversion process is given later in the chapter.

1.2.4 Deterministic Versus Random Signals

The mathematical analysis and processing of signals requires the availability of a mathematical description for the signal itself. This mathematical description, often referred to as the *signal model*, leads to another important classification of signals. Any signal that can be uniquely described by an explicit mathematical expression, a table of data, or a well-defined rule is called *deterministic*. This term is used to emphasize the fact that all past, present, and future values of the signal are known precisely, without any uncertainty.

In many practical applications, however, there are signals that either cannot be described to any reasonable degree of accuracy by explicit mathematical formulas, or such a description is too complicated to be of any practical use. The lack of such a relationship implies that such signals evolve in time in an unpredictable manner. We refer to these signals as *random*.The output of a noise generator, the seismic signal of Fig. 1.2.1, and the speech signal in Fig. 1.1.1 are examples of random signals.

The mathematical framework for the theoretical analysis of random signals is provided by the theory of probability and stochastic processes. Some basic elements of this approach, adapted to the needs of this book, are presented in Section 12.1.

It should be emphasized at this point that the classification of a *real-world* signal as deterministic or random is not always clear. Sometimes, both approaches lead to meaningful results that provide more insight into signal behavior. At other times, the wrong classification may lead to erroneous results, since some mathematical tools may apply only to deterministic signals while others may apply only to random signals. This will become clearer as we examine specific mathematical tools.

1.3 The Concept of Frequency in Continuous-Time and Discrete-Time Signals

The concept of frequency is familiar to students in engineering and the sciences. This concept is basic in, for example, the design of a radio receiver, a high-fidelity system, or a spectral filter for color photography. From physics we know that frequency is closely related to a specific type of periodic motion called harmonic oscillation, which is described by sinusoidal functions. The concept of frequency is directly related to the concept of time. Actually, it has the dimension of inverse time. Thus we should expect that the nature of time (continuous or discrete) would affect the nature of the frequency accordingly.

1.3.1 Continuous-Time Sinusoidal Signals

A simple harmonic oscillation is mathematically described by the following continuous-time sinusoidal signal:

$$x_a(t) = A\cos(\Omega t + \theta), \quad -\infty < t < \infty \tag{1.3.1}$$

shown in Fig. 1.3.1. The subscript a used with $x(t)$ denotes an analog signal. This signal is completely characterized by three parameters: A is the *amplitude* of the sinusoid, Ω is the *frequency* in radians per second (rad/s), and θ is the *phase* in radians. Instead of Ω, we often use the frequency F in cycles per second or hertz (Hz), where

$$\Omega = 2\pi F \tag{1.3.2}$$

In terms of F, (1.3.1) can be written as

$$x_a(t) = A\cos(2\pi Ft + \theta), \quad -\infty < t < \infty \tag{1.3.3}$$

We will use both forms, (1.3.1) and (1.3.3), in representing sinusoidal signals.

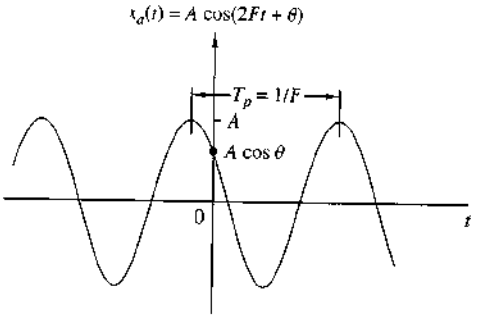


Figure 1.3.1
Example of an analog sinusoidal signal.

The analog sinusoidal signal in (1.3.3) is characterized by the following properties:

A1. For every fixed value of the frequency F, $x_a(t)$ is periodic. Indeed, it can easily be shown, using elementary trigonometry, that

$$x_a(t + T_p) = x_a(t)$$

where $T_p = 1/F$ is the fundamental period of the sinusoidal signal.

A2. Continuous-time sinusoidal signals with distinct (different) frequencies are themselves distinct.

A3. Increasing the frequency F results in an increase in the rate of oscillation of the signal, in the sense that more periods are included in a given time interval.

We observe that for $F = 0$, the value $T_p = \infty$ is consistent with the fundamental relation $F = 1/T_p$. Due to continuity of the time variable t, we can increase the frequency F, without limit, with a corresponding increase in the rate of oscillation.

The relationships we have described for sinusoidal signals carry over to the class of complex exponential signals

$$x_a(t) = Ae^{j(\Omega t+\theta)} \tag{1.3.4}$$

This can easily be seen by expressing these signals in terms of sinusoids using the Euler identity

$$e^{\pm j\phi} = \cos\phi \pm j \sin\phi \tag{1.3.5}$$

By definition, frequency is an inherently positive physical quantity. This is obvious if we interpret frequency as the number of cycles per unit time in a periodic signal. However, in many cases, only for mathematical convenience, we need to introduce negative frequencies. To see this we recall that the sinusoidal signal (1.3.1) may be expressed as

$$x_a(t) = A\cos(\Omega t + \theta) = \frac{A}{2} e^{j(\Omega t+\theta)} + \frac{A}{2} e^{-j(\Omega t+\theta)} \tag{1.3.6}$$

which follows from (1.3.5). Note that a sinusoidal signal can be obtained by adding two equal-amplitude complex-conjugate exponential signals, sometimes called phasors, illustrated in Fig. 1.3.2. As time progresses the phasors rotate in opposite directions with angular frequencies $\pm\Omega$ radians per second. Since a *positive frequency* corresponds to counterclockwise uniform angular motion, a *negative frequency* simply corresponds to clockwise angular motion.

For mathematical convenience, we use both negative and positive frequencies throughout this book. Hence the frequency range for analog sinusoids is $-\infty < F < \infty$.

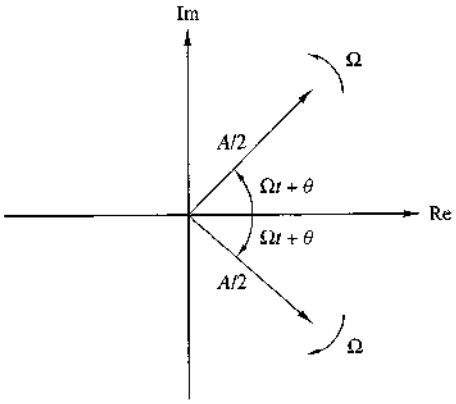


Figure 1.3.2
Representation of a cosine function by a pair of complex-conjugate exponentials (phasors).

1.3.2 Discrete-Time Sinusoidal Signals

A discrete-time sinusoidal signal may be expressed as

$$x(n) = A\cos(\omega n + \theta), \quad -\infty < n < \infty \tag{1.3.7}$$

where n is an integer variable, called the sample number, A is the *amplitude* of the sinusoid, ω is the *frequency* in radians per sample, and θ is the *phase* in radians.

If instead of ω we use the frequency variable f defined by

$$\omega \equiv 2\pi f \tag{1.3.8}$$

the relation (1.3.7) becomes

$$x(n) = A\cos(2\pi f n + \theta), \quad -\infty < n < \infty \tag{1.3.9}$$

The frequency f has dimensions of cycles per sample. In Section 1.4, where we consider the sampling of analog sinusoids, we relate the frequency variable f of a discrete-time sinusoid to the frequency F in cycles per second for the analog sinusoid. For the moment we consider the discrete-time sinusoid in (1.3.7) independently of the continuous-time sinusoid given in (1.3.1). Figure 1.3.3 shows a sinusoid with frequency $\omega = \pi/6$ radians per sample ($f = \frac{1}{12}$ cycles per sample) and phase $\theta = \pi/3$.

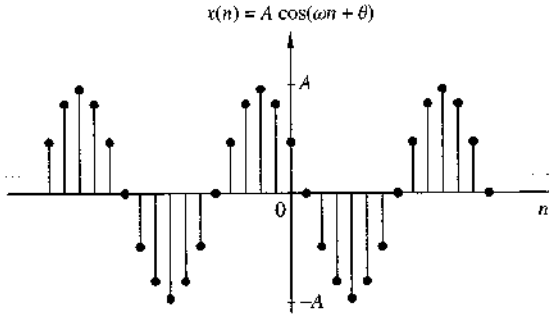


Figure 1.3.3
Example of a discrete-time sinusoidal signal ($\omega = \pi/6$ and $\theta = \pi/3$).

In contrast to continuous-time sinusoids, the discrete-time sinusoids are characterized by the following properties:

B1. *A discrete-time sinusoid is periodic only if its frequency f is a rational number.*

By definition, a discrete-time signal $x(n)$ is periodic with period $N (N > 0)$ if and only if

$$x(n + N) = x(n) \qquad \text{for all } n \tag{1.3.10}$$

The smallest value of N for which (1.3.10) is true is called the *fundamental period.*

The proof of the periodicity property is simple. For a sinusoid with frequency f_0 to be periodic, we should have

$$\cos[2\pi f_0(N + n) + \theta] = \cos(2\pi f_0 n + \theta)$$

This relation is true if and only if there exists an integer k such that

$$2\pi f_0 N = 2k\pi$$

or, equivalently,

$$f_0 = \frac{k}{N} \tag{1.3.11}$$

According to (1.3.11), a discrete-time sinusoidal signal is periodic only if its frequency f_0 can be expressed as the ratio of two integers (i.e., f_0 is rational).

To determine the fundamental period N of a periodic sinusoid, we express its frequency f_0 as in (1.3.11) and cancel common factors so that k and N are relatively prime. Then the fundamental period of the sinusoid is equal to N. Observe that a small change in frequency can result in a large change in the period. For example, note that $f_1 = 31/60$ implies that $N_1 = 60$, whereas $f_2 = 30/60$ results in $N_2 = 2$.

B2. *Discrete-time sinusoids whose frequencies are separated by an integer multiple of 2π are identical.*

To prove this assertion, let us consider the sinusoid $\cos(\omega_0 n + \theta)$. It easily follows that

$$\cos[(\omega_0 + 2\pi)n + \theta] = \cos(\omega_0 n + 2\pi n + \theta) = \cos(\omega_0 n + \theta) \tag{1.3.12}$$

As a result, all sinusoidal sequences

$$x_k(n) = A\cos(\omega_k n + \theta), \qquad k = 0, 1, 2, \ldots \tag{1.3.13}$$

where

$$\omega k = \omega_0 + 2k\pi, \qquad -\pi \le \omega_0 \le \pi$$

are *indistinguishable* (i.e., *identical*). Any sequence resulting from a sinusoid with a frequency $|\omega| > \pi$, or $|f| > \frac{1}{2}$, is identical to a sequence obtained from a sinusoidal signal with frequency $|\omega| < \pi$. Because of this similarity, we call the sinusoid having the frequency $|\omega| > \pi$ an *alias* of a corresponding sinusoid with frequency $|\omega| < \pi$. Thus we regard frequencies in the range $-\pi \le \omega \le \pi$, or $-\frac{1}{2} \le f \le \frac{1}{2}$, as unique

and all frequencies $|\omega| > \pi$, or $|f| > \frac{1}{2}$, as aliases. The reader should notice the difference between discrete-time sinusoids and continuous-time sinusoids, where the latter result in distinct signals for Ω or F in the entire range $-\infty < \Omega < \infty$ or $-\infty < F < \infty$.

B3. *The highest rate of oscillation in a discrete-time sinusoid is attained when $\omega = \pi$ (or $\omega = -\pi$) or, equivalently, $f = \frac{1}{2}$ (or $f = -\frac{1}{2}$).*

To illustrate this property, let us investigate the characteristics of the sinusoidal signal sequence

$$x(n) = \cos \omega_0 n$$

when the frequency varies from 0 to π. To simplify the argument, we take values of $\omega_0 = 0, \pi/8, \pi/4, \pi/2, \pi$ corresponding to $f = 0, \frac{1}{16}, \frac{1}{8}, \frac{1}{4}, \frac{1}{2}$, which result in periodic sequences having periods $N = \infty, 16, 8, 4, 2$, as depicted in Fig. 1.3.4. We note that the period of the sinusoid decreases as the frequency increases. In fact, we can see that the rate of oscillation increases as the frequency increases.

To see what happens for $\pi \leq \omega_0 \leq 2\pi$, we consider the sinusoids with frequencies $\omega_1 = \omega_0$ and $\omega_2 = 2\pi - \omega_0$. Note that as ω_1 varies from π to 2π, ω_2 varies from π to 0. It can be easily seen that

$$\begin{aligned} x_1(n) &= A \cos \omega_1 n = A \cos \omega_0 n \\ x_2(n) &= A \cos \omega_2 n = A \cos(2\pi - \omega_0) n \\ &= A \cos(-\omega_0 n) = x_1(n) \end{aligned} \tag{1.3.14}$$

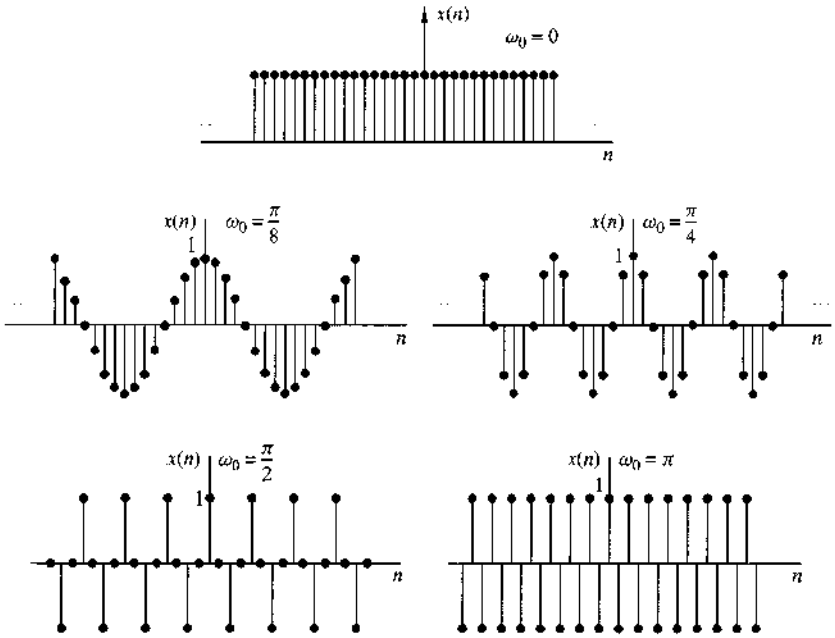


Figure 1.3.4 Signal $x(n) = \cos \omega_0 n$ for various values of the frequency ω_0.

Hence ω_2 is an alias of ω_1. If we had used a sine function instead of a cosine function, the result would basically be the same, except for a 180° phase difference between the sinusoids $x_1(n)$ and $x_2(n)$. In any case, as we increase the relative frequency ω_0 of a discrete-time sinusoid from π to 2π, its rate of oscillation decreases. For $\omega_0 = 2\pi$ the result is a constant signal, as in the case for $\omega_0 = 0$. Obviously, for $\omega_0 = \pi$ (or $f = \frac{1}{2}$) we have the highest rate of oscillation.

As for the case of continuous-time signals, negative frequencies can be introduced as well for discrete-time signals. For this purpose we use the identity

$$x(n) = A\cos(\omega n + \theta) = \frac{A}{2}e^{j(\omega n+\theta)} + \frac{A}{2}e^{-j(\omega n+\theta)} \tag{1.3.15}$$

Since discrete-time sinusoidal signals with frequencies that are separated by an integer multiple of 2π are identical, it follows that the frequencies in any interval $\omega_1 \le \omega \le \omega_1 + 2\pi$ constitute *all* the existing discrete-time sinusoids or complex exponentials. Hence the frequency range for discrete-time sinusoids is finite with duration 2π. Usually, we choose the range $0 \le \omega \le 2\pi$ or $-\pi \le \omega \le \pi$ ($0 \le f \le 1$, $-\frac{1}{2} \le f \le \frac{1}{2}$), which we call the *fundamental range.*

1.3.3 Harmonically Related Complex Exponentials

Sinusoidal signals and complex exponentials play a major role in the analysis of signals and systems. In some cases we deal with sets of *harmonically related* complex exponentials (or sinusoids). These are sets of periodic complex exponentials with fundamental frequencies that are multiples of a single positive frequency. Although we confine our discussion to complex exponentials, the same properties clearly hold for sinusoidal signals. We consider harmonically related complex exponentials in both continuous time and discrete time.

Continuous-time exponentials. The basic signals for continuous-time, harmonically related exponentials are

$$s_k(t) = e^{jk\Omega_0 t} = e^{j2\pi kF_0 t} \qquad k = 0, \pm 1, \pm 2, \ldots \tag{1.3.16}$$

We note that for each value of k, $s_k(t)$ is periodic with fundamental period $1/(kF_0) = T_p/k$ or fundamental frequency kF_0. Since a signal that is periodic with period T_p/k is also periodic with period $k(T_p/k) = T_p$ for any positive integer k, we see that all of the $s_k(t)$ have a common period of T_p. Furthermore, according to Section 1.3.1, F_0 is allowed to take any value and all members of the set are distinct, in the sense that if $k_1 \neq k_2$, then $s_{k1}(t) \neq s_{k2}(t)$.

From the basic signals in (1.3.16) we can construct a linear combination of harmonically related complex exponentials of the form

$$x_a(t) = \sum_{k=-\infty}^{\infty} c_k s_k(t) = \sum_{k=-\infty}^{\infty} c_k e^{jk\Omega_0 t} \tag{1.3.17}$$

where c_k, $k = 0, \pm 1, \pm 2, \ldots$ are arbitrary complex constants. The signal $x_a(t)$ is periodic with fundamental period $T_p = 1/F_0$, and its representation in terms of

(1.3.17) is called the *Fourier series* expansion for $x_a(t)$. The complex-valued constants are the Fourier series coefficients and the signal $s_k(t)$ is called the kth harmonic of $x_a(t)$.

Discrete-time exponentials. Since a discrete-time complex exponential is periodic if its relative frequency is a rational number, we choose $f_0 = 1/N$ and we define the sets of harmonically related complex exponentials by

$$s_k(n) = e^{j2\pi k f_0 n}, \qquad k = 0, \pm 1, \pm 2, \ldots \tag{1.3.18}$$

In contrast to the continuous-time case, we note that

$$s_{k+N}(n) = e^{j2\pi n(k+N)/N} = e^{j2\pi n} s_k(n) = s_k(n)$$

This means that, consistent with (1.3.10), there are only N distinct periodic complex exponentials in the set described by (1.3.18). Furthermore, all members of the set have a common period of N samples. Clearly, we can choose any consecutive N complex exponentials, say from $k = n_0$ to $k = n_0 + N - 1$, to form a harmonically related set with fundamental frequency $f_0 = 1/N$. Most often, for convenience, we choose the set that corresponds to $n_0 = 0$, that is, the set

$$s_k(n) = e^{j2\pi kn/N}, \qquad k = 0, 1, 2, \ldots, N-1 \tag{1.3.19}$$

As in the case of continuous-time signals, it is obvious that the linear combination

$$x(n) = \sum_{k=0}^{N-1} c_k s_k(n) = \sum_{k=0}^{N-1} c_k e^{j2\pi kn/N} \tag{1.3.20}$$

results in a periodic signal with fundamental period N. As we shall see later, this is the Fourier series representation for a periodic discrete-time sequence with Fourier coefficients $\{c_k\}$. The sequence $s_k(n)$ is called the kth harmonic of $x(n)$.

EXAMPLE 1.3.1

Stored in the memory of a digital signal processor is one cycle of the sinusoidal signal

$$x(n) = \sin\left(\frac{2\pi n}{N} + \theta\right)$$

where $\theta = 2\pi q/N$, where q and N are integers

(a) Determine how this table of values can be used to obtain values of harmonically related sinusoids having the same phase

(b) Determine how this table can be used to obtain sinusoids of the same frequency but different phase.

Solution.

(a) Let $x_k(n)$ denote the sinusoidal signal sequence

$$x_k(n) = \sin\left(\frac{2\pi nk}{N} + \theta\right)$$

This is a sinusoid with frequency $f_k = k/N$, which is harmonically related to $x(n)$. But $x_k(n)$ may be expressed as

$$x_k(n) = \sin\left[\frac{2\pi(kn)}{N} + \theta\right]$$
$$= x(kn)$$

Thus we observe that $x_k(0) = x(0)$, $x_k(1) = x(k)$, $x_k(2) = x(2k)$, and so on. Hence the sinusoidal sequence $x_k(n)$ can be obtained from the table of values of $x(n)$ by taking every kth value of $x(n)$, beginning with $x(0)$. In this manner we can generate the values of all harmonically related sinusoids with frequencies $f_k = k/N$ for $k = 0, 1, \ldots, N-1$.

(b) We can control the phase θ of the sinusoid with frequency $f_k = k/N$ by taking the first value of the sequence from memory location $q = \theta N/2\pi$, where q is an integer. Thus the initial phase θ controls the starting location in the table and we wrap around the table each time the index (kn) exceeds N.

1.4 Analog-to-Digital and Digital-to-Analog Conversion

Most signals of practical interest, such as speech, biological signals, seismic signals, radar signals, sonar signals, and various communications signals such as audio and video signals, are analog. To process analog signals by digital means, it is first necessary to convert them into digital form, that is, to convert them to a sequence of numbers having finite precision. This procedure is called *analog-to-digital (A/D) conversion*, and the corresponding devices are called *A/D converters (ADCs)*.

Conceptually, we view A/D conversion as a three-step process. This process is illustrated in Fig. 1.4.1.

1. *Sampling.* This is the conversion of a continuous-time signal into a discrete-time signal obtained by taking "samples" of the continuous-time signal at discrete-time instants. Thus, if $x_a(t)$ is the input to the sampler, the output is $x_a(nT) \equiv x(n)$, where T is called the *sampling interval*.

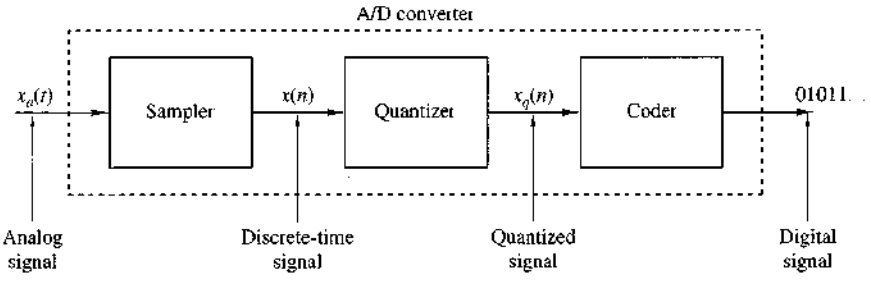


Figure 1.4.1 Basic parts of an analog-to-digital (A/D) converter.

2. *Quantization.* This is the conversion of a discrete-time continuous-valued signal into a discrete-time, discrete-valued (digital) signal. The value of each signal sample is represented by a value selected from a finite set of possible values. The difference between the unquantized sample $x(n)$ and the quantized output $x_q(n)$ is called the quantization error.
3. *Coding.* In the coding process, each discrete value $x_q(n)$ is represented by a b-bit binary sequence.

Although we model the A/D converter as a sampler followed by a quantizer and coder, in practice the A/D conversion is performed by a single device that takes $x_a(t)$ and produces a binary-coded number. The operations of sampling and quantization can be performed in either order but, in practice, sampling is always performed before quantization.

In many cases of practical interest (e.g., speech processing) it is desirable to convert the processed digital signals into analog form. (Obviously, we cannot listen to the sequence of samples representing a speech signal or see the numbers corresponding to a TV signal.) The process of converting a digital signal into an analog signal is known as *digital-to-analog (D/A) conversion.* All D/A converters "connect the dots" in a digital signal by performing some kind of interpolation, whose accuracy depends on the quality of the D/A conversion process. Figure 1.4.2 illustrates a simple form of D/A conversion, called a zero-order hold or a staircase approximation. Other approximations are possible, such as linearly connecting a pair of successive samples (linear interpolation), fitting a quadratic through three successive samples (quadratic interpolation), and so on. Is there an optimum (ideal) interpolator? For signals having a *limited frequency content* (finite bandwidth), the sampling theorem introduced in the following section specifies the optimum form of interpolation.

Sampling and quantization are treated in this section. In particular, we demonstrate that sampling does not result in a loss of information, nor does it introduce distortion in the signal if the signal bandwidth is finite. In principle, the analog signal can be reconstructed from the samples, provided that the sampling rate is sufficiently high to avoid the problem commonly called *aliasing.* On the other hand, quantization

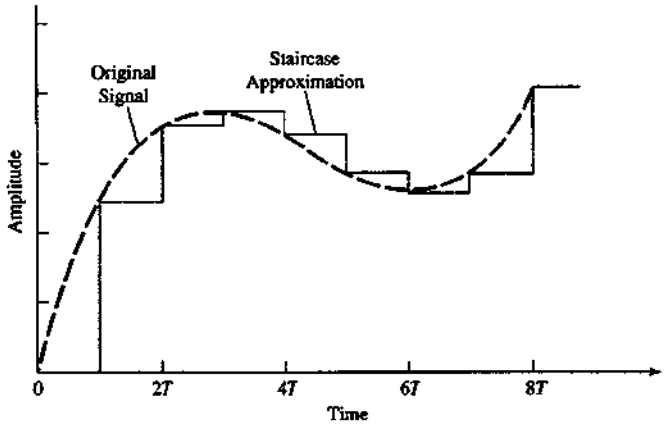


Figure 1.4.2
Zero-order hold digital-to-analog (D/A) conversion.

is a noninvertible or irreversible process that results in signal distortion. We shall show that the amount of distortion is dependent on the accuracy, as measured by the number of bits, in the A/D conversion process. The factors affecting the choice of the desired accuracy of the A/D converter are cost and sampling rate. In general, the cost increases with an increase in accuracy and/or sampling rate.

1.4.1 Sampling of Analog Signals

There are many ways to sample an analog signal. We limit our discussion to *periodic* or *uniform sampling*, which is the type of sampling used most often in practice. This is described by the relation

$$x(n) = x_a(nT), \qquad -\infty < n < \infty \tag{1.4.1}$$

where $x(n)$ is the discrete-time signal obtained by "taking samples" of the analog signal $x_a(t)$ every T seconds. This procedure is illustrated in Fig. 1.4.3. The time interval T between successive samples is called the *sampling period* or *sample interval* and its reciprocal $1/T = F_s$ is called the *sampling rate* (samples per second) or the *sampling frequency* (hertz).

Periodic sampling establishes a relationship between the time variables t and n of continuous-time and discrete-time signals, respectively. Indeed, these variables are linearly related through the sampling period T or, equivalently, through the sampling rate $F_s = 1/T$, as

$$t = nT = \frac{n}{F_s} \tag{1.4.2}$$

As a consequence of (1.4.2), there exists a relationship between the frequency variable F (or Ω) for analog signals and the frequency variable f (or ω) for discrete-time signals. To establish this relationship, consider an analog sinusoidal signal of the form

$$x_a(t) = A\cos(2\pi Ft + \theta) \tag{1.4.3}$$

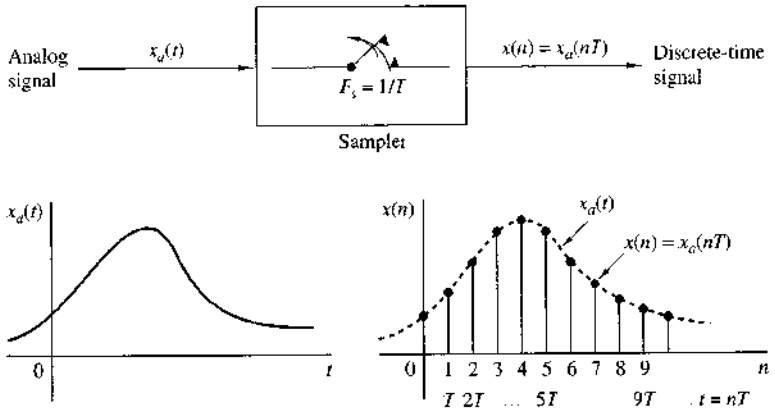


Figure 1.4.3 Periodic sampling of an analog signal.

which, when sampled periodically at a rate $F_s = 1/T$ samples per second, yields

$$\begin{aligned} x_a(nT) \equiv x(n) &= A\cos(2\pi FnT + \theta) \\ &= A\cos\left(\frac{2\pi nF}{F_s} + \theta\right) \end{aligned} \tag{1.4.4}$$

If we compare (1.4.4) with (1.3.9), we note that the frequency variables F and f are linearly related as

$$f = \frac{F}{F_s} \tag{1.4.5}$$

or, equivalently, as

$$\omega = \Omega T \tag{1.4.6}$$

The relation in (1.4.5) justifies the name *relative* or *normalized frequency*, which is sometimes used to describe the frequency variable f. As (1.4.5) implies, we can use f to determine the frequency F in hertz only if the sampling frequency F_s is known.

We recall from Section 1.3.1 that the ranges of the frequency variables F or Ω for continuous-time sinusoids are

$$\begin{aligned} -\infty < F < \infty \\ -\infty < \Omega < \infty \end{aligned} \tag{1.4.7}$$

However, the situation is different for discrete-time sinusoids. From Section 1.3.2 we recall that

$$\begin{aligned} -\frac{1}{2} < f < \frac{1}{2} \\ -\pi < \omega < \pi \end{aligned} \tag{1.4.8}$$

By substituting from (1.4.5) and (1.4.6) into (1.4.8), we find that the frequency of the continuous-time sinusoid when sampled at a rate $F_s = 1/T$ must fall in the range

$$-\frac{1}{2T} = -\frac{F_s}{2} \le F \le \frac{F_s}{2} = \frac{1}{2T} \tag{1.4.9}$$

or, equivalently,

$$-\frac{\pi}{T} = -\pi F_s \le \Omega \le \pi F_s = \frac{\pi}{T} \tag{1.4.10}$$

These relations are summarized in Table 1.1.

From these relations we observe that the fundamental difference between continuous-time and discrete-time signals is in their range of values of the frequency variables F and f, or Ω and ω. Periodic sampling of a continuous-time signal implies a mapping of the infinite frequency range for the variable F (or Ω) into a finite frequency range for the variable f (or ω). Since the highest frequency in a

TABLE 1.1 Relations Among Frequency Variables

Continuous-time signals		Discrete-time signals
$\Omega = 2\pi F$		$\omega = 2\pi f$
$\frac{\text{radians}}{\text{sec}}$ Hz		$\frac{\text{radians}}{\text{sample}}$ $\frac{\text{cycles}}{\text{sample}}$
	$\omega = \Omega T, f = F/F_s$	$-\pi \le \omega \le \pi$
		$-\frac{1}{2} \le f \le \frac{1}{2}$
	$\Omega = \omega/T, F = f \cdot F_s$	
$-\infty < \Omega < \infty$		$-\pi/T \le \Omega \le \pi/T$
$-\infty < F < \infty$		$-F_2/2 \le F \le F_s/2$

discrete-time signal is $\omega = \pi$ or $f = \frac{1}{2}$, it follows that, with a sampling rate F_s, the corresponding highest values of F and Ω are

$$\begin{aligned} F_{\max} &= \frac{F_s}{2} = \frac{1}{2T} \\ \Omega_{\max} &= \pi F_s = \frac{\pi}{T} \end{aligned} \tag{1.4.11}$$

Therefore, sampling introduces an ambiguity, since the highest frequency in a continuous-time signal that can be uniquely distinguished when such a signal is sampled at a rate $F_s = 1/T$ is $F_{\max} = F_s/2$, or $\Omega_{\max} = \pi F_s$. To see what happens to frequencies above $F_s/2$, let us consider the following example.

EXAMPLE 1.4.1

The implications of these frequency relations can be fully appreciated by considering the two analog sinusoidal signals

$$\begin{aligned} x_1(t) &= \cos 2\pi(10)t \\ x_2(t) &= \cos 2\pi(50)t \end{aligned} \tag{1.4.12}$$

which are sampled at a rate $F_s = 40$ Hz. The corresponding discrete-time signals or sequences are

$$\begin{aligned} x_1(n) &= \cos 2\pi \left(\frac{10}{40}\right) n = \cos \frac{\pi}{2} n \\ x_2(n) &= \cos 2\pi \left(\frac{50}{40}\right) n = \cos \frac{5\pi}{2} n \end{aligned} \tag{1.4.13}$$

However, $\cos 5\pi n/2 = \cos(2\pi n + \pi n/2) = \cos \pi n/2$. Hence $x_2(n) = x_1(n)$. Thus the sinusoidal signals are identical and, consequently, indistinguishable. If we are given the sampled values generated by $\cos(\pi/2)n$, there is some ambiguity as to whether these sampled values correspond to $x_1(t)$ or $x_2(t)$. Since $x_2(t)$ yields exactly the same values as $x_1(t)$ when the two are sampled at $F_s = 40$ samples per second, we say that the frequency $F_2 = 50$ Hz is an *alias* of the frequency $F_1 = 10$ Hz at the sampling rate of 40 samples per second.

It is important to note that F_2 is not the only alias of F_1. In fact at the sampling rate of 40 samples per second, the frequency $F_3 = 90$ Hz is also an alias of F_1, as is the frequency $F_4 = 130$ Hz, and so on. All of the sinusoids $\cos 2\pi(F_1 + 40k)t$, $k = 1, 2, 3, 4, \ldots$, sampled at 40 samples per second, yield identical values. Consequently, they are all aliases of $F_1 = 10$ Hz.

In general, the sampling of a continuous-time sinusoidal signal

$$x_a(t) = A\cos(2\pi F_0 t + \theta) \tag{1.4.14}$$

with a sampling rate $F_s = 1/T$ results in a discrete-time signal

$$x(n) = A\cos(2\pi f_0 n + \theta) \tag{1.4.15}$$

where $f_0 = F_0/F_s$ is the relative frequency of the sinusoid. If we assume that $-F_s/2 \le F_0 \le F_s/2$, the frequency f_0 of $x(n)$ is in the range $-\frac{1}{2} \le f_0 \le \frac{1}{2}$, which is the frequency range for discrete-time signals. In this case, the relationship between F_0 and f_0 is one-to-one, and hence it is possible to identify (or reconstruct) the analog signal $x_a(t)$ from the samples $x(n)$.

On the other hand, if the sinusoids

$$x_a(t) = A\cos(2\pi F_k t + \theta) \tag{1.4.16}$$

where

$$F_k = F_0 + kF_s, \qquad k = \pm 1, \pm 2, \ldots \tag{1.4.17}$$

are sampled at a rate F_s, it is clear that the frequency F_k is outside the fundamental frequency range $-F_s/2 \le F \le F_s/2$. Consequently, the sampled signal is

$$\begin{aligned} x(n) \equiv x_a(nT) &= A\cos\left(2\pi \frac{F_0 + kF_s}{F_s} n + \theta\right) \\ &= A\cos(2\pi n F_0/F_s + \theta + 2\pi kn) \\ &= A\cos(2\pi f_0 n + \theta) \end{aligned}$$

which is identical to the discrete-time signal in (1.4.15) obtained by sampling (1.4.14). Thus an infinite number of continuous-time sinusoids is represented by sampling the *same* discrete-time signal (i.e., by the same set of samples). Consequently, if we are given the sequence $x(n)$, an ambiguity exists as to which continuous-time signal $x_a(t)$ these values represent. Equivalently, we can say that the frequencies $F_k = F_0 + kF_s$, $-\infty < k < \infty$ (k integer) are indistinguishable from the frequency F_0 after sampling and hence they are aliases of F_0. The relationship between the frequency variables of the continuous-time and discrete-time signals is illustrated in Fig. 1.4.4.

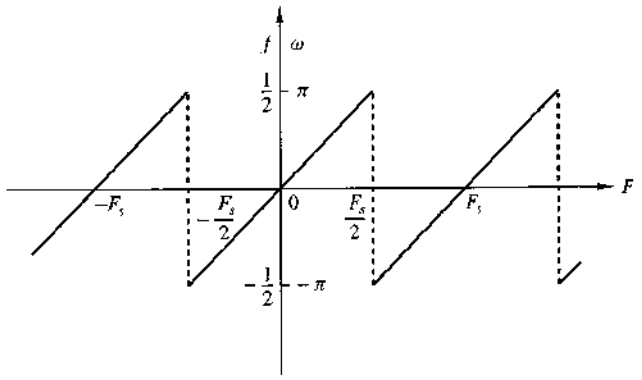


Figure 1.4.4
Relationship between the continuous-time and discrete-time frequency variables in the case of periodic sampling.

An example of aliasing is illustrated in Fig. 1.4.5, where two sinusoids with frequencies $F_0 = \frac{1}{8}$ Hz and $F_1 = -\frac{7}{8}$ Hz yield identical samples when a sampling rate of $F_s = 1$ Hz is used. From (1.4.17) it easily follows that for $k = -1$, $F_0 = F_1 + F_s = (-\frac{7}{8} + 1)$ Hz $= \frac{1}{8}$ Hz.

Since $F_s/2$, which corresponds to $\omega = \pi$, is the highest frequency that can be represented uniquely with a sampling rate F_s, it is a simple matter to determine the mapping of any (alias) frequency above $F_s/2$ ($\omega = \pi$) into the equivalent frequency below $F_s/2$. We can use $F_s/2$ or $\omega = \pi$ as the pivotal point and reflect or "fold" the alias frequency to the range $0 \le \omega \le \pi$. Since the point of reflection is $F_s/2$ ($\omega = \pi$), the frequency $F_s/2$ ($\omega = \pi$) is called the *folding frequency*.

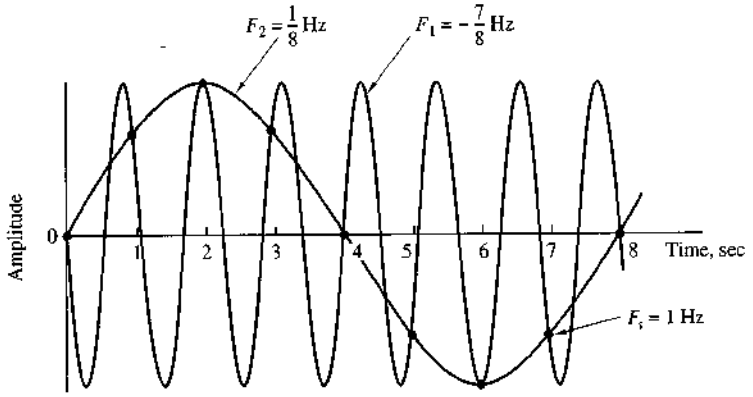


Figure 1.4.5 Illustration of aliasing.

EXAMPLE 1.4.2

Consider the analog signal

$$x_a(t) = 3\cos 100\pi t$$

(a) Determine the minimum sampling rate required to avoid aliasing.

(b) Suppose that the signal is sampled at the rate $F_s = 200$ Hz. What is the discrete-time signal obtained after sampling?

(c) Suppose that the signal is sampled at the rate $F_s = 75$ Hz. What is the discrete-time signal obtained after sampling?

(d) What is the frequency $0 < F < F_s/2$ of a sinusoid that yields samples identical to those obtained in part (c)?

Solution.

(a) The frequency of the analog signal is $F = 50$ Hz. Hence the minimum sampling rate required to avoid aliasing is $F_s = 100$ Hz.

(b) If the signal is sampled at $F_s = 200$ Hz, the discrete-time signal is

$$x(n) = 3\cos\frac{100\pi}{200}n = 3\cos\frac{\pi}{2}n$$

(c) If the signal is sampled at $F_s = 75$ Hz, the discrete-time signal is

$$\begin{aligned} x(n) &= 3\cos\frac{100\pi}{75}n = 3\cos\frac{4\pi}{3}n \\ &= 3\cos\left(2\pi - \frac{2\pi}{3}\right)n \\ &= 3\cos\frac{2\pi}{3}n \end{aligned}$$

(d) For the sampling rate of $F_s = 75$ Hz, we have

$$F = fF_s = 75f$$

The frequency of the sinusoid in part (c) is $f = \frac{1}{3}$. Hence

$$F = 25 \text{ Hz}$$

Clearly, the sinusoidal signal

$$\begin{aligned} y_a(t) &= 3\cos 2\pi Ft \\ &= 3\cos 50\pi t \end{aligned}$$

sampled at $F_s = 75$ samples/s yields identical samples. Hence $F = 50$ Hz is an alias of $F = 25$ Hz for the sampling rate $F_s = 75$ Hz.

1.4.2 The Sampling Theorem

Given any analog signal, how should we select the sampling period T or, equivalently, the sampling rate F_s? To answer this question, we must have some information about the characteristics of the signal to be sampled. In particular, we must have some general information concerning the *frequency content* of the signal. Such information is generally available to us. For example, we know generally that the major frequency components of a speech signal fall below 3000 Hz. On the other hand, television

signals, in general, contain important frequency components up to 5 MHz. The information content of such signals is contained in the amplitudes, frequencies, and phases of the various frequency components, but detailed knowledge of the characteristics of such signals is not available to us prior to obtaining the signals. In fact, the purpose of processing the signals is usually to extract this detailed information. However, if we know the maximum frequency content of the general class of signals (e.g., the class of speech signals, the class of video signals, etc.), we can specify the sampling rate necessary to convert the analog signals to digital signals.

Let us suppose that any analog signal can be represented as a sum of sinusoids of different amplitudes, frequencies, and phases, that is,

$$x_a(t) = \sum_{i=1}^{N} A_i \cos(2\pi F_i t + \theta_i) \tag{1.4.18}$$

where N denotes the number of frequency components. All signals, such as speech and video, lend themselves to such a representation over any short time segment. The amplitudes, frequencies, and phases usually change slowly with time from one time segment to another. However, suppose that the frequencies do not exceed some known frequency, say $F_{\max}$. For example, $F_{\max} = 3000$ Hz for the class of speech signals and $F_{\max} = 5$ MHz for television signals. Since the maximum frequency may vary slightly from different realizations among signals of any given class (e.g., it may vary slightly from speaker to speaker), we may wish to ensure that $F_{\max}$ does not exceed some predetermined value by passing the analog signal through a filter that severely attenuates frequency components above $F_{\max}$. Thus we are certain that no signal in the class contains frequency components (having significant amplitude or power) above $F_{\max}$. In practice, such filtering is commonly used prior to sampling.

From our knowledge of $F_{\max}$, we can select the appropriate sampling rate. We know that the highest frequency in an analog signal that can be unambiguously reconstructed when the signal is sampled at a rate $F_s = 1/T$ is $F_s/2$. Any frequency above $F_s/2$ or below $-F_s/2$ results in samples that are identical with a corresponding frequency in the range $-F_s/2 \le F \le F_s/2$. To avoid the ambiguities resulting from aliasing, we must select the sampling rate to be sufficiently high. That is, we must select $F_s/2$ to be greater than $F_{\max}$. Thus to avoid the problem of aliasing, F_s is selected so that

$$F_s > 2F_{\max} \tag{1.4.19}$$

where $F_{\max}$ is the largest frequency component in the analog signal. With the sampling rate selected in this manner, any frequency component, say $|F_i| < F_{\max}$, in the analog signal is mapped into a discrete-time sinusoid with a frequency

$$-\frac{1}{2} \le f_i = \frac{F_i}{F_s} \le \frac{1}{2} \tag{1.4.20}$$

or, equivalently,

$$-\pi \le \omega_i = 2\pi f_i \le \pi \tag{1.4.21}$$

Since, $|f| = \frac{1}{2}$ or $|\omega| = \pi$ is the highest (unique) frequency in a discrete-time signal, the choice of sampling rate according to (1.4.19) avoids the problem of aliasing.

In other words, the condition $F_s > 2F_{\max}$ ensures that all the sinusoidal components in the analog signal are mapped into corresponding discrete-time frequency components with frequencies in the fundamental interval. Thus all the frequency components of the analog signal are represented in sampled form without ambiguity, and hence the analog signal can be reconstructed without distortion from the sample values using an "appropriate" interpolation (digital-to-analog conversion) method. The "appropriate" or ideal interpolation formula is specified by the *sampling theorem*.

Sampling Theorem. If the highest frequency contained in an analog signal $x_a(t)$ is $F_{\max} = B$ and the signal is sampled at a rate $F_s > 2F_{\max} \equiv 2B$, then $x_a(t)$ can be exactly recovered from its sample values using the interpolation function

$$g(t) = \frac{\sin 2\pi Bt}{2\pi Bt} \tag{1.4.22}$$

Thus $x_a(t)$ may be expressed as

$$x_a(t) = \sum_{n=-\infty}^{\infty} x_a\left(\frac{n}{F_s}\right) g\left(t - \frac{n}{F_s}\right) \tag{1.4.23}$$

where $x_a(n/F_s) = x_a(nT) \equiv x(n)$ are the samples of $x_a(t)$.

When the sampling of $x_a(t)$ is performed at the minimum sampling rate $F_s = 2B$, the reconstruction formula in (1.4.23) becomes

$$x_a(t) = \sum_{n=-\infty}^{\infty} x_a\left(\frac{n}{2B}\right) \frac{\sin 2\pi B(t - n/2B)}{2\pi B(t - n/2B)} \tag{1.4.24}$$

The sampling rate $F_N = 2B = 2F_{\max}$ is called the *Nyquist rate*. Figure 1.4.6 illustrates the ideal D/A conversion process using the interpolation function in (1.4.22).

As can be observed from either (1.4.23) or (1.4.24), the reconstruction of $x_a(t)$ from the sequence $x(n)$ is a complicated process, involving a weighted sum of the interpolation function $g(t)$ and its time-shifted versions $g(t - nT)$ for $-\infty < n < \infty$, where the weighting factors are the samples $x(n)$. Because of the complexity and the infinite number of samples required in (1.4.23) or (1.4.24), these reconstruction formulas are primarily of theoretical interest. Practical interpolation methods are given in Chapter 6.

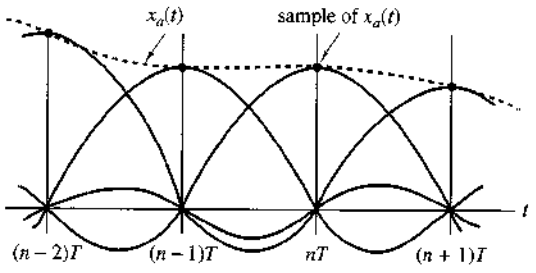


Figure 1.4.6
Ideal D/A conversion (interpolation).

EXAMPLE 1.4.3

Consider the analog signal

$$x_a(t) = 3\cos 50\pi t + 10\sin 300\pi t - \cos 100\pi t$$

What is the Nyquist rate for this signal?

Solution. The frequencies present in the signal above are

$$F_1 = 25 \text{ Hz}, \qquad F_2 = 150 \text{ Hz}, \qquad F_3 = 50 \text{ Hz}$$

Thus $F_{\max} = 150$ Hz and according to (1.4.19),

$$F_s > 2F_{\max} = 300 \text{ Hz}$$

The Nyquist rate is $F_N = 2F_{\max}$. Hence

$$F_N = 300 \text{ Hz}$$

Discussion. It should be observed that the signal component $10\sin 300\pi t$, sampled at the Nyquist rate $F_N = 300$, results in the samples $10\sin \pi n$, which are identically zero. In other words, we are sampling the analog sinusoid at its zero-crossing points, and hence we miss this signal component completely. This situation does not occur if the sinusoid is offset in phase by some amount θ. In such a case we have $10\sin(300\pi t + \theta)$ sampled at the Nyquist rate $F_N = 300$ samples per second, which yields the samples

$$\begin{aligned} 10\sin(\pi n + \theta) &= 10(\sin \pi n \cos\theta + \cos \pi n \sin\theta) \\ &= 10\sin\theta \cos \pi n \\ &= (-1)^n 10\sin\theta \end{aligned}$$

Thus if $\theta \neq 0$ or π, the samples of the sinusoid taken at the Nyquist rate are not all zero. However, we still cannot obtain the correct amplitude from the samples when the phase θ is unknown. A simple remedy that avoids this potentially troublesome situation is to sample the analog signal at a rate higher than the Nyquist rate.

EXAMPLE 1.4.4

Consider the analog signal

$$x_a(t) = 3\cos 2000\pi t + 5\sin 6000\pi t + 10\cos 12{,}000\pi t$$

(a) What is the Nyquist rate for this signal?

(b) Assume now that we sample this signal using a sampling rate $F_s = 5000$ samples/s. What is the discrete-time signal obtained after sampling?

(c) What is the analog signal $y_a(t)$ that we can reconstruct from the samples if we use ideal interpolation?

Solution.

(a) The frequencies existing in the analog signal are

$$F_1 = 1\text{ kHz}, \qquad F_2 = 3\text{ kHz}, \qquad F_3 = 6\text{ kHz}$$

Thus $F_{\max} = 6$ kHz, and according to the sampling theorem,

$$F_s > 2F_{\max} = 12\text{ kHz}$$

The Nyquist rate is

$$F_N = 12\text{ kHz}$$

(b) Since we have chosen $F_s = 5$ kHz, the folding frequency is

$$\frac{F_s}{2} = 2.5\text{ kHz}$$

and this is the maximum frequency that can be represented uniquely by the sampled signal. By making use of (1.4.2) we obtain

$$\begin{aligned} x(n) = x_a(nT) &= x_a\left(\frac{n}{F_s}\right) \\ &= 3\cos 2\pi\left(\tfrac{1}{5}\right)n + 5\sin 2\pi\left(\tfrac{3}{5}\right)n + 10\cos 2\pi\left(\tfrac{6}{5}\right)n \\ &= 3\cos 2\pi\left(\tfrac{1}{5}\right)n + 5\sin 2\pi\left(1-\tfrac{2}{5}\right)n + 10\cos 2\pi\left(1+\tfrac{1}{5}\right)n \\ &= 3\cos 2\pi\left(\tfrac{1}{5}\right)n + 5\sin 2\pi\left(-\tfrac{2}{5}\right)n + 10\cos 2\pi\left(\tfrac{1}{5}\right)n \end{aligned}$$

Finally, we obtain

$$x(n) = 13\cos 2\pi\left(\tfrac{1}{5}\right)n - 5\sin 2\pi\left(\tfrac{2}{5}\right)n$$

The same result can be obtained using Fig. 1.4.4. Indeed, since $F_s = 5$ kHz, the folding frequency is $F_s/2 = 2.5$ kHz. This is the maximum frequency that can be represented uniquely by the sampled signal. From (1.4.17) we have $F_0 = F_k - kF_s$. Thus F_0 can be obtained by subtracting from F_k an integer multiple of F_s such that $-F_s/2 \le F_0 \le F_s/2$. The frequency F_1 is less than $F_s/2$ and thus it is not affected by aliasing. However, the other two frequencies are above the folding frequency and they will be changed by the aliasing effect. Indeed,

$$F_2' = F_2 - F_s = -2\text{ kHz}$$

$$F_3' = F_3 - F_s = 1\text{ kHz}$$

From (1.4.5) it follows that $f_1 = \frac{1}{5}$, $f_2 = -\frac{2}{5}$, and $f_3 = \frac{1}{5}$, which are in agreement with the result above.

(c) Since the frequency components at only 1 kHz and 2 kHz are present in the sampled signal, the analog signal we can recover is

$$ya(t) = 13\cos 2000\pi t - 5\sin 4000\pi t$$

which is obviously different from the original signal $x_a(t)$. This distortion of the original analog signal was caused by the aliasing effect, due to the low sampling rate used.

Although aliasing is a pitfall to be avoided, there are two useful practical applications based on the exploitation of the aliasing effect. These applications are the stroboscope and the sampling oscilloscope. Both instruments are designed to operate as aliasing devices in order to represent high frequencies as low frequencies.

To elaborate, consider a signal with high-frequency components confined to a given frequency band $B_1 < F < B_2$, where $B_2 - B_1 \equiv B$ is defined as the bandwidth of the signal. We assume that $B << B_1 < B_2$. This condition means that the frequency components in the signal are much larger than the bandwidth B of the signal. Such signals are usually called bandpass or narrowband signals. Now, if this signal is sampled at a rate $F_s \geq 2B$, but $F_s << B_1$, then all the frequency components contained in the signal will be aliases of frequencies in the range $0 < F < F_s/2$. Consequently, if we observe the frequency content of the signal in the fundamental range $0 < F < F_s/2$, we know precisely the frequency content of the analog signal since we know the frequency band $B_1 < F < B_2$ under consideration. Consequently, if the signal is a narrowband (bandpass) signal, we can reconstruct the original signal from the samples, provided that the signal is sampled at a rate $F_s > 2B$, where B is the bandwidth. This statement constitutes another form of the sampling theorem, which we call the *bandpass form* in order to distinguish it from the previous form of the sampling theorem, which applies in general to all types of signals. The latter is sometimes called the *baseband form*. The *bandpass form* of the sampling theorem is described in detail in Section 6.4.

1.4.3 Quantization of Continuous-Amplitude Signals

As we have seen, a digital signal is a sequence of numbers (samples) in which each number is represented by a finite number of digits (finite precision).

The process of converting a discrete-time continuous-amplitude signal into a digital signal by expressing each sample value as a finite (instead of an infinite) number of digits is called *quantization*. The error introduced in representing the continuous-valued signal by a finite set of discrete value levels is called *quantization error* or *quantization noise*.

We denote the quantizer operation on the samples $x(n)$ as $Q[x(n)]$ and let $x_q(n)$ denote the sequence of quantized samples at the output of the quantizer. Hence

$$x_q(n) = Q[x(n)]$$

Then the quantization error is a sequence $e_q(n)$ defined as the difference between the quantized value and the actual sample value. Thus

$$e_q(n) = x_q(n) - x(n) \tag{1.4.25}$$

We illustrate the quantization process with an example. Let us consider the discrete-time signal

$$x(n) = \begin{cases} 0.9^n, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

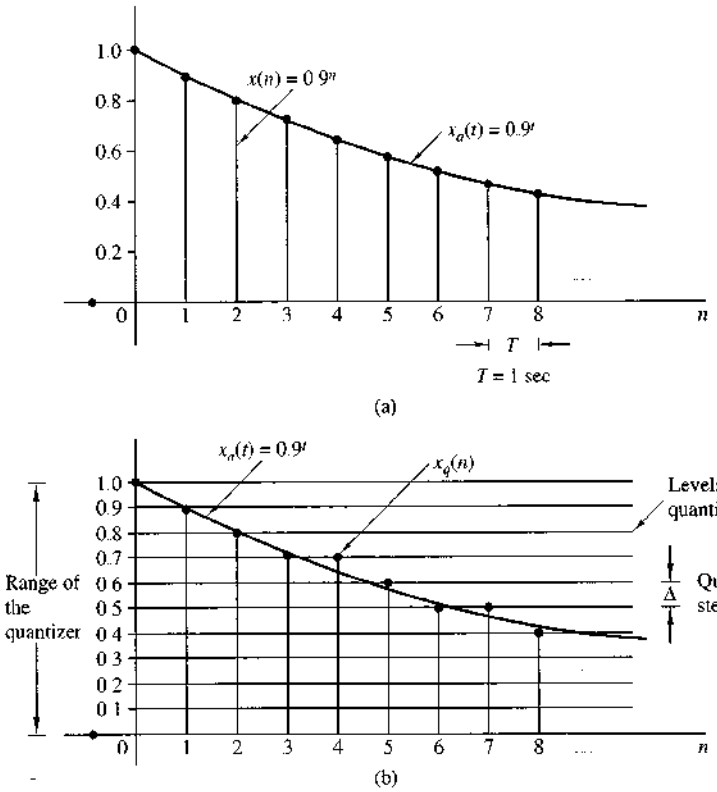


Figure 1.4.7 Illustration of quantization.

obtained by sampling the analog exponential signal $x_a(t) = 0.9^t$, $t \geq 0$ with a sampling frequency $F_s = 1$ Hz (see Fig. 1.4.7(a)). Observation of Table 1.2, which shows the values of the first 10 samples of $x(n)$, reveals that the description of the sample value $x(n)$ requires n significant digits. It is obvious that this signal cannot be processed by using a calculator or a digital computer since only the first few samples can be stored and manipulated. For example, most calculators process numbers with only eight significant digits.

However, let us assume that we want to use only one significant digit. To eliminate the excess digits, we can either simply discard them (*truncation*) or discard them by rounding the resulting number (*rounding*). The resulting quantized signals $x_q(n)$ are shown in Table 1.2. We discuss only quantization by rounding, although it is just as easy to treat truncation. The rounding process is graphically illustrated in Fig. 1.4.7(b). The values allowed in the digital signal are called the *quantization levels*, whereas the distance Δ between two successive quantization levels is called the *quantization step size* or *resolution*. The rounding quantizer assigns each sample of $x(n)$ to the nearest quantization level. In contrast, a quantizer that performs truncation would have assigned each sample of $x(n)$ to the quantization level below

TABLE 1.2 Numerical Illustration of Quantization with One Significant Digit Using Truncation or Rounding

n	$x(n)$ Discrete-time signal	$x_q(n)$ (Truncation)	$x_q(n)$ (Rounding)	$e_q(n) = x_q(n) - x(n)$ (Rounding)
0	1	1.0	1.0	0.0
1	0.9	0.9	0.9	0.0
2	0.81	0.8	0.8	−0.01
3	0.729	0.7	0.7	−0.029
4	0.6561	0.6	0.7	0.0439
5	0.59049	0.5	0.6	0.00951
6	0.531441	0.5	0.5	−0.031441
7	0.4782969	0.4	0.5	0.0217031
8	0.43046721	0.4	0.4	−0.03046721
9	0.387420489	0.3	0.4	0.012579511

it. The quantization error $e_q(n)$ in rounding is limited to the range of $-\Delta/2$ to $\Delta/2$, that is,

$$-\frac{\Delta}{2} \le e_q(n) \le \frac{\Delta}{2} \tag{1.4.26}$$

In other words, the instantaneous quantization error cannot exceed half of the quantization step (see Table 1.2).

If $x_{\min}$ and $x_{\max}$ represent the minimum and maximum values of $x(n)$ and L is the number of quantization levels, then

$$\Delta = \frac{x_{\max} - x_{\min}}{L - 1} \tag{1.4.27}$$

We define the *dynamic range* of the signal as $x_{\max} - x_{\min}$. In our example we have $x_{\max} = 1$, $x_{\min} = 0$, and $L = 11$, which leads to $\Delta = 0.1$. Note that if the dynamic range is fixed, increasing the number of quantization levels L results in a decrease of the quantization step size. Thus the quantization error decreases and the accuracy of the quantizer increases. In practice we can reduce the quantization error to an insignificant amount by choosing a sufficient number of quantization levels.

Theoretically, quantization of analog signals always results in a loss of information. This is a result of the ambiguity introduced by quantization. Indeed, quantization is an irreversible or noninvertible process (i.e., a many-to-one mapping) since all samples in a distance $\Delta/2$ about a certain quantization level are assigned the same value. This ambiguity makes the exact quantitative analysis of quantization extremely difficult. This subject is discussed further in Chapter 6, where we use statistical analysis.

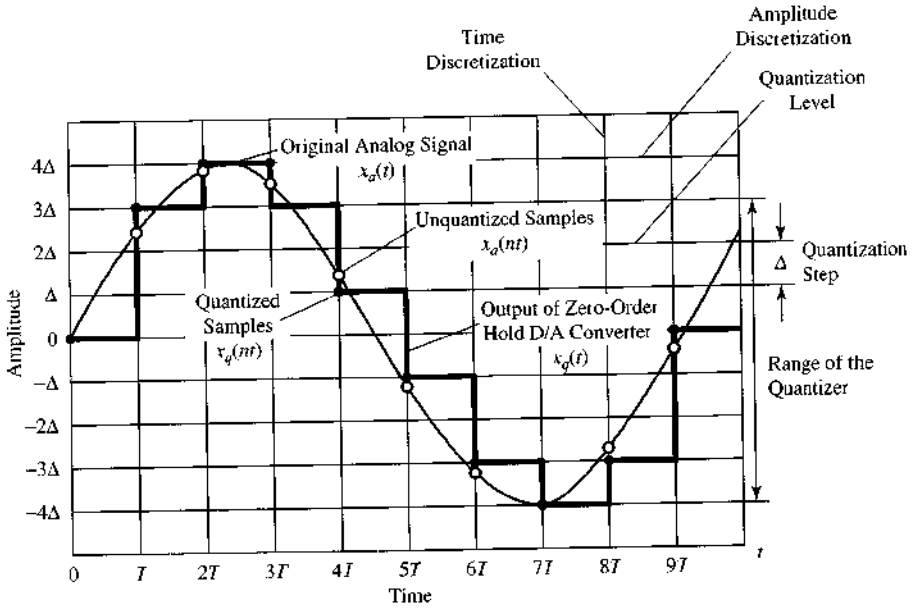


Figure 1.4.8 Sampling and quantization of a sinusoidal signal.

1.4.4 Quantization of Sinusoidal Signals

Figure 1.4.8 illustrates the sampling and quantization of an analog sinusoidal signal $x_a(t) = A\cos\Omega_0 t$ using a rectangular grid. Horizontal lines within the range of the quantizer indicate the allowed levels of quantization. Vertical lines indicate the sampling times. Thus, from the original analog signal $x_a(t)$ we obtain a discrete-time signal $x(n) = x_a(nT)$ by sampling and a discrete-time, discrete-amplitude signal $x_q(nT)$ after quantization. In practice, the staircase signal $x_q(t)$ can be obtained by using a zero-order hold. This analysis is useful because sinusoids are used as test signals in A/D converters.

If the sampling rate F_s satisfies the sampling theorem, quantization is the only error in the A/D conversion process.

Thus we can evaluate the quantization error by quantizing the analog signal $x_a(t)$ instead of the discrete-time signal $x(n) = x_a(nT)$. Inspection of Fig. 1.4.8 indicates that the signal $x_a(t)$ is almost linear between quantization levels (see Fig. 1.4.9). The

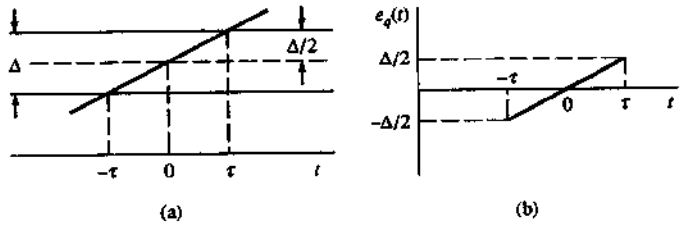


Figure 1.4.9 The quantization error $e_q(t) = x_a(t) - x_q(t)$.

corresponding quantization error $e_q(t) = x_a(t) - x_q(t)$ is shown in Fig. 1.4.9. In Fig. 1.4.9, τ denotes the time that $x_a(t)$ stays within the quantization levels. The mean-square error power P_q is

$$P_q = \frac{1}{2\tau}\int_{-\tau}^{\tau} e_q^2(t)\,dt = \frac{1}{\tau}\int_0^{\tau} e_q^2(t)\,dt \tag{1.4.28}$$

Since $e_q(t) = (\Delta/2\tau)t$, $-\tau \le t \le \tau$, we have

$$P_q = \frac{1}{\tau}\int_0^{\tau}\left(\frac{\Delta}{2\tau}\right)^2 t^2\,dt = \frac{\Delta^2}{12} \tag{1.4.29}$$

If the quantizer has b bits of accuracy and the quantizer covers the entire range $2A$, the quantization step is $\Delta = 2A/2^b$. Hence

$$P_q = \frac{A^2/3}{2^{2b}} \tag{1.4.30}$$

The average power of the signal $x_a(t)$ is

$$P_x = \frac{1}{T_p}\int_0^{T_p}(A\cos\Omega_0 t)^2\,dt = \frac{A^2}{2} \tag{1.4.31}$$

The quality of the output of the A/D converter is usually measured by the *signal-to-quantization noise ratio (SQNR)*, which provides the ratio of the signal power to the noise power:

$$\text{SQNR} = \frac{P_x}{P_q} = \frac{3}{2}\cdot 2^{2b}$$

Expressed in decibels (dB), the SQNR is

$$\text{SQNR(dB)} = 10\log_{10}\text{SQNR} = 1.76 + 6.02b \tag{1.4.32}$$

This implies that the SQNR increases approximately 6 dB for every bit added to the word length, that is, for each doubling of the quantization levels.

Although formula (1.4.32) was derived for sinusoidal signals, we shall see in Chapter 6 that a similar result holds for every signal whose dynamic range spans the range of the quantizer. This relationship is extremely important because it dictates the number of bits required by a specific application to assure a given signal-to-noise ratio. For example, most compact disc players use a sampling frequency of 44.1 kHz and 16-bit sample resolution, which implies a SQNR of more than 96 dB.

1.4.5 Coding of Quantized Samples

The coding process in an A/D converter assigns a unique binary number to each quantization level. If we have L levels we need at least L different binary numbers. With a word length of b bits we can create 2^b different binary numbers. Hence we have $2^b \ge L$, or equivalently, $b \ge \log_2 L$. Thus the number of bits required in the coder is the smallest integer greater than or equal to $\log_2 L$. In our example (Table 1.2) it can easily be seen that we need a coder with $b = 4$ bits. Commercially available A/D converters may be obtained with finite precision of $b = 16$ or less. Generally, the higher the sampling speed and the finer the quantization, the more expensive the device becomes.

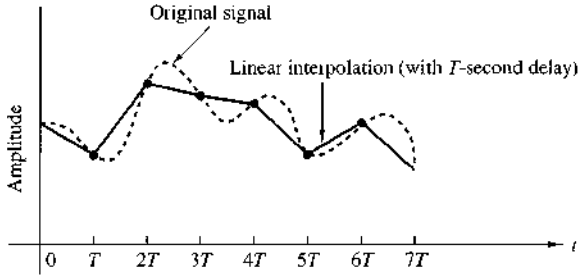


Figure 1.4.10
Linear point connector (with T-second delay).

1.4.6 Digital-to-Analog Conversion

To convert a digital signal into an analog signal we can use a digital-to-analog (D/A) converter. As stated previously, the task of a D/A converter is to interpolate between samples.

The sampling theorem specifies the optimum interpolation for a bandlimited signal. However, this type of interpolation is too complicated and, hence, impractical, as indicated previously. From a practical viewpoint, the simplest D/A converter is the zero-order hold shown in Fig. 1.4.2, which simply holds constant the value of one sample until the next one is received. Additional improvement can be obtained by using linear

interpolation as shown in Fig. 1.4.10 to connect successive samples with straight-line segments. Better interpolation can be achieved by using more sophisticated higher-order interpolation techniques.

In general, suboptimum interpolation techniques result in passing frequencies above the folding frequency. Such frequency components are undesirable and are usually removed by passing the output of the interpolator through a proper analog filter, which is called a *postfilter* or *smoothing filter*.

Thus D/A conversion usually involves a suboptimum interpolator followed by a postfilter. D/A converters are treated in more detail in Chapter 6.

1.4.7 Analysis of Digital Signals and Systems Versus Discrete-Time Signals and Systems

We have seen that a digital signal is defined as a function of an integer independent variable and its values are taken from a finite set of possible values. The usefulness of such signals is a consequence of the possibilities offered by digital computers. Computers operate on numbers, which are represented by a stringof 0's and 1's. The length of this string (*word length*) is fixed and finite and usually is 8, 12, 16, or 32 bits. The effects of finite word length in computations cause complications in the analysis of digital signal processing systems. To avoid these complications, we neglect the quantized nature of digital signals and systems in much of our analysis and consider them as discrete-time signals and systems.

In Chapters 6, 9, and 10 we investigate the consequences of using a finite word length. This is an important topic, since many digital signal processing problems are solved with small computers or microprocessors that employ fixed-point arithmetic.

Consequently, one must look carefully at the problem of finite-precision arithmetic and account for it in the design of software and hardware that performs the desired signal processing tasks.

1.5 Summary and References

In this introductory chapter we have attempted to provide the motivation for digital signal processing as an alternative to analog signal processing. We presented the basic elements of a digital signal processing system and defined the operations needed to convert an analog signal into a digital signal ready for processing. Of particular importance is the sampling theorem, which was introduced by Nyquist (1928) and later popularized in the classic paper by Shannon (1949). The sampling theorem as described in Section 1.4.2 is derived in Chapter 6. Sinusoidal signals were introduced primarily for the purpose of illustrating the aliasing phenomenon and for the subsequent development of the sampling theorem.

Quantization effects that are inherent in the A/D conversion of a signal were also introduced in this chapter. Signal quantization is best treated in statistical terms, as described in Chapters 6, 9, and 10.

Finally, the topic of signal reconstruction, or D/A conversion, was described briefly. Signal reconstruction based on staircase interpolation is treated in Section 6.3.

There are numerous practical applications of digital signal processing. The book edited by Oppenheim (1978) treats applications to speech processing, image processing, radar signal processing, sonar signal processing, and geophysical signal processing.

Problems

1.1 Classify the following signals according to whether they are (1) one- or multi-dimensional; (2) single or multichannel, (3) continuous time or discrete time, and (4) analog or digital (in amplitude). Give a brief explanation.

(a) Closing prices of utility stocks on the New York Stock Exchange.

(b) A color movie.

(c) Position of the steering wheel of a car in motion relative to car's reference frame.

(d) Position of the steering wheel of a car in motion relative to ground reference frame.

(e) Weight and height measurements of a child taken every month.

1.2 Determine which of the following sinusoids are periodic and compute their fundamental period.

(a) $\cos 0.01\pi n$ **(b)** $\cos\left(\pi \frac{30n}{105}\right)$ **(c)** $\cos 3\pi n$ **(d)** $\sin 3n$ **(e)** $\sin\left(\pi \frac{62n}{10}\right)$

1.3 Determine whether or not each of the following signals is periodic. In case a signal is periodic, specify its fundamental period

(a) $x_a(t) = 3\cos(5t + \pi/6)$

(b) $x(n) = 3\cos(5n + \pi/6)$

(c) $x(n) = 2\exp[j(n/6 - \pi)]$

(d) $x(n) = \cos(n/8)\cos(\pi n/8)$

(e) $x(n) = \cos(\pi n/2) - \sin(\pi n/8) + 3\cos(\pi n/4 + \pi/3)$

1.4 **(a)** Show that the fundamental period N_p of the signals

$$s_k(n) = e^{j2\pi kn/N}, \qquad k = 0, 1, 2, \ldots$$

is given by $N_p = N/\text{GCD}(k, N)$, where GCD is the greatest common divisor of k and N.

(b) What is the fundamental period of this set for $N = 7$?

(c) What is it for $N = 16$?

1.5 Consider the following analog sinusoidal signal:

$$x_a(t) = 3\sin(100\pi t)$$

(a) Sketch the signal $x_a(t)$ for $0 \le t \le 30$ ms.

(b) The signal $x_a(t)$ is sampled with a sampling rate $F_s = 300$ samples/s. Determine the frequency of the discrete-time signal $x(n) = x_a(nT)$, $T = 1/F_s$, and show that it is periodic.

(c) Compute the sample values in one period of $x(n)$. Sketch $x(n)$ on the same diagram with $x_a(t)$. What is the period of the discrete-time signal in milliseconds?

(d) Can you find a sampling rate F_s such that the signal $x(n)$ reaches its peak value of 3? What is the minimum F_s suitable for this task?

1.6 A continuous-time sinusoid $x_a(t)$ with fundamental period $T_p = 1/F_0$ is sampled at a rate $F_s = 1/T$ to produce a discrete-time sinusoid $x(n) = x_a(nT)$.

(a) Show that $x(n)$ is periodic if $T/T_p = k/N$ (i.e., T/T_p is a rational number).

(b) If $x(n)$ is periodic, what is its fundamental period T_p in seconds?

(c) Explain the statement: $x(n)$ is periodic if its fundamental period T_p, in seconds, is equal to an integer number of periods of $x_a(t)$.

1.7 An analog signal contains frequencies up to 10 kHz.

(a) What range of sampling frequencies allows exact reconstruction of this signal from its samples?

(b) Suppose that we sample this signal with a sampling frequency $F_s = 8$ kHz. Examine what happens to the frequency $F_1 = 5$ kHz.

(c) Repeat part (b) for a frequency $F_2 = 9$ kHz.

1.8 An analog electrocardiogram (ECG) signal contains useful frequencies up to 100 Hz.

(a) What is the Nyquist rate for this signal?

(b) Suppose that we sample this signal at a rate of 250 samples/s. What is the highest frequency that can be represented uniquely at this sampling rate?

1.9 An analog signal $x_a(t) = \sin(480\pi t) + 3\sin(720\pi t)$ is sampled 600 times per second.

(a) Determine the Nyquist sampling rate for $x_a(t)$.

(b) Determine the folding frequency.

(c) What are the frequencies, in radians, in the resulting discrete time signal $x(n)$?

(d) If $x(n)$ is passed through an ideal D/A converter, what is the reconstructed signal $y_a(t)$?

1.10 A digital communication link carries binary-coded words representing samples of an input signal

$$x_a(t) = 3\cos 600\pi t + 2\cos 1800\pi t$$

The link is operated at 10,000 bits/s and each input sample is quantized into 1024 different voltage levels.

(a) What are the sampling frequency and the folding frequency?

(b) What is the Nyquist rate for the signal $x_a(t)$?

(c) What are the frequencies in the resulting discrete-time signal $x(n)$?

(d) What is the resolution Δ?

1.11 Consider the simple signal processing system shown in Fig. P1.11. The sampling periods of the A/D and D/A converters are $T = 5$ ms and $T' = 1$ ms, respectively. Determine the output $y_a(t)$ of the system, if the input is

$$x_a(t) = 3\cos 100\pi t + 2\sin 250\pi t \qquad (t \text{ in seconds})$$

The postfilter removes any frequency component above $F_s/2$.

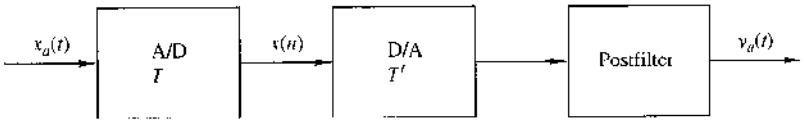


Figure P1.11

1.12 **(a)** Derive the expression for the discrete-time signal $x(n)$ in Example 1.4.2 using the periodicity properties of sinusoidal functions.

(b) What is the analog signal we can obtain from $x(n)$ if in the reconstruction process we assume that $F_s = 10$ kHz?

1.13 The discrete-time signal $x(n) = 6.35\cos(\pi/10)n$ is quantized with a resolution (a) $\Delta = 0.1$ or (b) $\Delta = 0.02$. How many bits are required in the A/D converter in each case?

1.14 Determine the bit rate and the resolution in the sampling of a seismic signal with dynamic range of 1 volt if the sampling rate is $F_s = 20$ samples/s and we use an 8-bit A/D converter. What is the maximum frequency that can be present in the resulting digital seismic signal?

1.15 *Sampling of sinusoidal signals: aliasing* Consider the following continuous-time sinusoidal signal

$$x_a(t) = \sin 2\pi F_0 t, \qquad -\infty < t < \infty$$

Since $x_a(t)$ is described mathematically, its sampled version can be described by values every T seconds. The sampled signal is described by the formula

$$x(n) = x_a(nT) = \sin 2\pi \frac{F_0}{F_s} n, \qquad -\infty < n < \infty$$

where $F_s = 1/T$ is the sampling frequency.

(a) Plot the signal $x(n)$, $0 \le n \le 99$ for $F_s = 5$ kHz and $F_0 = 0.5, 2, 3$, and 4.5 kHz. Explain the similarities and differences among the various plots.

(b) Suppose that $F_0 = 2$ kHz and $F_s = 50$ kHz.

1. Plot the signal $x(n)$. What is the frequency f_0 of the signal $x(n)$?
2. Plot the signal $y(n)$ created by taking the even-numbered samples of $x(n)$. Is this a sinusoidal signal? Why? If so, what is its frequency?

1.16 *Quantization error in A/D conversion of a sinusoidal signal* Let $x_q(n)$ be the signal obtained by quantizing the signal $x(n) = \sin 2\pi f_0 n$. The quantization error power P_q is defined by

$$P_q = \frac{1}{N} \sum_{n=0}^{N-1} e^2(n) = \frac{1}{N} \sum_{n=0}^{N-1} [x_q(n) - x(n)]^2$$

The "quality" of the quantized signal can be measured by the signal-to-quantization noise ratio (SQNR) defined by

$$\text{SQNR} = 10 \log_{10} \frac{P_x}{P_q}$$

where P_x is the power of the unquantized signal $x(n)$.

(a) For $f_0 = 1/50$ and $N = 200$, write a program to quantize the signal $x(n)$, using truncation, to 64, 128, and 256 quantization levels. In each case plot the signals $x(n)$, $x_q(n)$, and $e(n)$ and compute the corresponding SQNR.

(b) Repeat part (a) by using rounding instead of truncation.

(c) Comment on the results obtained in parts (a) and (b).

(d) Compare the experimentally measured SQNR with the theoretical SQNR predicted by formula (1.4.32) and comment on the differences and similarities.

CHAPTER

2

Discrete-Time Signals and Systems

In Chapter 1 we introduced the reader to a number of important types of signals and described the sampling process by which an analog signal is converted to a discrete-time signal. In addition, we presented in some detail the characteristics of discrete-time sinusoidal signals. The sinusoid is an important elementary signal that serves as a basic building block in more complex signals. However, there are other elementary signals that are important in our treatment of signal processing. These discrete-time signals are introduced in this chapter and are used as basis functions or building blocks to describe more complex signals.

The major emphasis in this chapter is the characterization of discrete-time systems in general and the class of linear time-invariant (LTI) systems in particular. A number of important time-domain properties of LTI systems are defined and developed, and an important formula, called the convolution formula, is derived which allows us to determine the output of an LTI system to any given arbitrary input signal. In addition to the convolution formula, difference equations are introduced as an alternative method for describing the input–output relationship of an LTI system, and in addition, recursive and nonrecursive realizations of LTI systems are treated.

Our motivation for the emphasis on the study of LTI systems is twofold. First, there is a large collection of mathematical techniques that can be applied to the analysis of LTI systems. Second, many practical systems are either LTI systems or can be approximated by LTI systems. Because of its importance in digital signal processing applications and its close resemblance to the convolution formula, we also introduce the correlation between two signals. The autocorrelation and crosscorrelation of signals are defined and their properties are presented.

2.1 Discrete-Time Signals

As we discussed in Chapter 1, a discrete-time signal $x(n)$ is a function of an independent variable that is an integer. It is graphically represented as in Fig. 2.1.1. It is important to note that a discrete-time signal is *not defined* at instants between two successive samples. Also, it is incorrect to think that $x(n)$ is equal to zero if n is not an integer. Simply, the signal $x(n)$ is not defined for noninteger values of n.

In the sequel we will assume that a discrete-time signal is defined for every integer value n for $-\infty < n < \infty$. By tradition, we refer to $x(n)$ as the "nth sample" of the signal even if the signal $x(n)$ is inherently discrete time (i.e., not obtained by sampling an analog signal). If, indeed, $x(n)$ was obtained from sampling an analog signal $x_a(t)$, then $x(n) \equiv x_a(nT)$, where T is the sampling period (i.e., the time between successive samples).

Besides the graphical representation of a discrete-time signal or sequence as illustrated in Fig. 2.1.1, there are some alternative representations that are often more convenient to use. These are:

1. Functional representation, such as

$$x(n) = \begin{cases} 1, & \text{for } n = 1, 3 \\ 4, & \text{for } n = 2 \\ 0, & \text{elsewhere} \end{cases} \tag{2.1.1}$$

2. Tabular representation, such as

n	$\cdots$	-2	-1	0	1	2	3	4	5	$\cdots$
$x(n)$	$\cdots$	0	0	0	1	4	1	0	0	$\cdots$

3. Sequence representation

An infinite-duration signal or sequence with the time origin ($n = 0$) indicated by the symbol $\uparrow$ is represented as

$$x(n) = \{\ldots 0, \underset{\uparrow}{0}, 1, 4, 1, 0, 0, \ldots\} \tag{2.1.2}$$

A sequence $x(n)$, which is zero for $n < 0$, can be represented as

$$x(n) = \{\underset{\uparrow}{0}, 1, 4, 1, 0, 0, \ldots\} \tag{2.1.3}$$

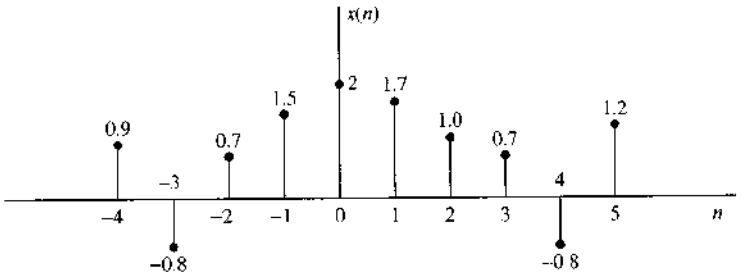


Figure 2.1.1 Graphical representation of a discrete-time signal.

The time origin for a sequence $x(n)$, which is zero for $n < 0$, is understood to be the first (leftmost) point in the sequence.

A finite-duration sequence can be represented as

$$x(n) = \{3, -1, \underset{\uparrow}{-2}, 5, 0, 4, -1\} \tag{2.1.4}$$

whereas a finite-duration sequence that satisfies the condition $x(n) = 0$ for $n < 0$ can be represented as

$$x(n) = \{\underset{\uparrow}{0}, 1, 4, 1\} \tag{2.1.5}$$

The signal in (2.1.4) consists of seven samples or points (in time), so it is called or identified as a seven-point sequence. Similarly, the sequence given by (2.1.5) is a four-point sequence.

2.1.1 Some Elementary Discrete-Time Signals

In our study of discrete-time signals and systems there are a number of basic signals that appear often and play an important role. These signals are defined below.

1. The *unit sample sequence* is denoted as $\delta(n)$ and is defined as

$$\delta(n) \equiv \begin{cases} 1, & \text{for } n = 0 \\ 0, & \text{for } n \neq 0 \end{cases} \tag{2.1.6}$$

   In words, the unit sample sequence is a signal that is zero everywhere, except at $n = 0$ where its value is unity. This signal is sometimes referred to as a *unit impulse*. In contrast to the analog signal $\delta(t)$, which is also called a unit impulse and is defined to be zero everywhere except at $t = 0$, and has unit area, the unit sample sequence is much less mathematically complicated. The graphical representation of $\delta(n)$ is shown in Fig. 2.1.2.

2. The *unit step signal* is denoted as $u(n)$ and is defined as

$$u(n) \equiv \begin{cases} 1, & \text{for } n \geq 0 \\ 0, & \text{for } n < 0 \end{cases} \tag{2.1.7}$$

   Figure 2.1.3 illustrates the unit step signal.

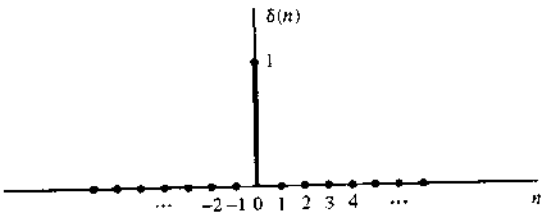


Figure 2.1.2
Graphical representation of the unit sample signal.

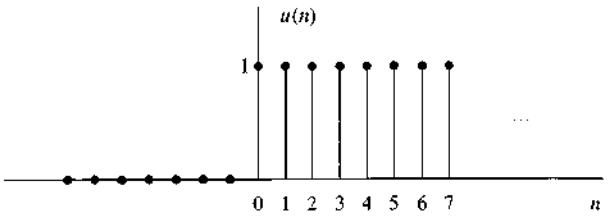


Figure 2.1.3
Graphical representation of the unit step signal.

3. The *unit ramp signal* is denoted as $u_r(n)$ and is defined as

$$u_r(n) \equiv \begin{cases} n, & \text{for } n \geq 0 \\ 0, & \text{for } n < 0 \end{cases} \tag{2.1.8}$$

This signal is illustrated in Fig. 2.1.4.

4. The *exponential signal* is a sequence of the form

$$x(n) = a^n \qquad \text{for all } n \tag{2.1.9}$$

If the parameter a is real, then $x(n)$ is a real signal. Figure 2.1.5 illustrates $x(n)$ for various values of the parameter a.

When the parameter a is complex valued, it can be expressed as

$$a \equiv re^{j\theta}$$

where r and θ are now the parameters. Hence we can express $x(n)$ as

$$\begin{aligned} x(n) &= r^n e^{j\theta n} \\ &= r^n(\cos\theta n + j \sin\theta n) \end{aligned} \tag{2.1.10}$$

Since $x(n)$ is now complex valued, it can be represented graphically by plotting the real part

$$x_R(n) \equiv r^n \cos\theta n \tag{2.1.11}$$

as a function of n, and separately plotting the imaginary part

$$x_I(n) \equiv r^n \sin\theta n \tag{2.1.12}$$

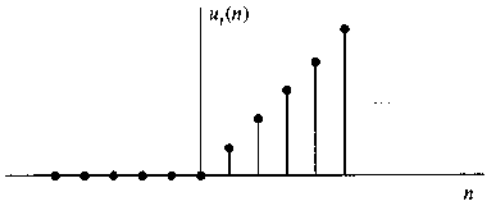


Figure 2.1.4
Graphical representation of the unit ramp signal.

as a function of n. Figure 2.1.6 illustrates the graphs of $x_R(n)$ and $x_I(n)$ for $r = 0.9$ and $\theta = \pi/10$. We observe that the signals $x_R(n)$ and $x_I(n)$ are a damped (decaying exponential) cosine function and a damped sine function. The angle variable θ is simply the frequency of the sinusoid, previously denoted by the (normalized) frequency variable ω. Clearly, if $r = 1$, the damping disappears and $x_R(n)$, $x_I(n)$, and $x(n)$ have a fixed amplitude, which is unity.

Alternatively, the signal $x(n)$ given by (2.1.10) can be represented graphically by the amplitude function

$$|x(n)| = A(n) \equiv r^n \tag{2.1.13}$$

and the phase function

$$\angle x(n) = \phi(n) \equiv \theta n \tag{2.1.14}$$

Figure 2.1.7 illustrates $A(n)$ and $\phi(n)$ for $r = 0.9$ and $\theta = \pi/10$. We observe that the phase function is linear with n. However, the phase is defined only over the interval $-\pi < \theta \leq \pi$ or, equivalently, over the interval $0 \leq \theta < 2\pi$. Consequently, by convention $\phi(n)$ is plotted over the finite interval $-\pi < \theta \leq \pi$ or $0 \leq \theta < 2\pi$. In other words, we subtract multiples of 2π from $\phi(n)$ before plotting. The subtraction of multiples of 2π from $\phi(n)$ is equivalent to interpreting the function $\phi(n)$ as $\phi(n)$, modulo 2π.

2.1.2 Classification of Discrete-Time Signals

The mathematical methods employed in the analysis of discrete-time signals and systems depend on the characteristics of the signals. In this section we classify discrete-time signals according to a number of different characteristics.

Energy signals and power signals. The energy E of a signal $x(n)$ is defined as

$$E \equiv \sum_{n=-\infty}^{\infty} |x(n)|^2 \tag{2.1.15}$$

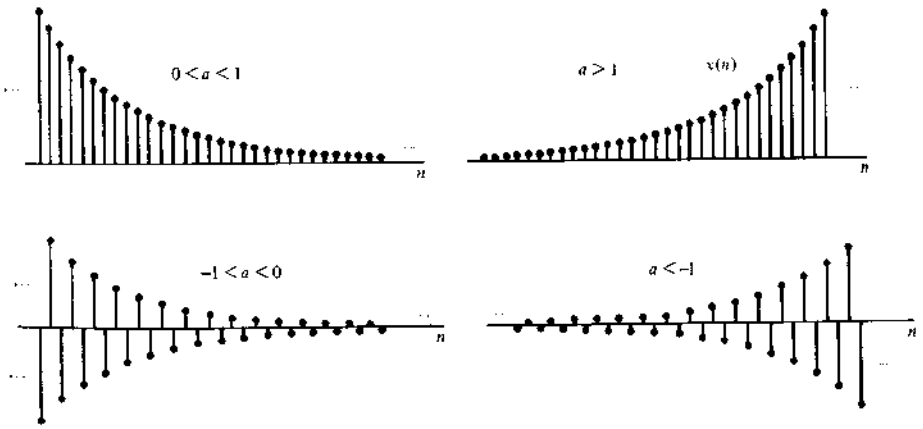


Figure 2.1.5 Graphical representation of exponential signals.

We have used the magnitude-squared values of $x(n)$, so that our definition applies to complex-valued signals as well as real-valued signals. The energy of a signal can be finite or infinite. If E is finite (i.e., $0 < E < \infty$), then $x(n)$ is called an *energy signal*. Sometimes we add a subscript x to E and write E_x to emphasize that E_x is the energy of the signal $x(n)$.

Many signals that possess infinite energy have a finite average power. The average power of a discrete-time signal $x(n)$ is defined as

$$P = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} |x(n)|^2 \tag{2.1.16}$$

If we define the signal energy of $x(n)$ over the finite interval $-N \le n \le N$ as

$$E_N \equiv \sum_{n=-N}^{N} |x(n)|^2 \tag{2.1.17}$$

then we can express the signal energy E as

$$E \equiv \lim_{N\to\infty} E_N \tag{2.1.18}$$

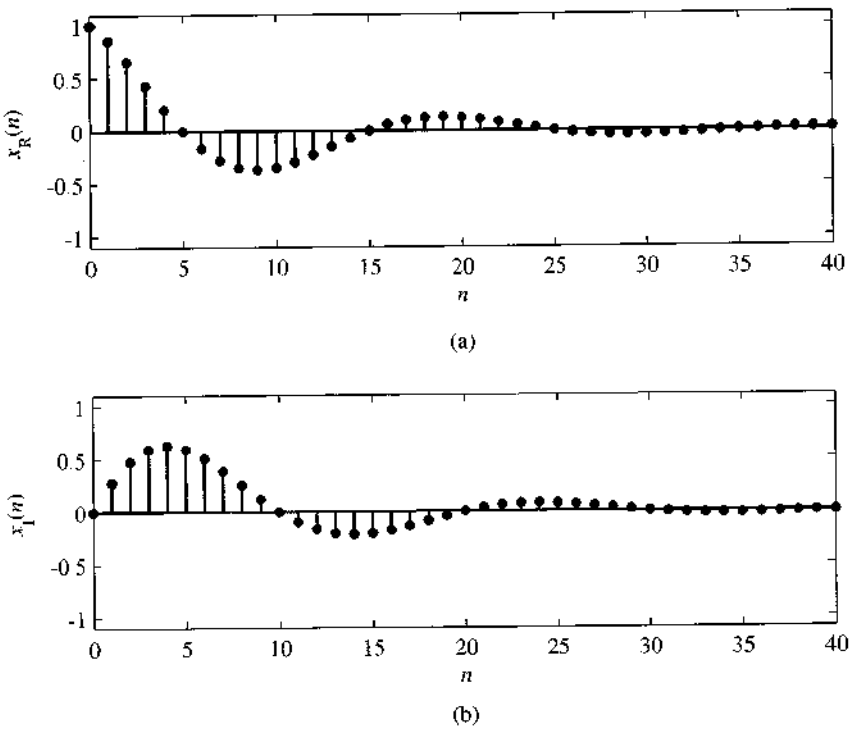


Figure 2.1.6 Graph of the real and imaginary components of a complex-valued exponential signal.

and the average power of the signal $x(n)$ as

$$P \equiv \lim_{N \to \infty} \frac{1}{2N+1} E_N \tag{2.1.19}$$

Clearly, if E is finite, $P = 0$. On the other hand, if E is infinite, the average power P may be either finite or infinite. If P is finite (and nonzero), the signal is called a *power signal.* The following example illustrates such a signal.

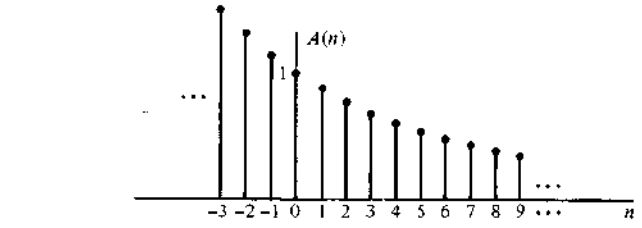
EXAMPLE 2.1.1

Determine the power and energy of the unit step sequence. The average power of the unit step signal is

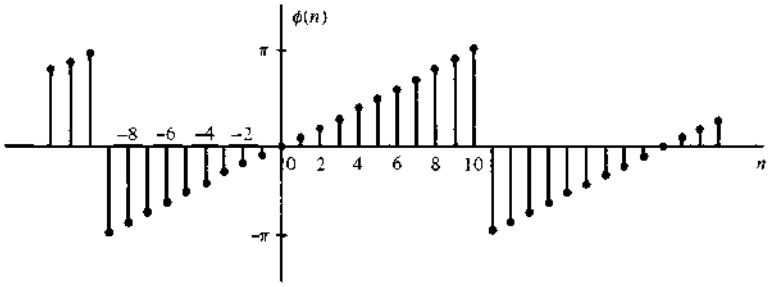
$$P = \lim_{N \to \infty} \frac{1}{2N+1} \sum_{n=0}^{N} u^2(n)$$

$$= \lim_{N \to \infty} \frac{N+1}{2N+1} = \lim_{N \to \infty} \frac{1+1/N}{2+1/N} = \frac{1}{2}$$

Consequently, the unit step sequence is a power signal. Its energy is infinite.



(a) Graph of $A(n) = r^n$, $r = 0.9$



(b) Graph of $\phi(n) = \frac{\pi}{10} n$, modulo 2π plotted in the range $(-\pi, \pi)$

Figure 2.1.7 Graph of amplitude and phase function of a complex-valued exponential signal: (a) graph of $A(n) = r^n$, $r = 0.9$; (b) graph of $\phi(n) = (\pi/10)n$, modulo 2π plotted in the range $(-\pi, \pi]$.

Similarly, it can be shown that the complex exponential sequence $x(n) = Ae^{j\omega_0 n}$ has average power A^2, so it is a power signal. On the other hand, the unit ramp sequence is neither a power signal nor an energy signal.

Periodic signals and aperiodic signals. As defined in Section 1.3, a signal $x(n)$ is periodic with period $N (N > 0)$ if and only if

$$x(n + N) = x(n) \text{ for all } n \tag{2.1.20}$$

The smallest value of N for which (2.1.20) holds is called the (fundamental) period. If there is no value of N that satisfies (2.1.20), the signal is called *nonperiodic* or *aperiodic*.

We have already observed that the sinusoidal signal of the form

$$x(n) = A \sin 2\pi f_0 n \tag{2.1.21}$$

is periodic when f_0 is a rational number, that is, if f_0 can be expressed as

$$f_0 = \frac{k}{N} \tag{2.1.22}$$

where k and N are integers.

The energy of a periodic signal $x(n)$ over a single period, say, over the interval $0 \le n \le N - 1$, is finite if $x(n)$ takes on finite values over the period. However, the energy of the periodic signal for $-\infty \le n \le \infty$ is infinite. On the other hand, the average power of the periodic signal is finite and it is equal to the average power over a single period. Thus if $x(n)$ is a periodic signal with fundamental period N and takes on finite values, its power is given by

$$P = \frac{1}{N} \sum_{n=0}^{N-1} |x(n)|^2 \tag{2.1.23}$$

Consequently, periodic signals are power signals.

Symmetric (even) and antisymmetric (odd) signals. A real-valued signal $x(n)$ is called symmetric (even) if

$$x(-n) = x(n) \tag{2.1.24}$$

On the other hand, a signal $x(n)$ is called antisymmetric (odd) if

$$x(-n) = -x(n) \tag{2.1.25}$$

We note that if $x(n)$ is odd, then $x(0) = 0$. Examples of signals with even and odd symmetry are illustrated in Fig. 2.1.8.

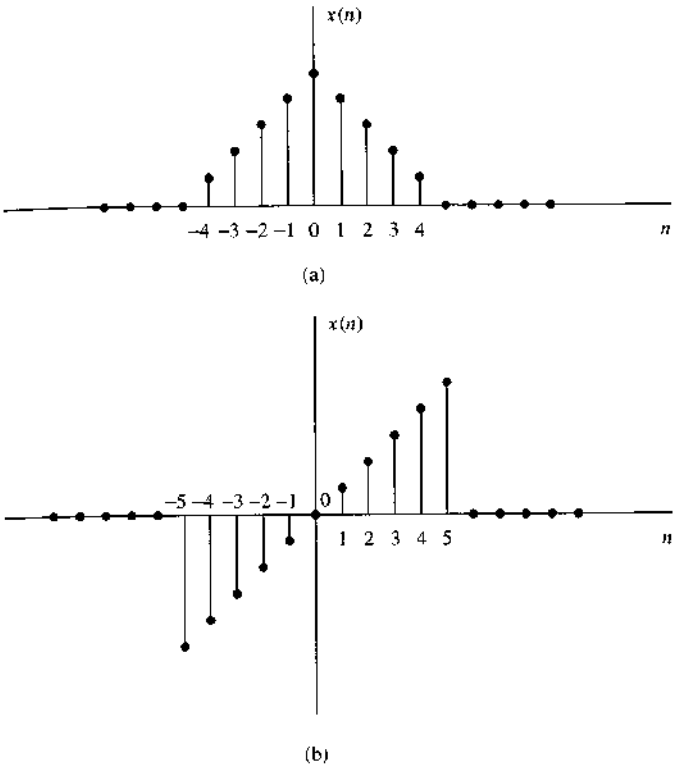


Figure 2.1.8 Example of even (a) and odd (b) signals.

We wish to illustrate that any arbitrary signal can be expressed as the sum of two signal components, one of which is even and the other odd. The even signal component is formed by adding $x(n)$ to $x(-n)$ and dividing by 2, that is,

$$x_e(n) = \frac{1}{2}[x(n) + x(-n)] \tag{2.1.26}$$

Clearly, $x_e(n)$ satisfies the symmetry condition (2.1.24). Similarly, we form an odd signal component $x_o(n)$ according to the relation

$$x_o(n) = \frac{1}{2}[x(n) - x(-n)] \tag{2.1.27}$$

Again, it is clear that $x_o(n)$ satisfies (2.1.25); hence it is indeed odd. Now, if we add the two signal components, defined by (2.1.26) and (2.1.27), we obtain $x(n)$, that is,

$$x(n) = x_e(n) + x_o(n) \tag{2.1.28}$$

Thus any arbitrary signal can be expressed as in (2.1.28).

2.1.3 Simple Manipulations of Discrete-Time Signals

In this section we consider some simple modifications or manipulations involving the independent variable and the signal amplitude (dependent variable).

Transformation of the independent variable (time). A signal $x(n)$ may be shifted in time by replacing the independent variable n by $n - k$, where k is an integer. If k is a positive integer, the time shift results in a delay of the signal by k units of time. If k is a negative integer, the time shift results in an advance of the signal by $|k|$ units in time.

EXAMPLE 2.1.2

A signal $x(n)$ is graphically illustrated in Fig. 2.1.9(a). Show a graphical representation of the signals $x(n-3)$ and $x(n+2)$.

Solution. The signal $x(n-3)$ is obtained by delaying $x(n)$ by three units in time. The result is illustrated in Fig. 2.1.9(b). On the other hand, the signal $x(n+2)$ is obtained by advancing $x(n)$ by two units in time. The result is illustrated in Fig. 2.1.9(c). Note that delay corresponds to shifting a signal to the right, whereas advance implies shifting the signal to the left on the time axis.

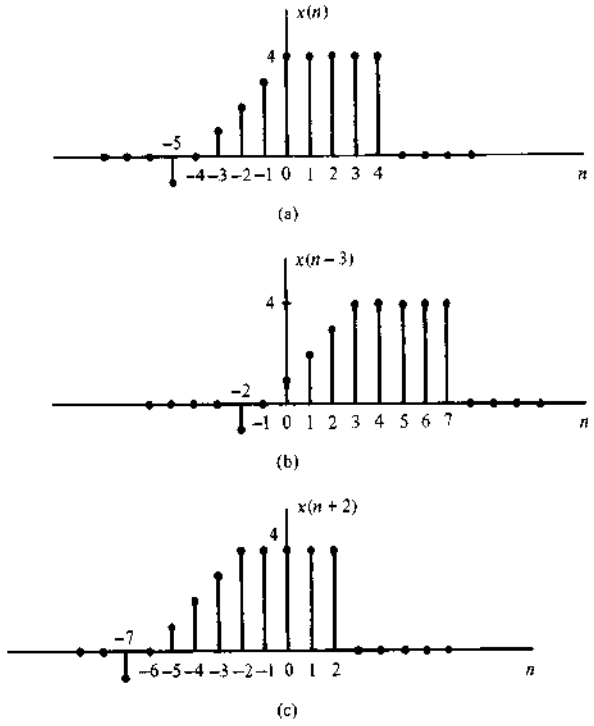


Figure 2.1.9
Graphical representation of a signal, and its delayed and advanced versions.

If the signal $x(n)$ is stored on magnetic tape or on a disk or, perhaps, in the memory of a computer, it is a relatively simple operation to modify the base by introducing a delay or an advance. On the other hand, if the signal is not stored but is being generated by some physical phenomenon in real time, it is not possible to advance the signal in time, since such an operation involves signal samples that have not yet been generated. Whereas it is always possible to insert a delay into signal samples that have already been generated, it is physically impossible to view the future signal samples. Consequently, in real-time signal processing applications, the operation of advancing the time base of the signal is physically unrealizable.

Another useful modification of the time base is to replace the independent variable n by $-n$. The result of this operation is a *folding* or a *reflection* of the signal about the time origin $n = 0$.

EXAMPLE 2.1.3

Show the graphical representation of the signals $x(-n)$ and $x(-n+2)$, where $x(n)$ is the signal illustrated in Fig. 2.1.10(a).

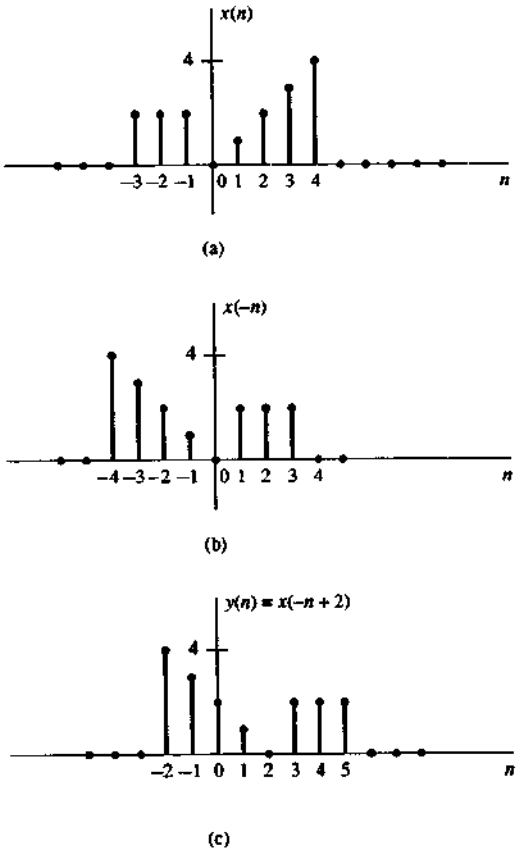


Figure 2.1.10 Graphical illustration of the folding and shifting operations

Solution. The new signal $y(n) = x(-n)$ is shown in Fig. 2.1.10(b). Note that $y(0) = x(0)$, $y(1) = x(-1)$, $y(2) = x(-2)$, and so on. Also, $y(-1) = x(1)$, $y(-2) = x(2)$, and so on. Therefore, $y(n)$ is simply $x(n)$ reflected or folded about the time origin $n = 0$. The signal $y(n) = x(-n + 2)$ is simply $x(-n)$ delayed by two units in time. The resulting signal is illustrated in Fig. 2.1.10(c). A simple way to verify that the result in Fig. 2.1.10(c) is correct is to compute samples, such as $y(0) = x(2)$, $y(1) = x(1)$, $y(2) = x(0)$, $y(-1) = x(3)$, and so on.

It is important to note that the operations of folding and time delaying (or advancing) a signal are not commutative. If we denote the time-delay operation by TD and the folding operation by FD, we can write

$$\begin{aligned} \mathrm{TD}_k[x(n)] &= x(n-k), \qquad k > 0 \\ \mathrm{FD}[x(n)] &= x(-n) \end{aligned} \tag{2.1.29}$$

Now

$$\mathrm{TD}_k\{\mathrm{FD}[x(n)]\} = \mathrm{TD}_k[x(-n)] = x(-n+k) \tag{2.1.30}$$

whereas

$$\mathrm{FD}\{\mathrm{TD}_k[x(n)]\} = \mathrm{FD}[x(n-k)] = x(-n-k) \tag{2.1.31}$$

Note that because the signs of n and k in $x(n-k)$ and $x(-n+k)$ are different, the result is a shift of the signals $x(n)$ and $x(-n)$ to the right by k samples, corresponding to a time delay.

A third modification of the independent variable involves replacing n by μn, where μ is an integer. We refer to this time-base modification as *time scaling* or *down-sampling*.

EXAMPLE 2.1.4

Show the graphical representation of the signal $y(n) = x(2n)$, where $x(n)$ is the signal illustrated in Fig. 2.1.11(a).

Solution. We note that the signal $y(n)$ is obtained from $x(n)$ by taking every other sample from $x(n)$, starting with $x(0)$. Thus $y(0) = x(0)$, $y(1) = x(2)$, $y(2) = x(4)$, ... and $y(-1) = x(-2)$, $y(-2) = x(-4)$, and so on. In other words, we have skipped the odd-numbered samples in $x(n)$ and retained the even-numbered samples. The resulting signal is illustrated in Fig. 2.1.11(b).

If the signal $x(n)$ was originally obtained by sampling an analog signal $x_a(t)$, then $x(n) = x_a(nT)$, where T is the sampling interval. Now, $y(n) = x(2n) = x_a(2Tn)$. Hence the time-scaling operation described in Example 2.1.4 is equivalent to changing the sampling rate from $1/T$ to $1/2T$, that is, to decreasing the rate by a factor of 2. This is a *down-sampling* operation.

Addition, multiplication, and scaling of sequences. Amplitude modifications include *addition, multiplication,* and *scaling* of discrete-time signals.

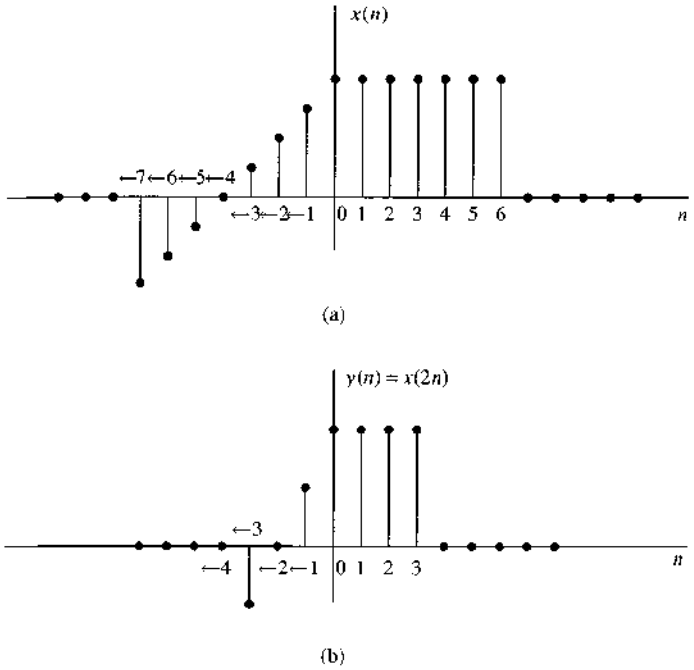


Figure 2.1.11 Graphical illustration of down-sampling operation.

Amplitude scaling of a signal by a constant A is accomplished by multiplying the value of every signal sample by A. Consequently, we obtain

$$y(n) = Ax(n), \qquad -\infty < n < \infty$$

The *sum* of two signals $x_1(n)$ and $x_2(n)$ is a signal $y(n)$, whose value at any instant is equal to the sum of the values of these two signals at that instant, that is,

$$y(n) = x_1(n) + x_2(n), \qquad -\infty < n < \infty$$

The *product* of two signals is similarly defined on a sample-to-sample basis as

$$y(n) = x_1(n)x_2(n), \qquad -\infty < n < \infty$$

2.2 Discrete-Time Systems

In many applications of digital signal processing we wish to design a device or an algorithm that performs some prescribed operation on a discrete-time signal. Such a device or algorithm is called a discrete-time system. More specifically, a *discrete-time system* is a device or algorithm that operates on a discrete-time signal, called the *input* or *excitation*, according to some well-defined rule, to produce another discrete-time

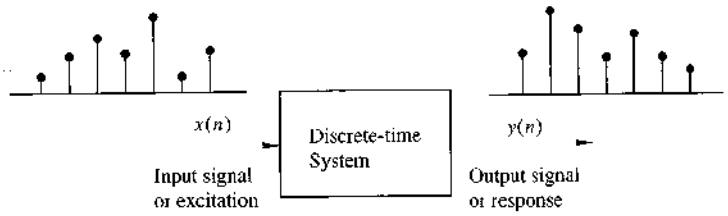


Figure 2.2.1 Block diagram representation of a discrete-time system.

signal called the *output* or *response* of the system. In general, we view a system as an operation or a set of operations performed on the input signal $x(n)$ to produce the output signal $y(n)$. We say that the input signal $x(n)$ is *transformed* by the system into a signal $y(n)$, and express the general relationship between $x(n)$ and $y(n)$ as

$$y(n) \equiv \mathcal{T}[x(n)] \tag{2.2.1}$$

where the symbol $\mathcal{T}$ denotes the transformation (also called an operator) or processing performed by the system on $x(n)$ to produce $y(n)$. The mathematical relationship in (2.2.1) is depicted graphically in Fig. 2.2.1.

There are various ways to describe the characteristics of the system and the operation it performs on $x(n)$ to produce $y(n)$. In this chapter we shall be concerned with the time-domain characterization of systems. We shall begin with an input–output description of the system. The input–output description focuses on the behavior at the terminals of the system and ignores the detailed internal construction or realization of the system. Later, in Chapter 9, we consider the implementation of discrete-time systems and describe the different structures for their realization.

2.2.1 Input–Output Description of Systems

The input–output description of a discrete-time system consists of a mathematical expression or a rule, which explicitly defines the relation between the input and output signals (*input–output relationship*). The exact internal structure of the system is either unknown or ignored. Thus the only way to interact with the system is by using its input and output terminals (i.e., the system is assumed to be a "black box" to the user). To reflect this philosophy, we use the graphical representation depicted in Fig. 2.2.1, and the general input–output relationship in (2.2.1) or, alternatively, the notation

$$x(n) \xrightarrow{\mathcal{T}} y(n) \tag{2.2.2}$$

which simply means that $y(n)$ is the response of the system $\mathcal{T}$ to the excitation $x(n)$. The following examples illustrate several different systems.

EXAMPLE 2.2.1

Determine the response of the following sytems to the input signal

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

(a) $y(n) = x(n)$ (identity system)

(b) $y(n) = x(n-1)$ (unit delay system)

(c) $y(n) = x(n+1)$ (unit advance system)

(d) $y(n) = \frac{1}{3}[x(n+1) + x(n) + x(n-1)]$ (moving average filter)

(e) $y(n) = \text{median}\{x(n+1), x(n), x(n-1)\}$ (median filter)

(f) $y(n) = \sum_{k=-\infty}^{n} x(k) = x(n) + x(n-1) + x(n-2) + \cdots$ (accumulator) (2.2.3)

Solution. First, we determine explicitly the sample values of the input signal

$$x(n) = \{\ldots, 0, 3, 2, \underset{\uparrow}{1}, 0, 1, 2, 3, 0, \ldots\}$$

Next, we determine the output of each system using its input–output relationship.

(a) In this case the output is exactly the same as the input signal. Such a system is known as the *identity* system.

(b) This system simply delays the input by one sample. Thus its output is given by

$$x(n) = \{\ldots, 0, 3, 2, \underset{\uparrow}{1}, 0, 1, 2, 3, 0, \ldots\}$$

(c) In this case the system "advances" the input one sample into the future. For example, the value of the output at time $n = 0$ is $y(0) = x(1)$. The response of this system to the given input is

$$x(n) = \{\ldots, 0, 3, 2, 1, \underset{\uparrow}{0}, 1, 2, 3, 0, \ldots\}$$

(d) The output of this system at any time is the mean value of the present, the immediate past, and the immediate future samples. For example, the output at time $n = 0$ is

$$y(0) = \frac{1}{3}[x(-1) + x(0) + x(1)] = \frac{1}{3}[1 + 0 + 1] = \frac{2}{3}$$

Repeating this computation for every value of n, we obtain the output signal

$$y(n) = \{\ldots, 0, 1, \frac{5}{3}, 2, 1, \underset{\uparrow}{\frac{2}{3}}, 1, 2, \frac{5}{3}, 1, 0, \ldots\}$$

(e) This system selects as its output at time n the median value of the three input samples $x(n-1)$, $x(n)$, and $x(n+1)$. Thus the response of this system to the input signal $x(n)$ is

$$y(n) = \{0, 2, 2, 1, 1, \underset{\uparrow}{1}, 2, 2, 0, 0, 0, \ldots\}$$

(f) This system is basically an *accumulator* that computes the running sum of all the past input values up to present time. The response of this system to the given input is

$$y(n) = \{\ldots, 0, 3, 5, 6, \underset{\uparrow}{6}, 7, 9, 12, 0, \ldots\}$$

We observe that for several of the systems considered in Example 2.2.1 the output at time $n = n_0$ depends not only on the value of the input at $n = n_0$ [i.e., $x(n_0)$], but also on the values of the input applied to the system before and after $n = n_0$. Consider, for instance, the accumulator in the example. We see that the output at time $n = n_0$ depends not only on the input at time $n = n_0$, but also on $x(n)$ at times $n = n_0 - 1, n_0 - 2$, and so on. By a simple algebraic manipulation the input–output relation of the accumulator can be written as

$$\begin{aligned} y(n) &= \sum_{k=-\infty}^{n} x(k) = \sum_{k=-\infty}^{n-1} x(k) + x(n) \\ &= y(n-1) + x(n) \end{aligned} \tag{2.2.4}$$

which justifies the term *accumulator*. Indeed, the system computes the current value of the output by adding (accumulating) the current value of the input to the previous output value.

There are some interesting conclusions that can be drawn by taking a close look into this apparently simple system. Suppose that we are given the input signal $x(n)$ for $n \geq n_0$, and we wish to determine the output $y(n)$ of this system for $n \geq n_0$. For $n = n_0, n_0 + 1, \ldots$, (2.2.4) gives

$$y(n_0) = y(n_0 - 1) + x(n_0)$$

$$y(n_0 + 1) = y(n_0) + x(n_0 + 1)$$

and so on. Note that we have a problem in computing $y(n_0)$, since it depends on $y(n_0 - 1)$. However,

$$y(n_0 - 1) = \sum_{k=-\infty}^{n_0-1} x(k)$$

that is, $y(n_0 - 1)$ "summarizes" the effect on the system from all the inputs which had been applied to the system before time n_0. Thus the response of the system for $n \geq n_0$ to the input $x(n)$ that is applied at time n_0 is the combined result of this input and all inputs that had been applied previously to the system. Consequently, $y(n)$, $n \geq n_0$ is not uniquely determined by the input $x(n)$ for $n \geq n_0$.

The additional information required to determine $y(n)$ for $n \geq n_0$ is the *initial condition* $y(n_0 - 1)$. This value summarizes the effect of all previous inputs to the system. Thus the initial condition $y(n_0 - 1)$ together with the input sequence $x(n)$ for $n \geq n_0$ uniquely determine the output sequence $y(n)$ for $n \geq n_0$.

If the accumulator had no excitation prior to n_0, the initial condition is $y(n_0 - 1) = 0$. In such a case we say that the system is *initially relaxed*. Since $y(n_0 - 1) = 0$, the output sequence $y(n)$ depends only on the input sequence $x(n)$ for $n \geq n_0$.

It is customary to assume that every system is relaxed at $n = -\infty$. In this case, if an input $x(n)$ is applied at $n = -\infty$, the corresponding output $y(n)$ is *solely* and *uniquely* determined by the given input.

EXAMPLE 2.2.2

The accumulator described by (2.2.30) is excited by the sequence $x(n) = nu(n)$. Determine its output under the condition that:

(a) It is initially relaxed [i.e., $y(-1) = 0$].

(b) Initially, $y(-1) = 1$.

Solution. The output of the system is defined as

$$y(n) = \sum_{k=-\infty}^{n} x(k) = \sum_{k=-\infty}^{-1} x(k) + \sum_{k=0}^{n} x(k)$$

$$= y(-1) + \sum_{k=0}^{n} x(k)$$

$$= y(-1) + \frac{n(n+1)}{2}$$

(a) If the system is initially relaxed, $y(-1) = 0$ and hence

$$y(n) = \frac{n(n+1)}{2}, \qquad n \geq 0$$

(b) On the other hand, if the initial condition is $y(-1) = 1$, then

$$y(n) = 1 + \frac{n(n+1)}{2} = \frac{n^2 + n + 2}{2}, \qquad n \geq 0$$

2.2.2 Block Diagram Representation of Discrete-Time Systems

It is useful at this point to introduce a block diagram representation of discrete-time systems. For this purpose we need to define some basic building blocks that can be interconnected to form complex systems.

An adder. Figure 2.2.2 illustrates a system (adder) that performs the addition of two signal sequences to form another (the sum) sequence, which we denote as $y(n)$. Note that it is not necessary to store either one of the sequences in order to perform the addition. In other words, the addition operation is *memoryless*.

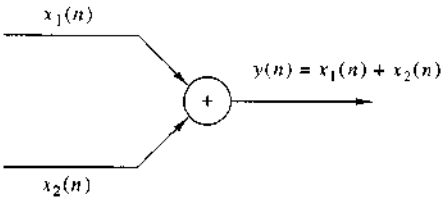


Figure 2.2.2
Graphical representation of an adder

A constant multiplier. This operation is depicted by Fig. 2.2.3, and simply represents applying a scale factor on the input $x(n)$. Note that this operation is also memoryless.

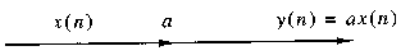


Figure 2.2.3
Graphical representation of a constant multiplier.

A signal multiplier. Figure 2.2.4 illustrates the multiplication of two signal sequences to form another (the product) sequence, denoted in the figure as $y(n)$. As in the preceding two cases, we can view the multiplication operation as memoryless.

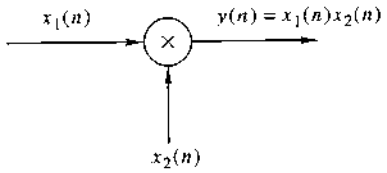


Figure 2.2.4
Graphical representation of a signal multiplier.

A unit delay element. The unit delay is a special system that simply delays the signal passing through it by one sample. Figure 2.2.5 illustrates such a system. If the input signal is $x(n)$, the output is $x(n-1)$. In fact, the sample $x(n-1)$ is stored in memory at time $n-1$ and it is recalled from memory at time n to form

$$y(n) = x(n-1)$$

Thus this basic building block requires memory. The use of the symbol z^{-1} to denote the unit of delay will become apparent when we discuss the z-transform in Chapter 3.

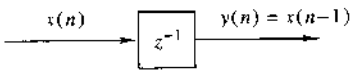


Figure 2.2.5
Graphical representation of the unit delay element.

A unit advance element. In contrast to the unit delay, a unit advance moves the input $x(n)$ ahead by one sample in time to yield $x(n+1)$. Figure 2.2.6 illustrates this operation, with the operator z being used to denote the unit advance. We observe that any such advance is physically impossible in real time, since, in fact, it involves looking into the future of the signal. On the other hand, if we store the signal in the memory of the computer, we can recall any sample at any time. In such a non-real-time application, it is possible to advance the signal $x(n)$ in time.

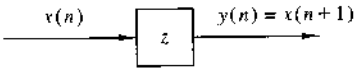


Figure 2.2.6
Graphical representation of the unit advance element.

EXAMPLE 2.2.3

Using basic building blocks introduced above, sketch the block diagram representation of the discrete-time system described by the input–output relation

$$y(n) = \frac{1}{4}y(n-1) + \frac{1}{2}x(n) + \frac{1}{2}x(n-1) \tag{2.2.5}$$

where $x(n)$ is the input and $y(n)$ is the output of the system.

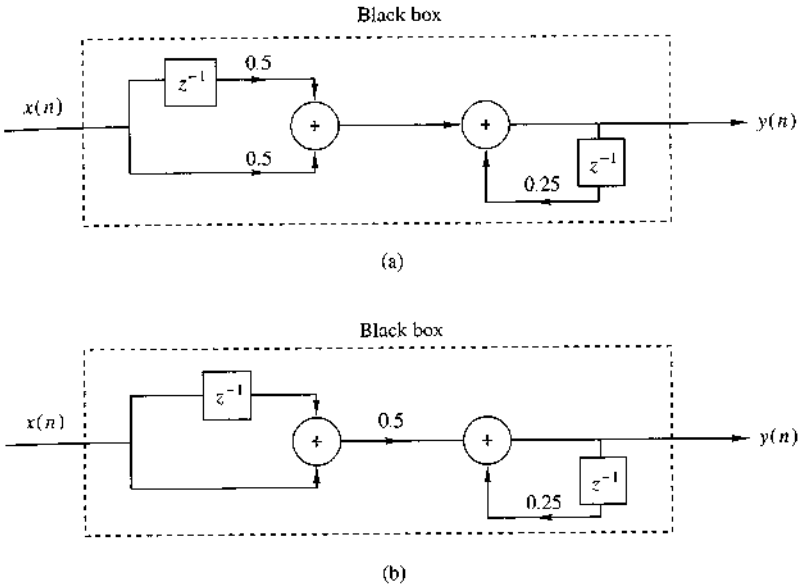


Figure 2.2.7 Block diagram realizations of the system $y(n) = 0.25y(n-1) + 0.5x(n) + 0.5x(n-1)$.

Solution. According to (2.2.5), the output $y(n)$ is obtained by multiplying the input $x(n)$ by 0.5, multiplying the previous input $x(n-1)$ by 0.5, adding the two products, and then adding the previous output $y(n-1)$ multiplied by $\frac{1}{4}$. Figure 2.2.7(a) illustrates this block diagram realization of the system. A simple rearrangement of (2.2.5), namely,

$$y(n) = \frac{1}{4}y(n-1) + \frac{1}{2}[x(n) + x(n-1)] \tag{2.2.6}$$

leads to the block diagram realization shown in Fig. 2.2.7(b). Note that if we treat "the system" from the "viewpoint" of an input–output or an external description, we are not concerned about how the system is realized. On the other hand, if we adopt an internal description of the system, we know exactly how the system building blocks are configured. In terms of such a realization, we can see that a system is *relaxed* at time $n = n_0$ if the outputs of all the *delays* existing in the system are zero at $n = n_0$ (i.e., all memory is *filled* with zeros).

2.2.3 Classification of Discrete-Time Systems

In the analysis as well as in the design of systems, it is desirable to classify the systems according to the general properties that they satisfy. In fact, the mathematical techniques that we develop in this and in subsequent chapters for analyzing and designing discrete-time systems depend heavily on the general characteristics of the systems that are being considered. For this reason it is necessary for us to develop a number of properties or categories that can be used to describe the general characteristics of systems.

We stress the point that for a system to possess a given property, the property must hold for every possible input signal to the system. If a property holds for some

input signals but not for others, the system does not possess that property. Thus a counterexample is sufficient to prove that a system does not possess a property. However, to prove that the system has some property, we must prove that this property holds for every possible input signal.

Static versus dynamic systems. A discrete-time system is called *static* or memoryless if its output at any instant n depends at most on the input sample at the same time, but not on past or future samples of the input. In any other case, the system is said to be *dynamic* or to have memory. If the output of a system at time n is completely determined by the input samples in the interval from $n - N$ to $n (N \geq 0)$, the system is said to have *memory* of duration N. If $N = 0$, the system is static. If $0 < N < \infty$, the system is said to have *finite memory*, whereas if $N = \infty$, the system is said to have *infinite memory*.

The systems described by the following input–output equations

$$y(n) = ax(n) \tag{2.2.7}$$

$$y(n) = nx(n) + bx^3(n) \tag{2.2.8}$$

are both static or memoryless. Note that there is no need to store any of the past inputs or outputs in order to compute the present output. On the other hand, the systems described by the following input–output relations

$$y(n) = x(n) + 3x(n-1) \tag{2.2.9}$$

$$y(n) = \sum_{k=0}^{n} x(n-k) \tag{2.2.10}$$

$$y(n) = \sum_{k=0}^{\infty} x(n-k) \tag{2.2.11}$$

are dynamic systems or systems with memory. The systems described by (2.2.9) and (2.2.10) have finite memory, whereas the system described by (2.2.11) has infinite memory.

We observe that static or memoryless systems are described in general by input-output equations of the form

$$y(n) = \mathcal{T}[x(n), n] \tag{2.2.12}$$

and they do not include delay elements (memory).

Time-invariant versus time-variant systems. We can subdivide the general class of systems into the two broad categories, time-invariant systems and time-variant systems. A system is called time-invariant if its input–output characteristics do not change with time. To elaborate, suppose that we have a system $\mathcal{T}$ in a relaxed state

which, when excited by an input signal $x(n)$, produces an output signal $y(n)$. Thus we write

$$y(n) = \mathcal{T}[x(n)] \tag{2.2.13}$$

Now suppose that the same input signal is delayed by k units of time to yield $x(n-k)$, and again applied to the same system. If the characteristics of the system do not change with time, the output of the relaxed system will be $y(n-k)$. That is, the output will be the same as the response to $x(n)$, except that it will be delayed by the same k units in time that the input was delayed. This leads us to define a time-invariant or shift-invariant system as follows.

Definition. A relaxed system $\mathcal{T}$ is *time invariant* or *shift invariant* if and only if

$$x(n) \xrightarrow{\mathcal{T}} y(n)$$

implies that

$$x(n-k) \xrightarrow{\mathcal{T}} y(n-k) \tag{2.2.14}$$

for every input signal $x(n)$ and every time shift k.

To determine if any given system is time invariant, we need to perform the test specified by the preceding definition. Basically, we excite the system with an arbitrary input sequence $x(n)$, which produces an output denoted as $y(n)$. Next we delay the input sequence by some amount k and recompute the output. In general, we can write the output as

$$y(n, k) = \mathcal{T}[x(n-k)]$$

Now if this output $y(n, k) = y(n-k)$, for all possible values of k, the system is time invariant. On the other hand, if the output $y(n, k) \neq y(n-k)$, even for one value of k, the system is time variant.

EXAMPLE 2.2.4

Determine if the systems shown in Fig. 2.2.8 are time invariant or time variant.

Solution.

(a) This system is described by the input–output equations

$$y(n) = \mathcal{T}[x(n)] = x(n) - x(n-1) \tag{2.2.15}$$

Now if the input is delayed by k units in time and applied to the system, it is clear from the block diagram that the output will be

$$y(n, k) = x(n-k) - x(n-k-1) \tag{2.2.16}$$

On the other hand, from (2.2.14) we note that if we delay $y(n)$ by k units in time, we obtain

$$y(n-k) = x(n-k) - x(n-k-1) \tag{2.2.17}$$

Since the right-hand sides of (2.2.16) and (2.2.17) are identical, it follows that $y(n, k) = y(n-k)$. Therefore, the system is time invariant.

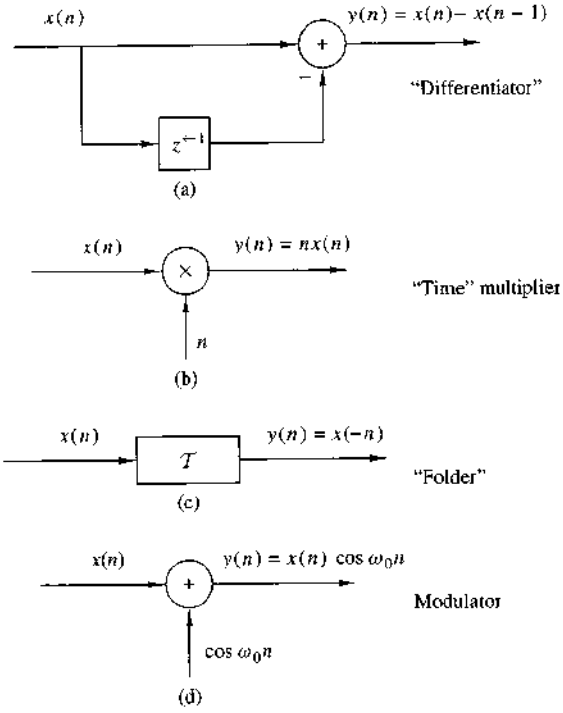


Figure 2.2.8
Examples of a time-invariant (a) and some time-variant systems (b)–(d).

(b) The input–output equation for this system is

$$y(n) = \mathcal{T}[x(n)] = nx(n) \tag{2.2.18}$$

The response of this system to $x(n-k)$ is

$$y(n,k) = nx(n-k) \tag{2.2.19}$$

Now if we delay $y(n)$ in (2.2.18) by k units in time, we obtain

$$\begin{aligned} y(n-k) &= (n-k)x(n-k) \\ &= nx(n-k) - kx(n-k) \end{aligned} \tag{2.2.20}$$

This system is time variant, since $y(n,k) \neq y(n-k)$.

(c) This system is described by the input–output relation

$$y(n) = \mathcal{T}[x(n)] = x(-n) \tag{2.2.21}$$

The response of this system to $x(n-k)$ is

$$y(n,k) = \mathcal{T}[x(n-k)] = x(-n-k) \tag{2.2.22}$$

Now, if we delay the output $y(n)$, as given by (2.2.21), by k units in time, the result will be

$$y(n-k) = x(-n+k) \tag{2.2.23}$$

Since $y(n,k) \neq y(n-k)$, the system is time variant.

(d) The input–output equation for this system is

$$y(n) = x(n)\cos\omega_0 n \tag{2.2.24}$$

The response of this system to $x(n-k)$ is

$$y(n,k) = x(n-k)\cos\omega_0 n \tag{2.2.25}$$

If the expression in (2.2.24) is delayed by k units and the result is compared to (2.2.25), it is evident that the system is time variant.

Linear versus nonlinear systems. The general class of systems can also be subdivided into linear systems and nonlinear systems. A linear system is one that satisfies the *superposition principle*. Simply stated, the principle of superposition requires that the response of the system to a weighted sum of signals be equal to the corresponding weighted sum of the responses (outputs) of the system to each of the individual input signals. Hence we have the following definition of linearity.

Definition. A system is linear if and only if

$$\mathcal{T}[a_1x_1(n) + a_2x_2(n)] = a_1\mathcal{T}[x_1(n)] + a_2\mathcal{T}[x_2(n)] \tag{2.2.26}$$

for any arbitrary input sequences $x_1(n)$ and $x_2(n)$, and any arbitrary constants a_1 and a_2. Figure 2.2.9 gives a pictorial illustration of the superposition principle.

The superposition principle embodied in the relation (2.2.26) can be separated into two parts. First, suppose that $a_2 = 0$. Then (2.2.26) reduces to

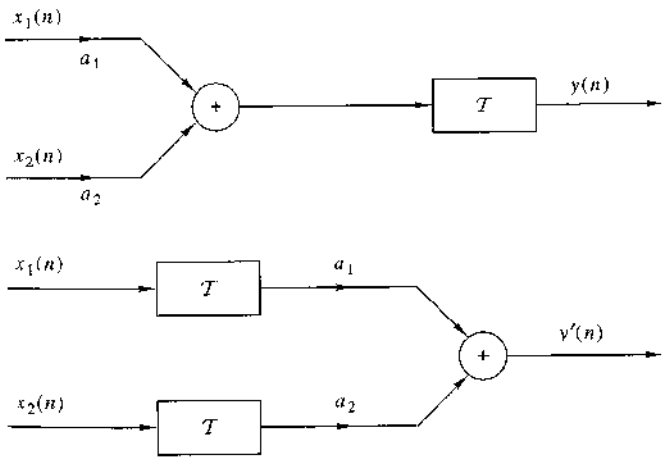


Figure 2.2.9 Graphical representation of the superposition principle. $\mathcal{T}$ is linear if and only if $y(n) = y'(n)$.

$$\mathcal{T}[a_1 x_1(n)] = a_1 \mathcal{T}[x_1(n)] = a_1 y_1(n) \tag{2.2.27}$$

where

$$y_1(n) = \mathcal{T}[x_1(n)]$$

The relation (2.2.27) demonstrates the *multiplicative* or *scaling property* of a linear system. That is, if the response of the system to the input $x_1(n)$ is $y_1(n)$, the response to $a_1 x_1(n)$ is simply $a_1 y_1(n)$. Thus any scaling of the input results in an identical scaling of the corresponding output.

Second, suppose that $a_1 = a_2 = 1$ in (2.2.26). Then

$$\begin{aligned} \mathcal{T}[x_1(n) + x_2(n)] &= \mathcal{T}[x_1(n)] + \mathcal{T}[x_1(n)] \\ &= y_1(n) + y_2(n) \end{aligned} \tag{2.2.28}$$

This relation demonstrates the *additivity property* of a linear system. The additivity and multiplicative properties constitute the superposition principle as it applies to linear systems.

The linearity condition embodied in (2.2.26) can be extended arbitrarily to any weighted linear combination of signals by induction. In general, we have

$$x(n) = \sum_{k=1}^{M-1} a_k x_k(n) \xrightarrow{\mathcal{T}} y(n) = \sum_{k=1}^{M-1} a_k y_k(n) \tag{2.2.29}$$

where

$$y_k(n) = \mathcal{T}[x_k(n)], \qquad k = 1, 2, \ldots, M-1 \tag{2.2.30}$$

We observe from (2.2.27) that if $a_1 = 0$, then $y(n) = 0$. In other words, a relaxed, linear system with zero input produces a zero output. If a system produces a nonzero output with a zero input, the system may be either nonrelaxed or nonlinear. If a relaxed system does not satisfy the superposition principle as given by the definition above, it is called *nonlinear*.

EXAMPLE 2.2.5

Determine if the systems described by the following input–output equations are linear or nonlinear.

(a) $y(n) = nx(n)$ **(b)** $y(n) = x(n^2)$ **(c)** $y(n) = x^2(n)$
(d) $y(n) = Ax(n) + B$ **(e)** $y(n) = e^{x(n)}$

Solution.

(a) For two input sequences $x_1(n)$ and $x_2(n)$, the corresponding outputs are

$$\begin{aligned} y_1(n) &= nx_1(n) \\ y_2(n) &= nx_2(n) \end{aligned} \tag{2.2.31}$$

A linear combination of the two input sequences results in the output

$$\begin{aligned} y_3(n) &= \mathcal{T}[a_1 x_1(n) + a_2 x_2(n)] = n[a_1 x_1(n) + a_2 x_2(n)] \\ &= a_1 n x_1(n) + a_2 n x_2(n) \end{aligned} \tag{2.2.32}$$

On the other hand, a linear combination of the two outputs in (2.2.31) results in the output

$$a_1 y_1(n) + a_2 y_2(n) = a_1 n x_1(n) + a_2 n x_2(n) \tag{2.2.33}$$

Since the right-hand sides of (2.2.32) and (2.2.33) are identical, the system is linear.

(b) As in part (a), we find the response of the system to two separate input signals $x_1(n)$ and $x_2(n)$. The result is

$$\begin{aligned} y_1(n) &= x_1(n^2) \\ y_2(n) &= x_2(n^2) \end{aligned} \tag{2.2.34}$$

The output of the system to a linear combination of $x_1(n)$ and $x_2(n)$ is

$$y_3(n) = \mathcal{T}[a_1 x_1(n) + a_2 x_2(n)] = a_1 x_1(n^2) + a_2 x_2(n^2) \tag{2.2.35}$$

Finally, a linear combination of the two outputs in (2.2.34) yields

$$a_1 y_1(n) + a_2 y_2(n) = a_1 x_1(n^2) + a_2 x_2(n^2) \tag{2.2.36}$$

By comparing (2.2.35) with (2.2.36), we conclude that the system is linear.

(c) The output of the system is the square of the input. (Electronic devices that have such an input–output characteristic are called square-law devices.) From our previous discussion it is clear that such a system is memoryless. We now illustrate that this system is nonlinear.

The responses of the system to two separate input signals are

$$\begin{aligned} y_1(n) &= x_1^2(n) \\ y_2(n) &= x_2^2(n) \end{aligned} \tag{2.2.37}$$

The response of the system to a linear combination of these two input signals is

$$\begin{aligned} y_3(n) &= \mathcal{T}[a_1 x_1(n) + a_2 x_2(n)] \\ &= [a_1 x_1(n) + a_2 x_2(n)]^2 \\ &= a_1^2 x_1^2(n) + 2a_1 a_2 x_1(n) x_2(n) + a_2^2 x_2^2(n) \end{aligned} \tag{2.2.38}$$

On the other hand, if the system is linear, it will produce a linear combination of the two outputs in (2.2.37), namely,

$$a_1 y_1(n) + a_2 y_2(n) = a_1 x_1^2(n) + a_2 x_2^2(n) \tag{2.2.39}$$

Since the actual output of the system, as given by (2.2.38), is not equal to (2.2.39), the system is nonlinear.

(d) Assuming that the system is excited by $x_1(n)$ and $x_2(n)$ separately, we obtain the corresponding outputs

$$\begin{aligned} y_1(n) &= Ax_1(n) + B \\ y_2(n) &= Ax_2(n) + B \end{aligned} \tag{2.2.40}$$

A linear combination of $x_1(n)$ and $x_2(n)$ produces the output

$$\begin{aligned} y_3(n) &= \mathcal{T}[a_1x_1(n) + a_2x_2(n)] \\ &= A[a_1x_1(n) + a_2x_2(n)] + B \\ &= Aa_1x_1(n) + a_2Ax_2(n) + B \end{aligned} \tag{2.2.41}$$

On the other hand, if the system were linear, its output to the linear combination of $x_1(n)$ and $x_2(n)$ would be a linear combination of $y_1(n)$ and $y_2(n)$, that is,

$$a_1y_1(n) + a_2y_2(n) = a_1Ax_1(n) + a_1B + a_2Ax_2(n) + a_2B \tag{2.2.42}$$

Clearly, (2.2.41) and (2.2.42) are different and hence the system fails to satisfy the linearity test.

The reason that this system fails to satisfy the linearity test is not that the system is nonlinear (in fact, the system is described by a linear equation) but the presence of the constant B. Consequently, the output depends on both the input excitation and on the parameter $B \neq 0$. Hence, for $B \neq 0$, the system is not relaxed. If we set $B = 0$, the system is now relaxed and the linearity test is satisfied.

(e) Note that the system described by the input–output equation

$$y(n) = e^{x(n)} \tag{2.2.43}$$

is non-relaxed. If $x(n) = 0$, we find that $y(n) = 1$. This is an indication that the system is nonlinear. This, in fact, is the conclusion reached when the linearity test is applied.

Causal versus noncausal systems. We begin with the definition of causal discrete-time systems.

Definition. A system is said to be *causal* if the output of the system at any time n [i.e., $y(n)$] depends only on present and past inputs [i.e., $x(n)$, $x(n-1)$, $x(n-2)$, ...], but does not depend on future inputs [i.e., $x(n+1)$, $x(n+2)$, ...]. In mathematical terms, the output of a causal system satisfies an equation of the form

$$y(n) = F[x(n), x(n-1), x(n-2), \ldots] \tag{2.2.44}$$

where $F[\,]$ is some arbitrary function.

If a system does not satisfy this definition, it is called *noncausal*. Such a system has an output that depends not only on present and past inputs but also on future inputs.

It is apparent that in real-time signal processing applications we cannot observe future values of the signal, and hence a noncausal system is physically unrealizable (i.e., it cannot be implemented). On the other hand, if the signal is recorded so that the processing is done off-line (nonreal time), it is possible to implement a noncausal system, since all values of the signal are available at the time of processing. This is often the case in the processing of geophysical signals and images.

EXAMPLE 2.2.6

Determine if the systems described by the following input–output equations are causal or noncausal.

(a) $y(n) = x(n) - x(n-1)$ **(b)** $y(n) = \sum_{k=-\infty}^{n} x(k)$ **(c)** $y(n) = ax(n)$ **(d)** $y(n) = x(n) + 3x(n+4)$ **(e)** $y(n) = x(n^2)$ **(f)** $y(n) = x(2n)$ **(g)** $y(n) = x(-n)$

Solution. The systems described in parts (a), (b), and (c) are clearly causal, since the output depends only on the present and past inputs. On the other hand, the systems in parts (d), (e), and (f) are clearly noncausal, since the output depends on future values of the input. The system in (g) is also noncausal, as we note by selecting, for example, $n = -1$, which yields $y(-1) = x(1)$. Thus the output at $n = -1$ depends on the input at $n = 1$, which is two units of time into the future.

Stable versus unstable systems. Stability is an important property that must be considered in any practical application of a system. Unstable systems usually exhibit erratic and extreme behavior and cause overflow in any practical implementation. Here, we define mathematically what we mean by a stable system, and later, in Section 2.3.6, we explore the implications of this definition for linear, time-invariant systems.

Definition. An arbitrary relaxed system is said to be bounded input–bounded output (BIBO) stable if and only if every bounded input produces a bounded output.

The condition that the input sequence $x(n)$ and the output sequence $y(n)$ are bounded is translated mathematically to mean that there exist some finite numbers, say M_x and M_y, such that

$$|x(n)| \leq M_x < \infty, \qquad |y(n)| \leq M_y < \infty \tag{2.2.45}$$

for all n. If, for some bounded input sequence $x(n)$, the output is unbounded (infinite), the system is classified as unstable.

EXAMPLE 2.2.7

Consider the nonlinear system described by the input–output equation

$$y(n) = y^2(n-1) + x(n)$$

As an input sequence we select the bounded signal

$$x(n) = C\delta(n)$$

where C is a constant. We also assume that $y(-1) = 0$. Then the output sequence is

$$y(0) = C, \quad y(1) = C^2, \quad y(2) = C^4, \quad \ldots, \quad y(n) = C^{2^n}$$

Clearly, the output is unbounded when $1 < |C| < \infty$. Therefore, the system is BIBO unstable, since a bounded input sequence has resulted in an unbounded output.

2.2.4 Interconnection of Discrete-Time Systems

Discrete-time systems can be interconnected to form larger systems. There are two basic ways in which systems can be interconnected: in cascade (series) or in parallel. These interconnections are illustrated in Fig. 2.2.10. Note that the two interconnected systems are different.

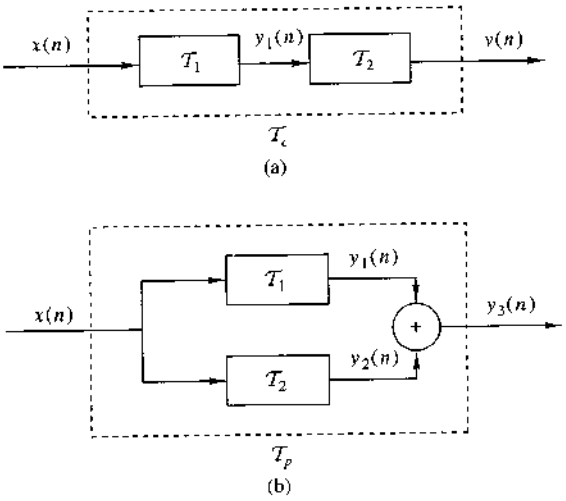


Figure 2.2.10
Cascade (a) and parallel (b) interconnections of systems.

In the cascade interconnection the output of the first system is

$$y_1(n) = \mathcal{T}_1[x(n)] \tag{2.2.46}$$

and the output of the second system is

$$\begin{aligned} y(n) &= \mathcal{T}_2[y_1(n)] \\ &= \mathcal{T}_2\{\mathcal{T}_1[x(n)]\} \end{aligned} \tag{2.2.47}$$

We observe that systems $\mathcal{T}_1$ and $\mathcal{T}_2$ can be combined or consolidated into a single overall system

$$\mathcal{T}_c \equiv \mathcal{T}_2\mathcal{T}_1 \tag{2.2.48}$$

Consequently, we can express the output of the combined system as

$$y(n) = \mathcal{T}_c[x(n)]$$

In general, the order in which the operations $\mathcal{T}_1$ and $\mathcal{T}_2$ are performed is important. That is,

$$\mathcal{T}_2\mathcal{T}_1 \neq \mathcal{T}_1\mathcal{T}_2$$

for arbitrary systems. However, if the systems $\mathcal{T}_1$ and $\mathcal{T}_2$ are linear and time invariant, then (a) $\mathcal{T}_c$ is time invariant and (b) $\mathcal{T}_2\mathcal{T}_1 = \mathcal{T}_1\mathcal{T}_2$, that is, the order in which the systems process the signal is not important. $\mathcal{T}_2\mathcal{T}_1$ and $\mathcal{T}_1\mathcal{T}_2$ yield identical output sequences.

The proof of (a) follows. The proof of (b) is given in Section 2.3.4. To prove time invariance, suppose that $\mathcal{T}_1$ and $\mathcal{T}_2$ are time invariant; then

$$x(n-k) \xrightarrow{\mathcal{T}_1} y_1(n-k)$$

and

$$y_1(n-k) \xrightarrow{\mathcal{T}_2} y(n-k)$$

Thus

$$x(n-k) \xrightarrow{\mathcal{T}_c=\mathcal{T}_2\mathcal{T}_1} y(n-k)$$

and therefore, $\mathcal{T}_c$ is time invariant.

In the parallel interconnection, the output of the system $\mathcal{T}_1$ is $y_1(n)$ and the output of the system $\mathcal{T}_2$ is $y_2(n)$. Hence the output of the parallel interconnection is

$$\begin{aligned} y_3(n) &= y_1(n) + y_2(n) \\ &= \mathcal{T}_1[x(n)] + \mathcal{T}_2[x(n)] \\ &= (\mathcal{T}_1 + \mathcal{T}_2)[x(n)] \\ &= \mathcal{T}_p[x(n)] \end{aligned}$$

where $\mathcal{T}_p = \mathcal{T}_1 + \mathcal{T}_2$.

In general, we can use parallel and cascade interconnection of systems to construct larger, more complex systems. Conversely, we can take a larger system and break it down into smaller subsystems for purposes of analysis and implementation. We shall use these notions later, in the design and implementation of digital filters.

2.3 Analysis of Discrete-Time Linear Time-Invariant Systems

In Section 2.2 we classified systems in accordance with a number of characteristic properties or categories, namely: linearity, causality, stability, and time invariance. Having done so, we now turn our attention to the analysis of the important class of linear, time-invariant (LTI) systems. In particular, we shall demonstrate that such systems are characterized in the time domain simply by their response to a unit sample sequence. We shall also demonstrate that any arbitrary input signal can be decomposed and represented as a weighted sum of unit sample sequences. As a consequence of the linearity and time-invariance properties of the system, the response of the system to any arbitrary input signal can be expressed in terms of the unit sample response of the system. The general form of the expression that relates the unit sample response of the system and the arbitrary input signal to the output signal, called the convolution sum or the convolution formula, is also derived. Thus we are able to determine the output of any linear, time-invariant system to any arbitrary input signal.

2.3.1 Techniques for the Analysis of Linear Systems

There are two basic methods for analyzing the behavior or response of a linear system to a given input signal. One method is based on the direct solution of the input–output equation for the system, which, in general, has the form

$$y(n) = F[y(n-1), y(n-2), \ldots, y(n-N), x(n), x(n-1), \ldots, x(n-M)]$$

where $F[\cdot]$ denotes some function of the quantities in brackets. Specifically, for an LTI system, we shall see later that the general form of the input–output relationship is

$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k) \tag{2.3.1}$$

where $\{a_k\}$ and $\{b_k\}$ are constant parameters that specify the system and are independent of $x(n)$ and $y(n)$. The input–output relationship in (2.3.1) is called a difference equation and represents one way to characterize the behavior of a discrete-time LTI system. The solution of (2.3.1) is the subject of Section 2.4.

The second method for analyzing the behavior of a linear system to a given input signal is first to decompose or resolve the input signal into a sum of elementary signals. The elementary signals are selected so that the response of the system to each signal component is easily determined. Then, using the linearity property of the system, the responses of the system to the elementary signals are added to obtain the total response of the system to the given input signal. This second method is the one described in this section.

To elaborate, suppose that the input signal $x(n)$ is resolved into a weighted sum of elementary signal components $\{x_k(n)\}$ so that

$$x(n) = \sum_{k} c_k x_k(n) \tag{2.3.2}$$

where the $\{c_k\}$ are the set of amplitudes (weighting coefficients) in the decomposition of the signal $x(n)$. Now suppose that the response of the system to the elementary signal component $x_k(n)$ is $y_k(n)$. Thus,

$$y_k(n) \equiv \mathcal{T}[x_k(n)] \tag{2.3.3}$$

assuming that the system is relaxed and that the response to $c_k x_k(n)$ is $c_k y_k(n)$, as a consequence of the scaling property of the linear system.

Finally, the total response to the input $x(n)$ is

$$\begin{aligned} y(n) = \mathcal{T}[x(n)] &= \mathcal{T}\left[\sum_{k} c_k x_k(n)\right] \\ &= \sum_{k} c_k \mathcal{T}[x_k(n)] \\ &= \sum_{k} c_k y_k(n) \end{aligned} \tag{2.3.4}$$

In (2.3.4) we used the additivity property of the linear system.

Although to a large extent, the choice of the elementary signals appears to be arbitrary, our selection is heavily dependent on the class of input signals that we wish to consider. If we place no restriction on the characteristics of the input signals,

their resolution into a weighted sum of unit sample (impulse) sequences proves to be mathematically convenient and completely general. On the other hand, if we restrict our attention to a subclass of input signals, there may be another set of elementary signals that is more convenient mathematically in the determination of the output. For example, if the input signal $x(n)$ is periodic with period N, we have already observed in Section 1.3.3 that a mathematically convenient set of elementary signals is the set of exponentials

$$x_k(n) = e^{j\omega_k n}, \qquad k = 0, 1, \ldots, N-1 \tag{2.3.5}$$

where the frequencies $\{\omega_k\}$ are harmonically related, that is,

$$\omega_k = \left(\frac{2\pi}{N}\right)k, \qquad k = 0, 1, \ldots, N-1 \tag{2.3.6}$$

The frequency $2\pi/N$ is called the fundamental frequency, and all higher-frequency components are multiples of the fundamental frequency component. This subclass of input signals is considered in more detail later.

For the resolution of the input signal into a weighted sum of unit sample sequences, we must first determine the response of the system to a unit sample sequence and then use the scaling and multiplicative properties of the linear system to determine the formula for the output given any arbitrary input. This development is described in detail as follows.

2.3.2 Resolution of a Discrete-Time Signal into Impulses

Suppose we have an arbitrary signal $x(n)$ that we wish to resolve into a sum of unit sample sequences. To utilize the notation established in the preceding section, we select the elementary signals $x_k(n)$ to be

$$x_k(n) = \delta(n-k) \tag{2.3.7}$$

where k represents the delay of the unit sample sequence. To handle an arbitrary signal $x(n)$ that may have nonzero values over an infinite duration, the set of unit impulses must also be infinite, to encompass the infinite number of delays.

Now suppose that we multiply the two sequences $x(n)$ and $\delta(n-k)$. Since $\delta(n-k)$ is zero everywhere except at $n = k$, where its value is unity, the result of this multiplication is another sequence that is zero everywhere except at $n = k$, where its value is $x(k)$, as illustrated in Fig. 2.3.1. Thus

$$x(n)\delta(n-k) = x(k)\delta(n-k) \tag{2.3.8}$$

is a sequence that is zero everywhere except at $n = k$, where its value is $x(k)$. If we repeat the multiplication of $x(n)$ with $\delta(n-m)$, where m is another delay ($m \neq k$), the result will be a sequence that is zero everywhere except at $n = m$, where its value is $x(m)$. Hence

$$x(n)\delta(n-m) = x(m)\delta(n-m) \tag{2.3.9}$$

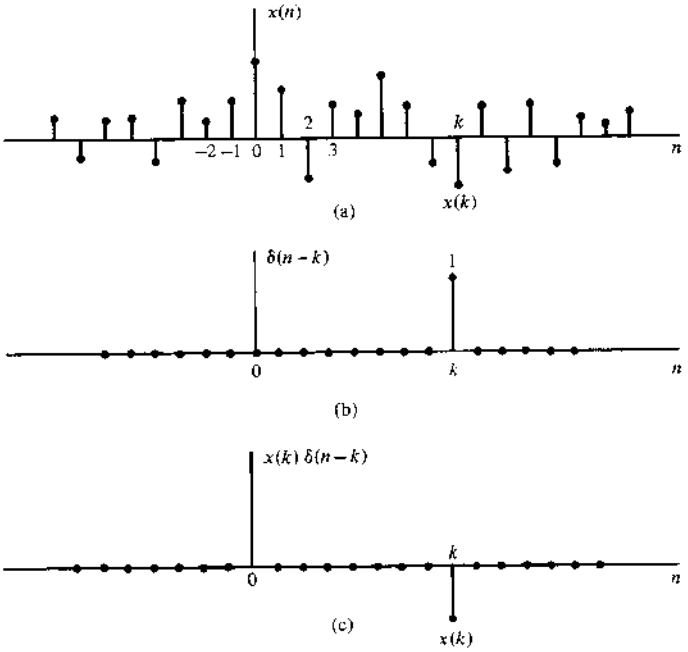


Figure 2.3.1 Multiplication of a signal $x(n)$ with a shifted unit sample sequence.

In other words, each multiplication of the signal $x(n)$ by a unit impulse at some delay k, [i.e., $\delta(n-k)$], in essence picks out the single value $x(k)$ of the signal $x(n)$ at the delay where the unit impulse is nonzero. Consequently, if we repeat this multiplication over all possible delays, $-\infty < k < \infty$, and sum all the product sequences, the result will be a sequence equal to the sequence $x(n)$, that is,

$$x(n) = \sum_{k=-\infty}^{\infty} x(k)\delta(n-k) \tag{2.3.10}$$

We emphasize that the right-hand side of (2.3.10) is the summation of an infinite number of scaled unit sample sequences where the unit sample sequence $\delta(n-k)$ has an amplitude value of $x(k)$. Thus the right-hand side of (2.3.10) gives the resolution or decomposition of any arbitrary signal $x(n)$ into a weighted (scaled) sum of shifted unit sample sequences.

EXAMPLE 2.3.1

Consider the special case of a finite-duration sequence given as

$$x(n) = \{2, \underset{\uparrow}{4}, 0, 3\}$$

Resolve the sequence $x(n)$ into a sum of weighted impulse sequences

Solution. Since the sequence $x(n)$ is nonzero for the time instants $n = -1, 0, 2$, we need three impulses at delays $k = -1, 0$. Following (2.3.10) we find that

$$x(n) = 2\delta(n+1) + 4\delta(n) + 3\delta(n-2)$$

2.3.3 Response of LTI Systems to Arbitrary Inputs: The Convolution Sum

Having resolved an arbitrary input signal $x(n)$ into a weighted sum of impulses, we are now ready to determine the response of any relaxed linear system to any input signal. First, we denote the response $y(n, k)$ of the system to the input unit sample sequence at $n = k$ by the special symbol $h(n, k)$, $-\infty < k < \infty$. That is,

$$y(n, k) \equiv h(n, k) = \mathcal{T}[\delta(n-k)] \tag{2.3.11}$$

In (2.3.11) we note that n is the time index and k is a parameter showing the location of the input impulse. If the impulse at the input is scaled by an amount $c_k \equiv x(k)$, the response of the system is the correspondingly scaled output, that is,

$$c_k h(n, k) = x(k)h(n, k) \tag{2.3.12}$$

Finally, if the input is the arbitrary signal $x(n)$ that is expressed as a sum of weighted impulses, that is,

$$x(n) = \sum_{k=-\infty}^{\infty} x(k)\delta(n-k) \tag{2.3.13}$$

then the response of the system to $x(n)$ is the corresponding sum of weighted outputs, that is,

$$\begin{aligned} y(n) = \mathcal{T}[x(n)] &= \mathcal{T}\left[\sum_{k=-\infty}^{\infty} x(k)\delta(n-k)\right] \\ &= \sum_{k=-\infty}^{\infty} x(k)\mathcal{T}[\delta(n-k)] \qquad (2.3.14) \\ &= \sum_{k=-\infty}^{\infty} x(k)h(n, k) \end{aligned}$$

Clearly, (2.3.14) follows from the superposition property of linear systems, and is known as the *superposition summation*.

We note that (2.3.14) is an expression for the response of a linear system to any arbitrary input sequence $x(n)$. This expression is a function of both $x(n)$ and the responses $h(n, k)$ of the system to the unit impulses $\delta(n-k)$ for $-\infty < k < \infty$. In deriving (2.3.14) we used the linearity property of the system but not its time-invariance property. Thus the expression in (2.3.14) applies to any relaxed linear (time-variant) system.

If, in addition, the system is time invariant, the formula in (2.3.14) simplifies considerably. In fact, if the response of the LTI system to the unit sample sequence $\delta(n)$ is denoted as $h(n)$, that is,

$$h(n) \equiv \mathcal{T}[\delta(n)] \tag{2.3.15}$$

then by the time-invariance property, the response of the system to the delayed unit sample sequence $\delta(n-k)$ is

$$h(n-k) = \mathcal{T}[\delta(n-k)] \tag{2.3.16}$$

Consequently, the formula in (2.3.14) reduces to

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k) \tag{2.3.17}$$

Now we observe that the relaxed LTI system is completely characterized by a single function $h(n)$, namely, its response to the unit sample sequence $\delta(n)$. In contrast, the general characterization of the output of a time-variant, linear system requires an infinite number of unit sample response functions, $h(n, k)$, one for each possible delay.

The formula in (2.3.17) that gives the response $y(n)$ of the LTI system as a function of the input signal $x(n)$ and the unit sample (impulse) response $h(n)$ is called a *convolution sum*. We say that the input $x(n)$ is convolved with the impulse response $h(n)$ to yield the output $y(n)$. We shall now explain the procedure for computing the response $y(n)$, both mathematically and graphically, given the input $x(n)$ and the impulse response $h(n)$ of the system.

Suppose that we wish to compute the output of the system at some time instant, say $n = n_0$. According to (2.3.17), the response at $n = n_0$ is given as

$$y(n_0) = \sum_{k=-\infty}^{\infty} x(k)h(n_0-k) \tag{2.3.18}$$

Our first observation is that the index in the summation is k, and hence both the input signal $x(k)$ and the impulse response $h(n_0 - k)$ are functions of k. Second, we observe that the sequences $x(k)$ and $h(n_0 - k)$ are multiplied together to form a product sequence. The output $y(n_0)$ is simply the sum over all values of the product sequence. The sequence $h(n_0 - k)$ is obtained from $h(k)$ by, first, folding $h(k)$ about $k = 0$ (the time origin), which results in the sequence $h(-k)$. The folded sequence is then shifted by n_0 to yield $h(n_0 - k)$. To summarize, the process of computing the convolution between $x(k)$ and $h(k)$ involves the following four steps.

1. *Folding.* Fold $h(k)$ about $k = 0$ to obtain $h(-k)$.
2. *Shifting.* Shift $h(-k)$ by n_0 to the right (left) if n_0 is positive (negative), to obtain $h(n_0 - k)$.
3. *Multiplication.* Multiply $x(k)$ by $h(n_0 - k)$ to obtain the product sequence $v_{n_0}(k) \equiv x(k)h(n_0 - k)$.
4. *Summation.* Sum all the values of the product sequence $v_{n_0}(k)$ to obtain the value of the output at time $n = n_0$.

We note that this procedure results in the response of the system at a single time instant, say $n = n_0$. In general, we are interested in evaluating the response of the system over all time instants $-\infty < n < \infty$. Consequently, steps 2 through 4 in the summary must be repeated, for all possible time shifts $-\infty < n < \infty$.

In order to gain a better understanding of the procedure for evaluating the convolution sum, we shall demonstrate the process graphically. The graphs will aid us in explaining the four steps involved in the computation of the convolution sum.

EXAMPLE 2.3.2

The impulse response of a linear time-invariant system is

$$h(n) = \{1, \underset{\uparrow}{2}, 1, -1\} \tag{2.3.19}$$

Determine the response of the system to the input signal

$$x(n) = \{\underset{\uparrow}{1}, 2, 3, 1\} \tag{2.3.20}$$

Solution. We shall compute the convolution according to the formula (2.3.17), but we shall use graphs of the sequences to aid us in the computation. In Fig. 2.3.2(a) we illustrate the input signal sequence $x(k)$ and the impulse response $h(k)$ of the system, using k as the time index in order to be consistent with (2.3.17).

The first step in the computation of the convolution sum is to fold $h(k)$. The folded sequence $h(-k)$ is illustrated in Fig. 2.3.2(b). Now we can compute the output at $n = 0$, according to (2.3.17), which is

$$y(0) = \sum_{k=-\infty}^{\infty} x(k)h(-k) \tag{2.3.21}$$

Since the shift $\bar{n} = 0$, we use $h(-k)$ directly without shifting it. The product sequence

$$v_0(k) \equiv x(k)h(-k) \tag{2.3.22}$$

is also shown in Fig. 2.3.2(b). Finally, the sum of all the terms in the product sequence yields

$$y(0) = \sum^{\infty} h = -\infty v_0(k) = 4$$

We continue the computation by evaluating the response of the system at $n = 1$. According to (2.3.17),

$$y(1) = \sum_{k=-\infty}^{\infty} x(k)h(1-k) \tag{2.3.23}$$

The sequence $h(1-k)$ is simply the folded sequence $h(-k)$ shifted to the right by one unit in time. This sequence is illustrated in Fig. 2.3.2(c). The product sequence

$$v_1(k) = x(k)h(1-k) \tag{2.3.24}$$

is also illustrated in Fig. 2.3.2(c). Finally, the sum of all the values in the product sequence yields

$$y(1) = \sum_{k=-\infty}^{\infty} v_1(k) = 8$$

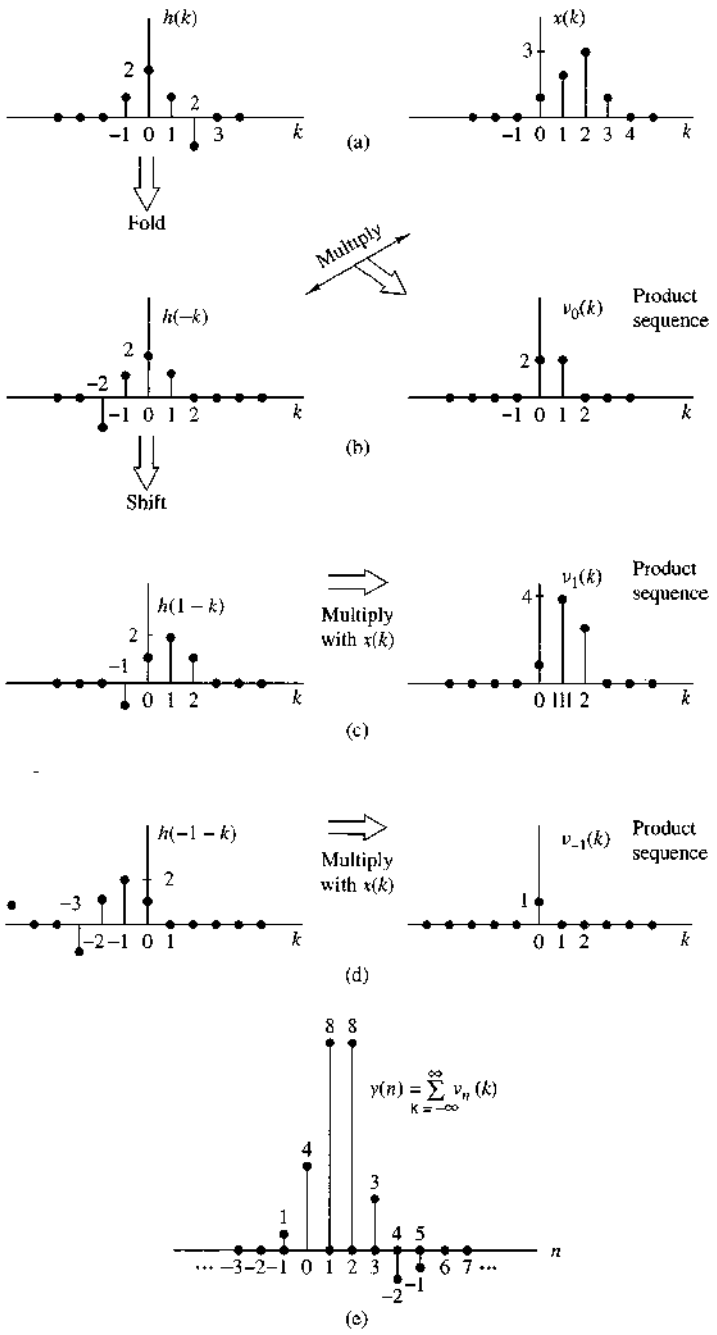


Figure 2.3.2 Graphical computation of convolution

In a similar manner, we obtain $y(2)$ by shifting $h(-k)$ two units to the right, forming the product sequence $v_2(k) = x(k)h(2-k)$ and then summing all the terms in the product sequence obtaining $y(2) = 8$. By shifting $h(-k)$ farther to the right, multiplying the corresponding sequence, and summing over all the values of the resulting product sequences, we obtain $y(3) = 3$, $y(4) = -2$, $y(5) = -1$. For $n > 5$, we find that $y(n) = 0$ because the product sequences contain all zeros. Thus we have obtained the response $y(n)$ for $n > 0$.

Next we wish to evaluate $y(n)$ for $n < 0$. We begin with $n = -1$. Then

$$y(-1) = \sum_{k=-\infty}^{\infty} x(k)h(-1-k) \tag{2.3.25}$$

Now the sequence $h(-1-k)$ is simply the folded sequence $h(-k)$ shifted one time unit to the left. The resulting sequence is illustrated in Fig. 2.3.2(d). The corresponding product sequence is also shown in Fig. 2.3.2(d). Finally, summing over the values of the product sequence, we obtain

$$y(-1) = 1$$

From observation of the graphs of Fig. 2.3.2, it is clear that any further shifts of $h(-1-k)$ to the left always result in an all-zero product sequence, and hence

$$y(n) = 0 \qquad \text{for } n \leq -2$$

Now we have the entire response of the system for $-\infty < n < \infty$, which we summarize below as

$$y(n) = \{\ldots, 0, 0, 1, \underset{\uparrow}{4}, 8, 8, 3, -2, -1, 0, 0, \ldots\} \tag{2.3.26}$$

In Example 2.3.2 we illustrated the computation of the convolution sum, using graphs of the sequences to aid us in visualizing the steps involved in the computation procedure.

Before working out another example, we wish to show that the convolution operation is commutative in the sense that it is irrelevant which of the two sequences is folded and shifted. Indeed, if we begin with (2.3.17) and make a change in the variable of the summation, from k to m, by defining a new index $m = n - k$, then $k = n - m$ and (2.3.17) becomes

$$y(n) = \sum_{m=-\infty}^{\infty} x(n-m)h(m) \tag{2.3.27}$$

Since m is a dummy index, we may simply replace m by k so that

$$y(n) = \sum_{k=-\infty}^{\infty} x(n-k)h(k) \tag{2.3.28}$$

The expression in (2.3.28) involves leaving the impulse response $h(k)$ unaltered, while the input sequence is folded and shifted. Although the output $y(n)$ in (2.3.28)

is identical to (2.3.17), the product sequences in the two forms of the convolution formula are not identical. In fact, if we define the two product sequences as

$$v_n(k) = x(k)h(n-k)$$

$$w_n(k) = x(n-k)h(k)$$

it can be easily shown that

$$v_n(k) = w_n(n-k)$$

and therefore,

$$y(n) = \sum_{k=-\infty}^{\infty} v_n(k) = \sum_{k=-\infty}^{\infty} w_n(n-k)$$

since both sequences contain the same sample values in a different arrangement. The reader is encouraged to rework Example 2.3.2 using the convolution sum in (2.3.28).

EXAMPLE 2.3.3

Determine the output $y(n)$ of a relaxed linear time-invariant system with impulse response

$$h(n) = a^n u(n), \quad |a|1$$

when the input is a unit step sequence, that is,

$$x(n) = u(n)$$

Solution. In this case both $h(n)$ and $x(n)$ are infinite-duration sequences. We use the form of the convolution formula given by (2.3.28) in which $x(k)$ is folded. The sequences $h(k)$, $x(k)$, and $x(-k)$ are shown in Fig. 2.3.3. The product sequences $v_0(k)$, $v_1(k)$, and $v_2(k)$ corresponding to $x(-k)h(k)$, $x(1-k)h(k)$, and $x(2-k)h(k)$ are illustrated in Fig. 2.3.3(c), (d), and (e), respectively. Thus we obtain the outputs

$$y(0) = 1$$

$$y(1) = 1 + a$$

$$y(2) = 1 + a + a^2$$

Clearly, for $n > 0$, the output is

$$y(n) = 1 + a + a^2 + \cdots + a^n$$

$$= \frac{1 - a^{n+1}}{1 - a} \tag{2.3.29}$$

On the other hand, for $n < 0$, the product sequences consist of all zeros. Hence

$$y(n) = 0, \qquad n < 0$$

A graph of the output $y(n)$ is illustrated in Fig. 2.3.3(f) for the case $0 < a < 1$. Note the exponential rise in the output as a function of n. Since $|a| < 1$, the final value of the output as n approaches infinity is

$$y(\infty) = \lim_{n\to\infty} y(n) = \frac{1}{1-a} \tag{2.3.30}$$

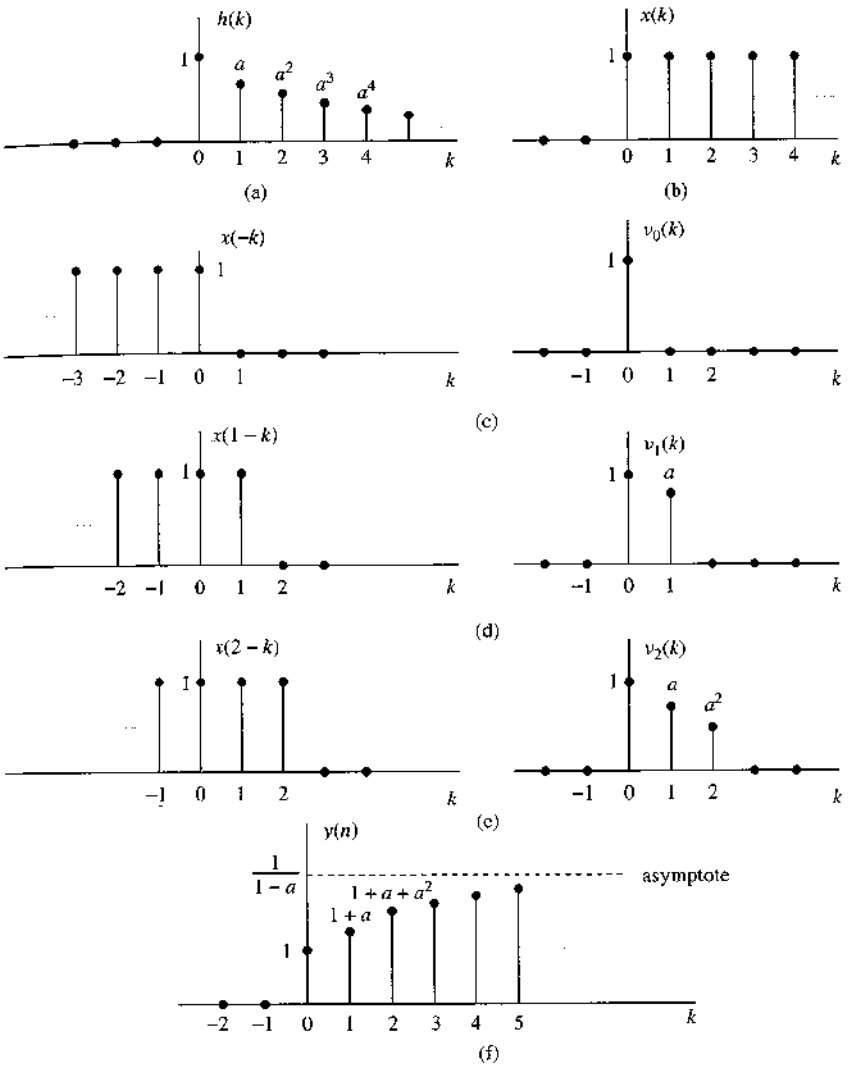


Figure 2.3.3 Graphical computation of convolution in Example 2.3.3.

To summarize, the convolution formula provides us with a means for computing the response of a relaxed, linear time-invariant system to any arbitrary input signal $x(n)$. It takes one of two equivalent forms, either (2.3.17) or (2.3.28), where $x(n)$ is the input signal to the system, $h(n)$ is the impulse response of the system, and $y(n)$ is the *output* of the system in response to the input signal $x(n)$. The evaluation of the convolution formula involves four operations, namely: *folding* either the impulse response as specified by (2.3.17) or the input sequence as specified by (2.3.28) to yield either $h(-k)$ or $x(-k)$, respectively, *shifting* the folded sequence by n units in time to yield either $h(n-k)$ or $x(n-k)$, *multiplying* the two sequences to yield the product

sequence, either $x(k)h(n-k)$ or $x(n-k)h(k)$, and finally *summing* all the values in the product sequence to yield the output $y(n)$ of the system at time n. The folding operation is done only once. However, the other three operations are repeated for all possible shifts $-\infty < n < \infty$ in order to obtain $y(n)$ for $-\infty < n < \infty$.

2.3.4 Properties of Convolution and the Interconnection of LTI Systems

In this section we investigate some important properties of convolution and interpret these properties in terms of interconnecting linear time-invariant systems. We should stress that these properties hold for every input signal.

It is convenient to simplify the notation by using an asterisk to denote the convolution operation. Thus

$$y(n) = x(n) * h(n) \equiv \sum_{k=-\infty}^{\infty} x(k)h(n-k) \tag{2.3.31}$$

In this notation the sequence following the asterisk [i.e., the impulse response $h(n)$] is folded and shifted. The input to the system is $x(n)$. On the other hand, we also showed that

$$y(n) = h(n) * x(n) \equiv \sum_{k=-\infty}^{\infty} h(k)x(n-k) \tag{2.3.32}$$

In this form of the convolution formula, it is the input signal that is folded. Alternatively, we may interpret this form of the convolution formula as resulting from an interchange of the roles of $x(n)$ and $h(n)$. In other words, we may regard $x(n)$ as the impulse response of the system and $h(n)$ as the excitation or input signal. Figure 2.3.4 illustrates this interpretation.

Identity and Shifting Properties. We also note that the unit sample sequence $\delta(n)$ is the identity element for convolution, that is

$$y(n) = x(n) * \delta(n) = x(n)$$

If we shift $\delta(n)$ by k, the convolution sequence is shifted also by k, that is

$$x(n) * \delta(n-k) = y(n-k) = x(n-k)$$

We can view convolution more abstractly as a mathematical operation between two signal sequences, say $x(n)$ and $h(n)$, that satisfies a number of properties. The property embodied in (2.3.31) and (2.3.32) is called the commutative law.

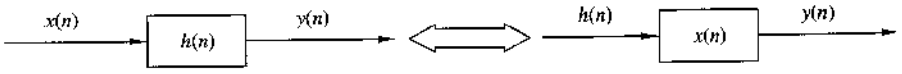


Figure 2.3.4 Interpretation of the commutative property of convolution.

Commutative law

$$x(n) * h(n) = h(n) * x(n) \tag{2.3.33}$$

Viewed mathematically, the convolution operation also satisfies the associative law, which can be stated as follows.

Associative law

$$[x(n) * h_1(n)] * h_2(n) = x(n) * [h_1(n) * h_2(n)] \tag{2.3.34}$$

From a physical point of view, we can interpret $x(n)$ as the input signal to a linear time-invariant system with impulse response $h_1(n)$. The output of this system, denoted as $y_1(n)$, becomes the input to a second linear time-invariant system with impulse response $h_2(n)$. Then the output is

$$\begin{aligned} y(n) &= y_1(n) * h_2(n) \\ &= [x(n) * h_1(n)] * h_2(n) \end{aligned}$$

which is precisely the left-hand side of (2.3.34). Thus the left-hand side of (2.3.34) corresponds to having two linear time-invariant systems in cascade. Now the right-hand side of (2.3.34) indicates that the input $x(n)$ is applied to an equivalent system having an impulse response, say $h(n)$, which is equal to the convolution of the two impulse responses. That is,

$$h(n) = h_1(n) * h_2(n)$$

and

$$y(n) = x(n) * h(n)$$

Furthermore, since the convolution operation satisfies the commutative property, one can interchange the order of the two systems with responses $h_1(n)$ and $h_2(n)$ without altering the overall input–output relationship. Figure 2.3.5 graphically illustrates the associative property.

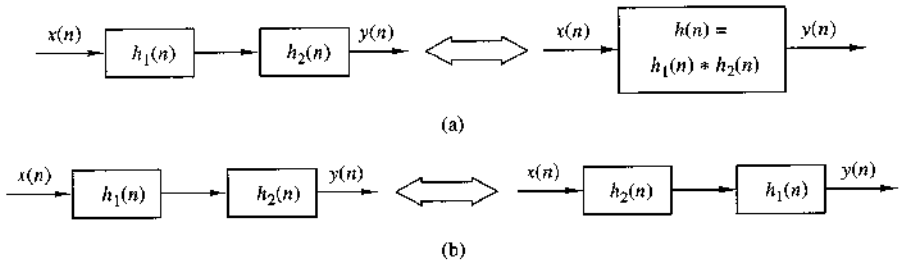


Figure 2.3.5 Implications of the associative (a) and the associative and commutative (b) properties of convolution.

EXAMPLE 2.3.4

Determine the impulse response for the cascade of two linear time-invariant systems having impulse responses

$$h_1(n) = (\frac{1}{2})^n u(n)$$

and

$$h_2(n) = (\frac{1}{4})^n u(n)$$

Solution. To determine the overall impulse response of the two systems in cascade, we simply convolve $h_1(n)$ with $h_2(n)$. Hence

$$h(n) = \sum_{k=-\infty}^{\infty} h_1(k)h_2(n-k)$$

where $h_2(n)$ is folded and shifted. We define the product sequence

$$v_n(k) = h_1(k)h_2(n-k)$$

$$= (\frac{1}{2})^k (\frac{1}{4})^{n-k}$$

which is nonzero for $k \geq 0$ and $n - k \geq 0$ or $n \geq k \geq 0$. On the other hand, for $n < 0$, we have $v_n(k) = 0$ for all k, and hence

$$h(n) = 0, n < 0$$

For $n \geq k \geq 0$, the sum of the values of the product sequence $v_n(k)$ over all k yields

$$h(n) = \sum_{k=0}^{n} (\frac{1}{2})^k (\frac{1}{4})^{n-k}$$

$$= (\frac{1}{4})^n \sum_{k=0}^{n} 2^k$$

$$= (\frac{1}{4})^n (2^{n+1} - 1)$$

$$= (\frac{1}{2})^n [2 - (\frac{1}{2})^n], \quad n \geq 0$$

The generalization of the associative law to more than two systems in cascade follows easily from the discussion given above. Thus if we have L linear time-invariant systems in cascade with impulse responses $h_1(n), h_2(n), \ldots, h_L(n)$, there is an equivalent linear time-invariant system having an impulse response that is equal to the $(L-1)$-fold convolution of the impulse responses. That is,

$$h(n) = h_1(n) * h_2(n) * \cdots * h_L(n) \tag{2.3.35}$$

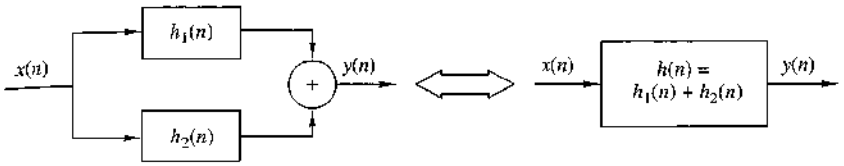


Figure 2.3.6 Interpretation of the distributive property of convolution: two LTI systems connected in parallel can be replaced by a single system with $h(n) = h_1(n) + h_2(n)$.

The commutative law implies that the order in which the convolutions are performed is immaterial. Conversely, any linear time-invariant system can be decomposed into a cascade interconnection of subsystems. A method for accomplishing the decomposition will be described later.

Another property that is satisfied by the convolution operation is the distributive law, which may be stated as follows.

Distributive law

$$x(n) * [h_1(n) + h_2(n)] = x(n) * h_1(n) + x(n) * h_2(n) \tag{2.3.36}$$

Interpreted physically, this law implies that if we have two linear time-invariant systems with impulse responses $h_1(n)$ and $h_2(n)$ excited by the same input signal $x(n)$, the sum of the two responses is identical to the response of an overall system with impulse response

$$h(n) = h_1(n) + h_2(n)$$

Thus the overall system is viewed as a parallel combination of the two linear time-invariant systems as illustrated in Fig. 2.3.6.

The generalization of (2.3.36) to more than two linear time-invariant systems in parallel follows easily by mathematical induction. Thus the interconnection of L linear time-invariant systems in parallel with impulse responses $h_1(n), h_2(n), \ldots, h_L(n)$ and excited by the same input $x(n)$ is equivalent to one overall system with impulse response

$$h(n) = \sum_{j=1}^{L} h_j(n) \tag{2.3.37}$$

Conversely, any linear time-invariant system can be decomposed into a parallel interconnection of subsystems.

2.3.5 Causal Linear Time-Invariant Systems

In Section 2.2.3 we defined a causal system as one whose output at time n depends only on present and past inputs but does not depend on future inputs. In other words, the output of the system at some time instant n, say $n = n_0$, depends only on values of $x(n)$ for $n \leq n_0$.

In the case of a linear time-invariant system, causality can be translated to a condition on the impulse response. To determine this relationship, let us consider a

linear time-invariant system having an output at time $n = n_0$ given by the convolution formula

$$y(n_0) = \sum_{k=-\infty}^{\infty} h(k)x(n_0 - k)$$

Suppose that we subdivide the sum into two sets of terms, one set involving present and past values of the input [i.e., $x(n)$ for $n \leq n_0$] and one set involving future values of the input [i.e., $x(n)$, $n > n_0$]. Thus we obtain

$$\begin{aligned} y(n_0) &= \sum_{k=0}^{\infty} h(k)x(n_0 - k) + \sum_{k=-\infty}^{-1} h(k)x(n_0 - k) \\ &= [h(0)x(n_0) + h(1)x(n_0 - 1) + h(2)x(n_0 - 2) + \cdots] \\ &\quad + [h(-1)x(n_0 + 1) + h(-2)x(n_0 + 2) + \cdots] \end{aligned}$$

We observe that the terms in the first sum involve $x(n_0)$, $x(n_0 - 1), \ldots$, which are the present and past values of the input signal. On the other hand, the terms in the second sum involve the input signal components $x(n_0+1), x(n_0+2), \ldots$. Now, if the output at time $n = n_0$ is to depend only on the present and past inputs, then, clearly, the impulse response of the system must satisfy the condition

$$h(n) = 0, \qquad n < 0 \tag{2.3.38}$$

Since $h(n)$ is the response of the relaxed linear time-invariant system to a unit impulse applied at $n = 0$, it follows that $h(n) = 0$ for $n < 0$ is both a necessary and a sufficient condition for causality. Hence an *LTI system is causal if and only if its impulse response is zero for negative values of n*.

Since for a causal system, $h(n) = 0$ for $n < 0$, the limits on the summation of the convolution formula may be modified to reflect this restriction. Thus we have the two equivalent forms

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n - k) \tag{2.3.39}$$

$$= \sum_{k=-\infty}^{n} x(k)h(n - k) \tag{2.3.40}$$

As indicated previously, causality is required in any real-time signal processing application, since at any given time n we have no access to future values of the input signal. Only the present and past values of the input signal are available in computing the present output.

It is sometimes convenient to call a sequence that is zero for $n < 0$, a *causal sequence*, and one that is nonzero for $n < 0$ and $n > 0$, a *noncausal sequence*. This terminology means that such a sequence could be the unit sample response of a causal or a noncausal system, respectively.

If the input to a causal linear time-invariant system is a causal sequence [i.e., if $x(n) = 0$ for $n < 0$], the limits on the convolution formula are further restricted. In this case the two equivalent forms of the convolution formula become

$$y(n) = \sum_{k=0}^{n} h(k)x(n-k) \tag{2.3.41}$$

$$= \sum_{k=0}^{n} x(k)h(n-k) \tag{2.3.42}$$

We observe that in this case, the limits on the summations for the two alternative forms are identical, and the upper limit is growing with time. Clearly, the response of a causal system to a causal input sequence is causal, since $y(n) = 0$ for $n < 0$.

EXAMPLE 2.3.5

Determine the unit step response of the linear time-invariant system with impulse response

$$h(n) = a^n u(n), \qquad |a| < 1$$

Solution. Since the input signal is a unit step, which is a causal signal, and the system is also causal, we can use one of the special forms of the convolution formula, either (2.3.41) or (2.3.42). Since $x(n) = 1$ for $n \geq 0$, (2.3.41) is simpler to use. Because of the simplicity of this problem, one can skip the steps involved with sketching the folded and shifted sequences. Instead, we use direct substitution of the signals sequences in (2.3.41) and obtain

$$y(n) = \sum_{k=0}^{n} a^k$$

$$= \frac{1 - a^{n+1}}{1 - a}$$

and $y(n) = 0$ for $n < 0$. We note that this result is identical to that obtained in Example 2.3.3. In this simple case, however, we computed the convolution algebraically without resorting to the detailed procedure outlined previously.

2.3.6 Stability of Linear Time-Invariant Systems

As indicated previously, stability is an important property that must be considered in any practical implementation of a system. We defined an arbitrary relaxed system as BIBO stable if and only if its output sequence $y(n)$ is bounded for every bounded input $x(n)$.

If $x(n)$ is bounded, there exists a constant M_x such that

$$|x(n)| \leq M_x < \infty$$

Similarly, if the output is bounded, there exists a constant M_y such that

$$|y(n)| < M_y < \infty$$

for all n.

Now, given such a bounded input sequence $x(n)$ to a linear time-invariant system, let us investigate the implications of the definition of stability on the characteristics of the system. Toward this end, we work again with the convolution formula

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k)$$

If we take the absolute value of both sides of this equation, we obtain

$$|y(n)| = \left| \sum_{k=-\infty}^{\infty} h(k)x(n-k) \right|$$

Now, the absolute value of the sum of terms is always less than or equal to the sum of the absolute values of the terms. Hence

$$|y(n)| \leq \sum_{k=-\infty}^{\infty} |h(k)||x(n-k)|$$

If the input is bounded, there exists a finite number M_x such that $|x(n)| \leq M_x$. By substituting this upper bound for $x(n)$ in the equation above, we obtain

$$|y(n)| \leq M_x \sum_{k=-\infty}^{\infty} |h(k)|$$

From this expression we observe that the output is bounded if the impulse response of the system satisfies the condition

$$S_h \equiv \sum_{k=-\infty}^{\infty} |h(k)| < \infty \tag{2.3.43}$$

That is, *a linear time-invariant system is stable if its impulse response is absolutely summable*. This condition is not only sufficient but it is also necessary to ensure the stability of the system. Indeed, we shall show that if $S_h = \infty$, there is a bounded input for which the output is not bounded. We choose the bounded input

$$x(n) = \begin{cases} \dfrac{h^*(-n)}{|h(-n)|}, & h(n) \neq 0 \\ 0, & h(n) = 0 \end{cases}$$

where $h^*(n)$ is the complex conjugate of $h(n)$. It is sufficient to show that there is one value of n for which $y(n)$ is unbounded. For $n = 0$ we have

$$y(0) = \sum_{k=-\infty}^{\infty} x(-k)h(k) = \sum_{k=-\infty}^{\infty} \frac{|h(k)|^2}{|h(k)|} = S_h$$

Thus, if $S_h = \infty$, a bounded input produces an unbounded output since $y(0) = \infty$.

The condition in (2.3.43) implies that the impulse response $h(n)$ goes to zero as n approaches infinity. As a consequence, the output of the system goes to zero as n approaches infinity if the input is set to zero beyond $n > n_0$. To prove this, suppose that $|x(n)| < M_x$ for $n < n_0$ and $x(n) = 0$ for $n \geq n_0$. Then, at $n = n_0 + N$, the system output is

$$y(n_0 + N) = \sum_{k=-\infty}^{N-1} h(k)x(n_0 + N - k) + \sum_{k=N}^{\infty} h(k)x(n_0 + N - k)$$

But the first sum is zero since $x(n) = 0$ for $n \geq n_0$. For the remaining part, we take the absolute value of the output, which is

$$|y(n_0 + N)| = \left| \sum_{k=N}^{\infty} h(k)x(n_0 + N - k) \right| \leq \sum_{k=N}^{\infty} |h(k)||x(n_0 + N - k)|$$

$$\leq M_x \sum_{k=N}^{\infty} |h(k)|$$

Now, as N approaches infinity,

$$\lim_{N \to \infty} \sum_{k=N}^{\infty} |h(n)| = 0$$

and hence

$$\lim_{N \to \infty} |y(n_0 + N)| = 0$$

This result implies that any excitation at the input to the system, which is of a finite duration, produces an output that is "transient" in nature; that is, its amplitude decays with time and dies out eventually, when the system is stable.

EXAMPLE 2.3.6

Determine the range of values of the parameter a for which the linear time-invariant system with impulse response

$$h(n) = a^n u(n)$$

is stable.

Solution. First, we note that the system is causal. Consequently, the lower index on the summation in (2.3.43) begins with $k = 0$. Hence

$$\sum_{k=0}^{\infty} |a^k| = \sum_{k=0}^{\infty} |a|^k = 1 + |a| + |a|^2 + \cdots$$

Clearly, this geometric series converges to

$$\sum_{k=0}^{\infty} |a|^k = \frac{1}{1 - |a|}$$

provided that $|a| < 1$. Otherwise, it diverges. Therefore, the system is stable if $|a| < 1$. Otherwise, it is unstable. In effect, $h(n)$ must decay exponentially toward zero as n approaches infinity for the system to be stable.

EXAMPLE 2.3.7

Determine the range of values of a and b for which the linear time-invariant system with impulse response

$$h(n) = \begin{cases} a^n, & n \geq 0 \\ b^n, & n < 0 \end{cases}$$

is stable.

Solution. This system is noncasual. The condition on stability given by (2.3.43) yields

$$\sum_{n=-\infty}^{\infty} |h(n)| = \sum_{n=0}^{\infty} |a|^n + \sum_{n=-\infty}^{-1} |b|^n$$

From Example 2.3.6 we have already determined that the first sum converges for $|a| < 1$. The second sum can be manipulated as follows:

$$\sum_{n=-\infty}^{-1} |b|^n = \sum_{n=1}^{\infty} \frac{1}{|b|^n} = \frac{1}{|b|}\left(1 + \frac{1}{|b|} + \frac{1}{|b|^2} + \cdots\right)$$

$$= \beta(1 + \beta + \beta^2 + \cdots) = \frac{\beta}{1-\beta}$$

where $\beta = 1/|b|$ must be less than unity for the geometric series to converge. Consequently, the system is stable if both $|a| < 1$ and $|b| > 1$ are satisfied.

2.3.7 Systems with Finite-Duration and Infinite-Duration Impulse Response

Up to this point we have characterized a linear time-invariant system in terms of its impulse response $h(n)$. It is also convenient, however, to subdivide the class of linear time-invariant systems into two types, those that have a finite-duration impulse response (FIR) and those that have an infinite-duration impulse response (IIR). Thus an FIR system has an impulse response that is zero outside of some finite time interval. Without loss of generality, we focus our attention on causal FIR systems, so that

$$h(n) = 0, \qquad n < 0 \text{ and } n \geq M$$

The convolution formula for such a system reduces to

$$y(n) = \sum_{k=0}^{M-1} h(k)x(n-k)$$

A useful interpretation of this expression is obtained by observing that the output at any time n is simply a weighted linear combination of the input signal samples $x(n)$, $x(n-1), \ldots, x(n-M+1)$. In other words, the system simply weights, by the values of the impulse response $h(k)$, $k = 0, 1, \ldots, M-1$, the most recent M signal samples

and sums the resulting M products. In effect, the system acts as a *window* that views only the most recent M input signal samples in forming the output. It neglects or simply "forgets" all prior input samples [i.e., $x(n-M), x(n-M-1), \ldots$]. Thus we say that an FIR system has a finite memory of length-M samples.

In contrast, an IIR linear time-invariant system has an infinite-duration impulse response. Its output, based on the convolution formula, is

$$y(n) = \sum_{k=0}^{\infty} h(k)x(n-k)$$

where causality has been assumed, although this assumption is not necessary. Now, the system output is a weighted [by the impulse response $h(k)$] linear combination of the input signal samples $x(n)$, $x(n-1)$, $x(n-2)$, ... Since this weighted sum involves the present and all the past input samples, we say that the system has an infinite memory.

We investigate the characteristics of FIR and IIR systems in more detail in subsequent chapters.

2.4 Discrete-Time Systems Described by Difference Equations

Up to this point we have treated linear and time-invariant systems that are characterized by their unit sample response $h(n)$. In turn, $h(n)$ allows us to determine the output $y(n)$ of the system for any given input sequence $x(n)$ by means of the convolution summation,

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k) \tag{2.4.1}$$

In general, then, we have shown that any linear time-invariant system is characterized by the input–output relationship in (2.4.1). Moreover, the convolution summation formula in (2.4.1) suggests a means for the realization of the system. In the case of FIR systems, such a realization involves additions, multiplications, and a finite number of memory locations. Consequently, an FIR system is readily implemented directly, as implied by the convolution summation.

If the system is IIR, however, its practical implementation as implied by convolution is clearly impossible, since it requires an infinite number of memory locations, multiplications, and additions. A question that naturally arises, then, is whether or not it is possible to realize IIR systems other than in the form suggested by the convolution summation. Fortunately, the answer is yes, there is a practical and computationally efficient means for implementing a family of IIR systems, as will be demonstrated in this section. Within the general class of IIR systems, this family of discrete-time systems is more conveniently described by difference equations. This family or subclass of IIR systems is very useful in a variety of practical applications, including the implementation of digital filters, and the modeling of physical phenomena and physical systems.

2.4.1 Recursive and Nonrecursive Discrete-Time Systems

As indicated above, the convolution summation formula expresses the output of the linear time-invariant system explicitly and only in terms of the input signal. However, this need not be the case, as is shown here. There are many systems where it is either necessary or desirable to express the output of the system not only in terms of the present and past values of the input, but also in terms of the already available past output values. The following problem illustrates this point.

Suppose that we wish to compute the *cumulative average* of a signal $x(n)$ in the interval $0 \le k \le n$, defined as

$$y(n) = \frac{1}{n+1} \sum_{k=0}^{n} x(k), \qquad n = 0, 1, \ldots \tag{2.4.2}$$

As implied by (2.4.2), the computation of $y(n)$ requires the storage of all the input samples $x(k)$ for $0 \le k \le n$. Since n is increasing, our memory requirements grow linearly with time.

Our intuition suggests, however, that $y(n)$ can be computed more efficiently by utilizing the previous output value $y(n-1)$. Indeed, by a simple algebraic rearrangement of (2.4.2), we obtain

$$\begin{aligned}(n+1)y(n) &= \sum_{k=0}^{n-1} x(k) + x(n) \\ &= ny(n-1) + x(n)\end{aligned}$$

and hence

$$y(n) = \frac{n}{n+1} y(n-1) + \frac{1}{n+1} x(n) \tag{2.4.3}$$

Now, the cumulative average $y(n)$ can be computed recursively by multiplying the previous output value $y(n-1)$ by $n/(n+1)$, multiplying the present input $x(n)$ by $1/(n+1)$, and adding the two products. Thus the computation of $y(n)$ by means of (2.4.3) requires two multiplications, one addition, and one memory location, as illustrated in Fig. 2.4.1. This is an example of a *recursive system*. In general, a system whose output $y(n)$ at time n depends on any number of past output values $y(n-1)$, $y(n-2)$, ... is called a recursive system.

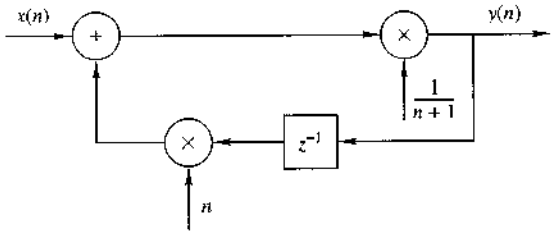


Figure 2.4.1
Realization of a recursive cumulative averaging system.

To determine the computation of the recursive system in (2.4.3) in more detail, suppose that we begin the process with $n = 0$ and proceed forward in time. Thus, according to (2.4.3), we obtain

$$y(0) = x(0)$$

$$y(1) = \frac{1}{2}y(0) + \frac{1}{2}x(1)$$

$$y(2) = \frac{2}{3}y(1) + \frac{1}{3}x(2)$$

and so on. If one grows fatigued with this computation and wishes to pass the problem to someone else at some time, say $n = n_0$, the only information that one needs to provide his or her successor is the past value $y(n_0 - 1)$ and the new input samples $x(n), x(n+1), \ldots$. Thus the successor begins with

$$y(n_0) = \frac{n_0}{n_0 + 1}y(n_0 - 1) + \frac{1}{n_0 + 1}x(n_0)$$

and proceeds forward in time until some time, say $n = n_1$, when he or she becomes fatigued and passes the computational burden to someone else with the information on the value $y(n_1 - 1)$, and so on.

The point we wish to make in this discussion is that if one wishes to compute the response (in this case, the cumulative average) of the system (2.4.3) to an input signal $x(n)$ applied at $n = n_0$, we need the value $y(n_0 - 1)$ and the input samples $x(n)$ for $n \geq n_0$. The term $y(n_0 - 1)$ is called the *initial condition* for the system in (2.4.3) and contains all the information needed to determine the response of the system for $n \geq n_0$ to the input signal $x(n)$, independent of what has occurred in the past.

The following example illustrates the use of a (nonlinear) recursive system to compute the square root of a number.

EXAMPLE 2.4.1 Square-Root Algorithm

Many computers and calculators compute the square root of a positive number A, using the iterative algorithm

$$s_n = \frac{1}{2}\left(s_{n-1} + \frac{A}{s_{n-1}}\right), \qquad n = 0, 1, \ldots$$

where s_{n-1} is an initial guess (estimate) of $\sqrt{A}$. As the iteration converges we have $s_n \approx s_{n-1}$. Then it easily follows that $s_n \approx \sqrt{A}$.

Consider now the recursive system

$$y(n) = \frac{1}{2}\left[y(n-1) + \frac{x(n)}{y(n-1)}\right] \tag{2.4.4}$$

which is realized as in Fig. 2.4.2. If we excite this system with a step of amplitude A [i.e., $x(n) = Au(n)$] and use as an initial condition $y(-1)$ an estimate of $\sqrt{A}$, the response $y(n)$ of the system will tend toward $\sqrt{A}$ as n increases. Note that in contrast to the system (2.4.3), we do not need to specify exactly the initial condition. A rough estimate is sufficient for the proper performance of the system. For example, if we let $A = 2$ and $y(-1) = 1$, we obtain $y(0) = \frac{3}{2}$, $y(1) = 1.4166667$, $y(2) = 1.4142157$. Similarly, for $y(-1) = 1.5$, we have $y(0) = 1.416667$, $y(1) = 1.4142157$. Compare these values with the $\sqrt{2}$, which is approximately 1.4142136.

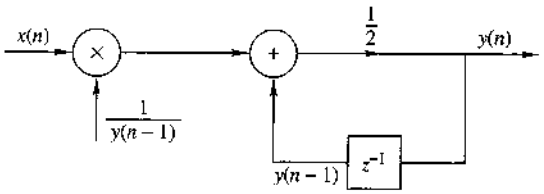


Figure 2.4.2
Realization of the square-root system

We have now introduced two simple recursive systems, where the output $y(n)$ depends on the previous output value $y(n-1)$ and the current input $x(n)$. Both systems are causal. In general, we can formulate more complex causal recursive systems, in which the output $y(n)$ is a function of several past output values and present and past inputs. The system should have a finite number of delays or, equivalently, should require a finite number of storage locations to be practically implemented. Thus the output of a causal and practically realizable recursive system can be expressed in general as

$$y(n) = F[y(n-1), y(n-2), \ldots, y(n-N), x(n), x(n-1), \ldots, x(n-M)] \quad (2.4.5)$$

where $F[\cdot]$ denotes some function of its arguments. This is a recursive equation specifying a procedure for computing the system output in terms of previous values of the output and present and past inputs.

In contrast, if $y(n)$ depends only on the present and past inputs, then

$$y(n) = F[x(n), x(n-1), \ldots, x(n-M)] \quad (2.4.6)$$

Such a system is called *nonrecursive*. We hasten to add that the causal FIR systems described in Section 2.3.7 in terms of the convolution sum formula have the form of (2.4.6). Indeed, the convolution summation for a causal FIR system is

$$\begin{aligned} y(n) &= \sum_{k=0}^{M} h(k)x(n-k) \\ &= h(0)x(n) + h(1)x(n-1) + \cdots + h(M)x(n-M) \\ &= F[x(n), x(n-1), \ldots, x(n-M)] \end{aligned}$$

where the function $F[\cdot]$ is simply a linear weighted sum of present and past inputs and the impulse response values $h(n)$, $0 \le n \le M$, constitute the weighting coefficients. Consequently, the causal linear time-invariant FIR systems described by the convolution formula in Section 2.3.7 are nonrecursive. The basic differences between nonrecursive and recursive systems are illustrated in Fig. 2.4.3. A simple inspection of this figure reveals that the fundamental difference between these two systems is the feedback loop in the recursive system, which feeds back the output of the system into the input. This feedback loop contains a delay element. The presence of this delay is crucial for the realizability of the system, since the absence of this delay would force the system to compute $y(n)$ in terms of $y(n)$, which is not possible for discrete-time systems.

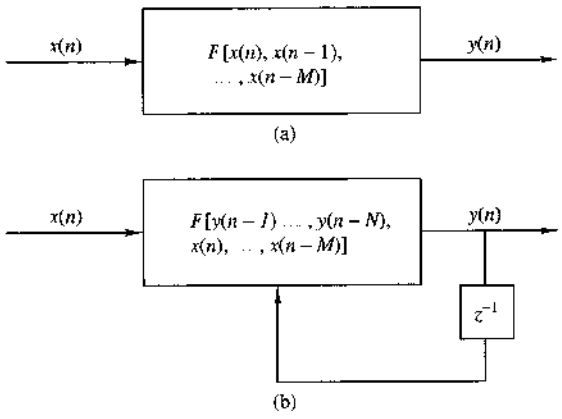


Figure 2.4.3
Basic form for a causal and realizable (a) nonrecursive and (b) recursive system.

The presence of the feedback loop or, equivalently, the recursive nature of (2.4.5) creates another important difference between recursive and nonrecursive systems. For example, suppose that we wish to compute the output $y(n_0)$ of a system when it is excited by an input applied at time $n = 0$. If the system is recursive, to compute $y(n_0)$, we first need to compute all the previous values $y(0), y(1), \ldots, y(n_0 - 1)$. In contrast, if the system is nonrecursive, we can compute the output $y(n_0)$ immediately without having $y(n_0 - 1), y(n_0 - 2), \ldots$. In conclusion, the output of a recursive system should be computed in order [i.e., $y(0), y(1), y(2), \ldots$], whereas for a nonrecursive system, the output can be computed in any order [i.e., $y(200), y(15), y(3), y(300)$, etc.]. This feature is desirable in some practical applications.

2.4.2 Linear Time-Invariant Systems Characterized by Constant-Coefficient Difference Equations

In Section 2.3 we treated linear time-invariant systems and characterized them in terms of their impulse responses. In this subsection we focus our attention on a family of linear time-invariant systems described by an input–output relation called a difference equation with constant coeffficients. Systems described by constant-coefficient linear difference equations are a subclass of the recursive and nonrecursive systems introduced in the preceding subsection. To bring out the important ideas, we begin by treating a simple recursive system described by a first-order difference equation.

Suppose that we have a recursive system with an input–output equation

$$y(n) = ay(n - 1) + x(n) \tag{2.4.7}$$

where a is a constant. Figure 2.4.4 shows a block diagram realization of the system. In comparing this system with the cumulative averaging system described by the input–output equation (2.4.3), we observe that the system in (2.4.7) has a constant coefficient (independent of time), whereas the system described in (2.4.3) has time-variant coefficients. As we will show, (2.4.7) is an input–output equation for a linear time-invariant system, whereas (2.4.3) describes a linear time-variant system.

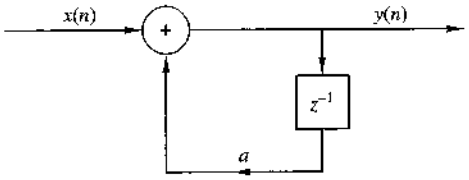


Figure 2.4.4
Block diagram realization of a simple recursive system.

Now, suppose that we apply an input signal $x(n)$ to the system for $n \geq 0$. We make no assumptions about the input signal for $n < 0$, but we do assume the existence of the initial condition $y(-1)$. Since (2.4.7) describes the system output implicitly, we must solve this equation to obtain an explicit expression for the system output. Suppose that we compute successive values of $y(n)$ for $n \geq 0$, beginning with $y(0)$. Thus

$$\begin{aligned}
y(0) &= ay(-1) + x(0) \\
y(1) &= ay(0) + x(1) = a^2 y(-1) + ax(0) + x(1) \\
y(2) &= ay(1) + x(2) = a^3 y(-1) + a^2 x(0) + ax(1) + x(2) \\
&\vdots \quad \vdots \\
y(n) &= ay(n-1) + x(n) \\
&= a^{n+1} y(-1) + a^n x(0) + a^{n-1} x(1) + \cdots + ax(n-1) + x(n)
\end{aligned}$$

or, more compactly,

$$y(n) = a^{n+1} y(-1) + \sum_{k=0}^{n} a^k x(n-k), \qquad n \geq 0 \tag{2.4.8}$$

The response $y(n)$ of the system as given by the right-hand side of (2.4.8) consists of two parts. The first part, which contains the term $y(-1)$, is a result of the initial condition $y(-1)$ of the system. The second part is the response of the system to the input signal $x(n)$.

If the system is initially relaxed at time $n = 0$, then its memory (i.e., the output of the delay) should be zero. Hence $y(-1) = 0$. Thus a recursive system is relaxed if it starts with zero initial conditions. Because the memory of the system describes, in some sense, its "state," we say that the system is at zero state and its corresponding output is called the *zero-state response*, and is denoted by $y_{zs}(n)$. Obviously, the zero-state response of the system (2.4.7) is given by

$$y_{zs}(n) = \sum_{k=0}^{n} a^k x(n-k), \qquad n \geq 0 \tag{2.4.9}$$

It is interesting to note that (2.4.9) is a convolution summation involving the input signal convolved with the impulse response

$$h(n) = a^n u(n) \tag{2.4.10}$$

We also observe that the system described by the first-order difference equation in (2.4.7) is causal. As a result, the lower limit on the convolution summation in (2.4.9) is $k = 0$. Furthermore, the condition $y(-1) = 0$ implies that the input signal can be assumed causal and hence the upper limit on the convolution summation in (2.4.9) is n, since $x(n-k) = 0$ for $k > n$. In effect, we have obtained the result that the relaxed recursive system described by the first-order difference equation in (2.4.7) is a linear time-invariant IIR system with impulse response given by (2.4.10).

Now, suppose that the system described by (2.4.7) is initially nonrelaxed [i.e., $y(-1) \neq 0$] and the input $x(n) = 0$ for all n. Then the output of the system with zero input is called the *zero-input response* or *natural response* and is denoted by $y_{zi}(n)$. From (2.4.7), with $x(n) = 0$ for $-\infty < n < \infty$, we obtain

$$y_{zi}(n) = a^{n+1}y(-1), \qquad n \geq 0 \tag{2.4.11}$$

We observe that a recursive system with nonzero initial condition is nonrelaxed in the sense that it can produce an output without being excited. Note that the zero-input response is due to the memory of the system.

To summarize, the zero-input response is obtained by setting the input signal to zero, making it independent of the input. It depends only on the nature of the system and the initial condition. Thus the zero-input response is a characteristic of the system itself, and it is also known as the *natural* or *free response* of the system. On the other hand, the zero-state response depends on the nature of the system and the input signal. Since this output is a response forced upon it by the input signal, it is usually called the *forced response* of the system. In general, the total response of the system can be expressed as $y(n) = y_{zi}(n) + y_{zs}(n)$.

The system described by the first-order difference equation in (2.4.7) is the simplest possible recursive system in the general class of recursive systems described by linear constant-coefficient difference equations. The general form for such an equation is

$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k) \tag{2.4.12}$$

or, equivalently,

$$\sum_{k=0}^{N} a_k y(n-k) = \sum_{k=0}^{M} b_k x(n-k), \qquad a_0 \equiv 1 \tag{2.4.13}$$

The integer N is called the *order* of the difference equation or the order of the system. The negative sign on the right-hand side of (2.4.12) is introduced as a matter of convenience to allow us to express the difference equation in (2.4.13) without any negative signs.

Equation (2.4.12) expresses the output of the system at time n directly as a weighted sum of past outputs $y(n-1), y(n-2), \ldots, y(n-N)$ as well as past and present input signals samples. We observe that in order to determine $y(n)$ for $n \geq 0$, we need the input $x(n)$ for all $n \geq 0$, and the initial conditions $y(-1)$,

$y(-2), \ldots, y(-N)$. In other words, the initial conditions summarize all that we need to know about the past history of the response of the system to compute the present and future outputs. The general solution of the N-order constant-coefficient difference equation is considered in the following subsection.

At this point we restate the properties of linearity, time invariance, and stability in the context of recursive systems described by linear constant-coefficient difference equations. As we have observed, a recursive system may be relaxed or nonrelaxed, depending on the initial conditions. Hence the definitions of these properties must take into account the presence of the initial conditions.

We begin with the definition of linearity. A system is linear if it satisfies the following three requirements:

1. The total response is equal to the sum of the zero-input and zero-state responses [i.e., $y(n) = y_{zi}(n) + y_{zs}(n)$].
2. The principle of superposition applies to the zero-state response (*zero-state linear*).
3. The principle of superposition applies to the zero-input response (*zero-input linear*).

A system that does not satisfy *all three* separate requirements is by definition nonlinear. Obviously, for a relaxed system, $y_{zi}(n) = 0$, and thus requirement 2, which is the definition of linearity given in Section 2.2.4, is sufficient.

We illustrate the application of these requirements by a simple example.

EXAMPLE 2.4.2

Determine if the recursive system defined by the difference equation

$$y(n) = ay(n-1) + x(n)$$

is linear.

Solution. By combining (2.4.9) and (2.4.11), we obtain (2.4.8), which can be expressed as

$$y(n) = y_{zi}(n) + y_{zs}(n)$$

Thus the first requirement for linearity is satisfied.

To check for the second requirement, let us assume that $x(n) = c_1x_1(n) + c_2x_2(n)$. Then (2.4.9) gives

$$\begin{aligned} y_{zs}(n) &= \sum_{k=0}^{n} a^k [c_1x_1(n-k) + c_2x_2(n-k)] \\ &= c_1 \sum_{k=0}^{n} a^k x_1(n-k) + c_2 \sum_{k=0}^{n} a^k x_2(n-k) \\ &= c_1 y_{zs}^{(1)}(n) + c_2 y_{zs}^{(2)}(n) \end{aligned}$$

Hence $y_{zs}(n)$ satisfies the principle of superposition, and thus the system is zero-state linear.

Now let us assume that $y(-1) = c_1 y_1(-1) + c_2 y_2(-1)$. From (2.4.11) we obtain

$$\begin{aligned} y_{zi}(n) &= a^{n+1}[c_1 y_1(-1) + c_2 y_2(-1)] \\ &= c_1 a^{n+1} y_1(-1) + c_2 a^{n+1} y_2(-1) \\ &= c_1 y_{zi}^{(1)}(n) + c_2 y_{zi}^{(2)}(n) \end{aligned}$$

Hence the system is zero-input linear.

Since the system satisfies all three conditions for linearity, it is linear.

Although it is somewhat tedious, the procedure used in Example 2.4.2 to demonstrate linearity for the system described by the first-order difference equation carries over directly to the general recursive systems described by the constant-coefficient difference equation given in (2.4.13). Hence, a recursive system described by the linear difference equation in (2.4.13) also satisfies all three conditions in the definition of linearity, and therefore it is linear.

The next question that arises is whether or not the causal linear system described by the linear constant-coefficient difference equation in (2.4.13) is time invariant. This is fairly easy, when dealing with systems described by explicit input–output mathematical relationships. Clearly, the system described by (2.4.13) is time invariant because the coefficients a_k and b_k are constants. On the other hand, if one or more of these coefficients depends on time, the system is time variant, since its properties change as a function of time. Thus we conclude that *the recursive system described by a linear constant-coefficient difference equation is linear and time invariant.*

The final issue is the stability of the recursive system described by the linear, constant-coefficient difference equation in (2.4.13). In Section 2.3.6 we introduced the concept of bounded input–bounded output (BIBO) stability for relaxed systems. For nonrelaxed systems that may be nonlinear, BIBO stability should be viewed with some care. However, in the case of a linear time-invariant recursive system described by the linear constant-coefficient difference equation in (2.4.13), it suffices to state that such a system is BIBO stable if and only if for every bounded input and every bounded initial condition, the total system response is bounded.

EXAMPLE 2.4.3

Determine if the linear time-invariant recursive system described by the difference equation given in (2.4.7) is stable.

Solution. Let us assume that the input signal $x(n)$ is bounded in amplitude, that is, $|x(n)| \le M_x < \infty$ for all $n \ge 0$. From (2.4.8) we have

$$\begin{aligned} |y(n)| &\le |a^{n+1} y(-1)| + \left| \sum_{k=0}^{n} a^k x(n-k) \right|, && n \ge 0 \\ &\le |a|^{n+1} |y(-1)| + M_x \sum_{k=0}^{n} |a|^k, && n \ge 0 \\ &\le |a|^{n+1} |y(-1)| + M_x \frac{1 - |a|^{n+1}}{1 - |a|} = M_y, && n \ge 0 \end{aligned}$$

If n is finite, the bound M_y is finite and the output is bounded independently of the value of a. However, as $n \to \infty$, the bound M_y remains finite only if $|a| < 1$ because $|a|^n \to 0$ as $n \to \infty$. Then $M_y = M_x/(1 - |a|)$.

Thus the system is stable only if $|a| < 1$.

For the simple first-order system in Example 2.4.3, we were able to express the condition for BIBO stability in terms of the system parameter a, namely $|a| < 1$. We should stress, however, that this task becomes more difficult for higher-order systems. Fortunately, as we shall see in subsequent chapters, other simple and more efficient techniques exist for investigating the stability of recursive systems.

2.4.3 Solution of Linear Constant-Coefficient Difference Equations

Given a linear constant-coefficient difference equation as the input–output relationship describing a linear time-invariant system, our objective in this subsection is to determine an explicit expression for the output $y(n)$. The method that is developed is termed the *direct method*. An alternative method based on the z-transform is described in Chapter 3. For reasons that will become apparent later, the z-transform approach is called the *indirect method*.

Basically, the goal is to determine the output $y(n)$, $n \geq 0$, of the system given a specific input $x(n)$, $n \geq 0$, and a set of initial conditions. The direct solution method assumes that the total solution is the sum of two parts:

$$y(n) = y_h(n) + y_p(n)$$

The part $y_h(n)$ is known as the *homogeneous* or *complementary* solution, whereas $y_p(n)$ is called the *particular* solution.

The homogeneous solution of a difference equation. We begin the problem of solving the linear constant-coefficient difference equation given by (2.4.13) by obtaining first the solution to the *homogeneous difference equation*

$$\sum_{k=0}^{N} a_k y(n-k) = 0 \tag{2.4.14}$$

The procedure for solving a linear constant-coefficient difference equation directly is very similar to the procedure for solving a linear constant-coefficient differential equation. Basically, we assume that the solution is in the form of an exponential, that is,

$$y_h(n) = \lambda^n \tag{2.4.15}$$

where the subscript h on $y(n)$ is used to denote the solution to the homogeneous difference equation. If we substitute this assumed solution in (2.4.14), we obtain the polynomial equation

$$\sum_{k=0}^{N} a_k \lambda^{n-k} = 0$$

or

$$\lambda^{n-N}(\lambda^N + a_1\lambda^{N-1} + a_2\lambda^{N-2} + \cdots + a_{N-1}\lambda + a_N) = 0 \tag{2.4.16}$$

The polynomial in parentheses is called the *characteristic polynomial* of the system. In general, it has N roots, which we denote as $\lambda_1, \lambda_2, \ldots, \lambda_N$. The roots can be real or complex valued. In practice the coefficients $a_1, a_2, \ldots, a_N$ are usually real. Complex-valued roots occur as complex-conjugate pairs. Some of the N roots may be identical, in which case we have multiple-order roots.

For the moment, let us assume that the roots are distinct, that is, there are no multiple-order roots. Then the most general solution to the homogeneous difference equation in (2.4.14) is

$$y_h(n) = C_1\lambda_1^n + C_2\lambda_2^n + \cdots + C_N\lambda_N^n \tag{2.4.17}$$

where $C_1, C_2, \ldots, C_N$ are weighting coefficients.

These coefficients are determined from the initial conditions specified for the system. Since the input $x(n) = 0$, (2.4.17) can be used to obtain the *zero-input response* of the system. The following examples illustrate the procedure.

EXAMPLE 2.4.4

Determine the homogeneous solution of the system described by the first-order difference equation

$$y(n) + a_1y(n-1) = x(n) \tag{2.4.18}$$

Solution. The assumed solution obtained by setting $x(n) = 0$ is

$$y_h(n) = \lambda^n$$

When we substitute this solution in (2.4.18), we obtain [with $x(n) = 0$]

$$\lambda^n + a_1\lambda^{n-1} = 0$$

$$\lambda^{n-1}(\lambda + a_1) = 0$$

$$\lambda = -a_1$$

Therefore, the solution to the homogeneous difference equation is

$$y_h(n) = C\lambda^n = C(-a_1)^n \tag{2.4.19}$$

The zero-input response of the system can be determined from (2.4.18) and (2.4.19). With $x(n) = 0$, (2.4.18) yields

$$y(0) = -a_1y(-1)$$

On the other hand, from (2.4.19) we have

$$y_h(0) = C$$

and hence the zero-input response of the system is

$$y_{zi}(n) = (-a_1)^{n+1}y(-1), \qquad n \geq 0 \tag{2.4.20}$$

With $a = -a_1$, this result is consistent with (2.4.11) for the first-order system, which was obtained earlier by iteration of the difference equation.

EXAMPLE 2.4.5

Determine the zero-input response of the system described by the homogeneous second-order difference equation

$$y(n) - 3y(n-1) - 4y(n-2) = 0 \tag{2.4.21}$$

Solution. First we determine the solution to the homogeneous equation. We assume the solution to be the exponential

$$y_h(n) = \lambda^n$$

Upon substitution of this solution into (2.4.21), we obtain the characteristic equation

$$\lambda^n - 3\lambda^{n-1} - 4\lambda^{n-2} = 0$$

$$\lambda^{n-2}(\lambda^2 - 3\lambda - 4) = 0$$

Therefore, the roots are $\lambda = -1, 4$, and the general form of the solution to the homogeneous equation is

$$\begin{aligned} y_h(n) &= C_1\lambda_1^n + C_2\lambda_2^n \\ &= C_1(-1)^n + C_2(4)^n \end{aligned} \tag{2.4.22}$$

The zero-input response of the system can be obtained from the homogenous solution by evaluating the constants in (2.4.22), given the initial conditions $y(-1)$ and $y(-2)$. From the difference equation in (2.4.21) we have

$$y(0) = 3y(-1) + 4y(-2)$$

$$\begin{aligned} y(1) &= 3y(0) + 4y(-1) \\ &= 3[3y(-1) + 4y(-2)] + 4y(-1) \\ &= 13y(-1) + 12y(-2) \end{aligned}$$

On the other hand, from (2.4.22) we obtain

$$y(0) = C_1 + C_2$$

$$y(1) = -C_1 + 4C_2$$

By equating these two sets of relations, we have

$$C_1 + C_2 = 3y(-1) + 4y(-2)$$

$$-C_1 + 4C_2 = 13y(-1) + 12y(-2)$$

The solution of these two equations is

$$C_1 = -\frac{1}{5}y(-1) + \frac{4}{5}y(-2)$$

$$C_2 = \frac{16}{5}y(-1) + \frac{16}{5}y(-2)$$

Therefore, the zero-input response of the system is

$$\begin{aligned} y_{zi}(n) &= [-\frac{1}{5}y(-1) + \frac{4}{5}y(-2)](-1)^n \\ &\quad + [\frac{16}{5}y(-1) + \frac{16}{5}y(-2)](4)^n, \qquad n \geq 0 \end{aligned} \tag{2.4.23}$$

For example, if $y(-2) = 0$ and $y(-1) = 5$, then $C_1 = -1$, $C_2 = 16$, and hence

$$y_{zi}(n) = (-1)^{n+1} + (4)^{n+2}, \qquad n \geq 0$$

These examples illustrate the method for obtaining the homogeneous solution and the zero-input response of the system when the characteristic equation contains distinct roots. On the other hand, if the characteristic equation contains multiple roots, the form of the solution given in (2.4.17) must be modified. For example, if λ_1 is a root of multiplicity m, then (2.4.17) becomes

$$\begin{aligned} y_h(n) &= C_1\lambda_1^n + C_2 n\lambda_1^n + C_3 n^2\lambda_1^n + \cdots + C_m n^{m-1}\lambda_1^n \\ &\quad + C_{m+1}\lambda_{m+1}^n + \cdots + C_N\lambda_N^n \end{aligned} \tag{2.4.24}$$

The particular solution of the difference equation. The particular solution $y_p(n)$ is required to satisfy the difference equation (2.4.13) for the specific input signal $x(n)$, $n \geq 0$. In other words, $y_p(n)$ is any solution satisfying

$$\sum_{k=0}^{N} a_k y_p(n-k) = \sum_{k=0}^{M} b_k x(n-k), \qquad a_0 = 1 \tag{2.4.25}$$

To solve (2.4.25), we assume for $y_p(n)$, a form that depends on the form of the input $x(n)$. The following example illustrates the procedure.

EXAMPLE 2.4.6

Determine the particular solution of the first-order difference equation

$$y(n) + a_1 y(n-1) = x(n), \qquad |a_1| < 1 \tag{2.4.26}$$

when the input $x(n)$ is a unit step sequence, that is,

$$x(n) = u(n)$$

Solution. Since the input sequence $x(n)$ is a constant for $n \geq 0$, the form of the solution that we assume is also a constant. Hence the assumed solution of the difference equation to the forcing function $x(n)$, called the *particular solution* of the difference equation, is

$$y_p(n) = Ku(n)$$

where K is a scale factor determined so that (2.4.26) is satisfied. Upon substitution of this assumed solution into (2.4.26), we obtain

$$Ku(n) + a_1 Ku(n-1) = u(n)$$

To determine K, we must evaluate this equation for any $n \geq 1$, where none of the terms vanish. Thus

$$K + a_1 K = 1$$

$$K = \frac{1}{1+a_1}$$

Therefore, the particular solution to the difference equation is

$$y_p(n) = \frac{1}{1+a_1} u(n) \tag{2.4.27}$$

In this example, the input $x(n)$, $n \geq 0$, is a constant and the form assumed for the particular solution is also a constant. If $x(n)$ is an exponential, we would assume that the particular solution is also an exponential. If $x(n)$ were a sinusoid, then $y_p(n)$ would also be a sinusoid. Thus our assumed form for the particular solution takes the basic form of the signal $x(n)$. Table 2.1 provides the general form of the particular solution for several types of excitation.

EXAMPLE 2.4.7

Determine the particular solution of the difference equation

$$y(n) = \frac{5}{6} y(n-1) - \frac{1}{6} y(n-2) + x(n)$$

when the forcing function $x(n) = 2^n$, $n \geq 0$ and zero elsewhere.

TABLE 2.1 General Form of the Particular Solution for Several Types of Input Signals

Input Signal, $x(n)$	Particular Solution, $y_p(n)$
A (constant)	K
AM^n	KM^n
An^M	$K_0 n^M + K_1 n^{M-1} + \cdots + K_M$
$A^n n^M$	$A^n(K_0 n^M + K_1 n^{M-1} + \cdots + K_M)$
$\left\{ \begin{matrix} A\cos\omega_0 n \\ A\sin\omega_0 n \end{matrix} \right\}$	$K_1 \cos\omega_0 n + K_2 \sin\omega_0 n$

Solution. The form of the particular solution is

$$y_p(n) = K2^n, \qquad n \geq 0$$

Upon substitution of $y_p(n)$ into the difference equation, we obtain

$$K2^n u(n) = \frac{5}{6}K2^{n-1}u(n-1) - \frac{1}{6}K2^{n-2}u(n-2) + 2^n u(n)$$

To determine the value of K, we can evaluate this equation for any $n \geq 2$, where none of the terms vanish. Thus we obtain

$$4K = \frac{5}{6}(2K) - \frac{1}{6}K + 4$$

and hence $K = \frac{8}{5}$. Therefore, the particular solution is

$$y_p(n) = \frac{8}{5}2^n, \qquad n \geq 0$$

We have now demonstrated how to determine the two components of the solution to a difference equation with constant coefficients. These two components are the homogeneous solution and the particular solution. From these two components, we construct the total solution from which we can obtain the zero-state response.

The total solution of the difference equation. The linearity property of the linear constant-coefficient difference equation allows us to add the homogeneous solution and the particular solution in order to obtain the *total solution*. Thus

$$y(n) = y_h(n) + y_p(n)$$

The resultant sum $y(n)$ contains the constant parameters $\{C_i\}$ embodied in the homogeneous solution component $y_h(n)$. These constants can be determined to satisfy the initial conditions. The following example illustrates the procedure.

EXAMPLE 2.4.8

Determine the total solution $y(n)$, $n \geq 0$, to the difference equation

$$y(n) + a_1 y(n-1) = x(n) \tag{2.4.28}$$

when $x(n)$ is a unit step sequence [i.e., $x(n) = u(n)$] and $y(-1)$ is the initial condition.

Solution. From (2.4.19) of Example 2.4.4, the homogeneous solution is

$$y_h(n) = C(-a_1)^n$$

and from (2.4.26) of Example 2.4.6, the particular solution is

$$y_p(n) = \frac{1}{1+a_1}u(n)$$

Consequently, the total solution is

$$y(n) = C(-a_1)^n + \frac{1}{1+a_1}, \qquad n \geq 0 \tag{2.4.29}$$

where the constant C is determined to satisfy the initial condition $y(-1)$.

In particular, suppose that we wish to obtain the zero-state response of the system described by the first-order difference equation in (2.4.28). Then we set $y(-1) = 0$. To evaluate C, we evaluate (2.4.28) at $n = 0$, obtaining

$$y(0) + a_1 y(-1) = 1$$

Hence,

$$y(0) = 1 - a_1 y(-1)$$

On the other hand, (2.4.29) evaluated at $n = 0$ yields

$$y(0) = C + \frac{1}{1+a_1}$$

By equating these two relations, we obtain

$$C + \frac{1}{1+a_1} = -a_1 y(-1) + 1$$

$$C = -a_1 y(-1) + \frac{a_1}{1+a_1}$$

Finally, if we substitute this value of C into (2.4.29), we obtain

$$\begin{aligned} y(n) &= (-a_1)^{n+1} y(-1) + \frac{1-(-a_1)^{n+1}}{1+a_1}, \qquad n \geq 0 \\ &= y_{zi}(n) + y_{zs}(n) \end{aligned} \tag{2.4.30}$$

We observe that the system response as given by (2.4.30) is consistent with the response $y(n)$ given in (2.4.8) for the first-order system (with $a = -a_1$), which was obtained by solving the difference equation iteratively. Furthermore, we note that the value of the constant C depends both on the initial condition $y(-1)$ and on the excitation function. Consequently, the value of C influences both the zero-input response and the zero-state response.

We further observe that the particular solution to the difference equation can be obtained from the zero-state response of the system. Indeed, if $|a_1| < 1$, which is the condition for stability of the system, as will be shown in Section 2.4.4, the limiting value of $y_{zs}(n)$ as n approaches infinity is the particular solution, that is,

$$y_p(n) = \lim_{n\to\infty} y_{zs}(n) = \frac{1}{1+a_1}$$

Since this component of the system response does not go to zero as n approaches infinity, it is usually called the *steady-state response* of the system. This response persists as long as the input persists. The component that dies out as n approaches infinity is called the *transient response* of the system.

The following example illustrates the evaluation of the total solution for a second-order recursive system.

EXAMPLE 2.4.9

Determine the response $y(n)$, $n \geq 0$, of the system described by the second-order difference equation

$$y(n) - 3y(n-1) - 4y(n-2) = x(n) + 2x(n-1) \tag{2.4.31}$$

when the input sequence is

$$x(n) = 4^n u(n)$$

Solution. We have already determined the solution to the homogeneous difference equation for this system in Example 2.4.5. From (2.4.22) we have

$$y_h(n) = C_1(-1)^n + C_2(4)^n \tag{2.4.32}$$

The particular solution to (2.4.31) is assumed to be an exponential sequence of the same form as $x(n)$. Normally, we could assume a solution of the form

$$y_p(n) = K(4)^n u(n)$$

However, we observe that $y_p(n)$ is already contained in the homogeneous solution, so that this particular solution is redundant. Instead, we select the particular solution to be linearly independent of the terms contained in the homogeneous solution. In fact, we treat this situation in the same manner as we have already treated multiple roots in the characteristic equation. Thus we assume that

$$y_p(n) = Kn(4)^n u(n) \tag{2.4.33}$$

Upon substitution of (2.4.33) into (2.4.31), we obtain

$$Kn(4)^n u(n) - 3K(n-1)(4)^{n-1}u(n-1) - 4K(n-2)(4)^{n-2}u(n-2) = (4)^n u(n) + 2(4)^{n-1}u(n-1)$$

To determine K, we evaluate this equation for any $n \geq 2$, where none of the unit step terms vanish. To simplify the arithmetic, we select $n = 2$, from which we obtain $K = \frac{6}{5}$. Therefore,

$$y_p(n) = \frac{6}{5}n(4)^n u(n) \tag{2.4.34}$$

The total solution to the difference equation is obtained by adding (2.4.32) to (2.4.34). Thus

$$y(n) = C_1(-1)^n + C_2(4)^n + \frac{6}{5}n(4)^n, \qquad n \geq 0 \tag{2.4.35}$$

where the constants C_1 and C_2 are determined such that the initial conditions are satisfied. To accomplish this, we return to (2.4.31), from which we obtain

$$y(0) = 3y(-1) + 4y(-2) + 1$$

$$y(1) = 3y(0) + 4y(-1) + 6$$

$$= 13y(-1) + 12y(-2) + 9$$

On the other hand, (2.4.35) evaluated at $n = 0$ and $n = 1$ yields

$$y(0) = C_1 + C_2$$

$$y(1) = -C_1 + 4C_2 + \frac{24}{5}$$

We can now equate these two sets of relations to obtain C_1 and C_2. In so doing, we have the response due to initial conditions $y(-1)$ and $y(-2)$ (the zero-input response), and the zero-state response.

Since we have already solved for the zero-input response in Example 2.4.5, we can simplify the computations above by setting $y(-1) = y(-2) = 0$. Then we have

$$C_1 + C_2 = 1$$

$$-C_1 + 4C_2 + \frac{24}{5} = 9$$

Hence $C_1 = -\frac{1}{25}$ and $C_2 = \frac{26}{25}$. Finally, we have the zero-state response to the forcing function $x(n) = (4)^n u(n)$ in the form

$$y_{zs}(n) = -\frac{1}{25}(-1)^n + \frac{26}{25}(4)^n + \frac{6}{5}n(4)^n, \qquad n \geq 0 \tag{2.4.36}$$

The total response of the system, which includes the response to arbitrary initial conditions, is the sum of (2.4.23) and (2.4.36).

2.4.4 The Impulse Response of a Linear Time-Invariant Recursive System

The impulse response of a linear time-invariant system was previously defined as the response of the system to a unit sample excitation [i.e., $x(n) = \delta(n)$]. In the case of a recursive system, $h(n)$ is simply equal to the zero-state response of the system when the input $x(n) = \delta(n)$ and the system is initially relaxed.

For example, in the simple first-order recursive system given in (2.4.7), the zero-state response given in (2.4.8), is

$$y_{zs}(n) = \sum_{k=0}^{n} a^k x(n-k) \tag{2.4.37}$$

When $x(n) = \delta(n)$ is substituted into (2.4.37), we obtain

$$y_{zs}(n) = \sum_{k=0}^{n} a^k \delta(n-k)$$

$$= a^n, \qquad n \geq 0$$

Hence the impulse response of the first-order recursive system described by (2.4.7) is

$$h(n) = a^n u(n) \tag{2.4.38}$$

as indicated in Section 2.4.2.

In the general case of an arbitrary, linear time-invariant recursive system, the zero-state response expressed in terms of the convolution summation is

$$y_{zs}(n) = \sum_{k=0}^{n} h(k)x(n-k), \qquad n \geq 0 \tag{2.4.39}$$

When the input is an impulse [i.e., $x(n) = \delta(n)$], (2.4.39) reduces to

$$y_{zs}(n) = h(n) \tag{2.4.40}$$

Now, let us consider the problem of determining the impulse response $h(n)$ given a linear constant-coefficient difference equation description of the system. In terms of our discussion in the preceding subsection, we have established the fact that the total response of the system to any excitation function consists of the sum of two solutions of the difference equation: the solution to the homogeneous equation plus the particular solution to the excitation function. In the case where the excitation is an impulse, the particular solution is zero, since $x(n) = 0$ for $n > 0$, that is,

$$y_p(n) = 0$$

Consequently, the response of the system to an impulse consists only of the solution to the homogeneous equation, with the $\{C_k\}$ parameters evaluated to satisfy the initial conditions dictated by the impulse. The following example illustrates the procedure for obtaining $h(n)$ given the difference equation for the system.

EXAMPLE 2.4.10

Determine the impulse response $h(n)$ for the system described by the second-order difference equation

$$y(n) - 3y(n-1) - 4y(n-2) = x(n) + 2x(n-1) \tag{2.4.41}$$

Solution. We have already determined in Example 2.4.5 that the solution to the homogeneous difference equation for this system is

$$y_h(n) = C_1(-1)^n + C_2(4)^n, \qquad n \geq 0 \tag{2.4.42}$$

Since the particular solution is zero when $x(n) = \delta(n)$, the impulse response of the system is simply given by (2.4.42), where C_1 and C_2 must be evaluated to satisfy (2.4.41).

For $n = 0$ and $n = 1$, (2.4.41) yields

$$y(0) = 1$$

$$y(1) = 3y(0) + 2 = 5$$

where we have imposed the conditions $y(-1) = y(-2) = 0$, since the system must be relaxed. On the other hand, (2.4.42) evaluated at $n = 0$ and $n = 1$ yields

$$y(0) = C_1 + C_2$$

$$y(1) = -C_1 + 4C_2$$

By solving these two sets of equations for C_1 and C_2, we obtain

$$C_1 = -\frac{1}{5}, \qquad C_2 = \frac{6}{5}$$

Therefore, the impulse response of the system is

$$h(n) = \left[-\frac{1}{5}(-1)^n + \frac{6}{5}(4)^n\right]u(n)$$

When the system is described by an Nth-order linear difference equation of the type given in (2.4.13), the solution of the homogeneous equation is

$$y_h(n) = \sum_{k=1}^{N} C_k \lambda^n k$$

when the roots $\{\lambda_k\}$ of the characteristic polynomial are distinct. Hence the impulse response of the system is identical in form, that is,

$$h(n) = \sum_{k=1}^{N} C_k \lambda_k^n \tag{2.4.43}$$

where the parameters $\{C_k\}$ are determined by setting the initial conditions $y(-1) = \cdots = y(-N) = 0$.

This form of $h(n)$ allows us to easily relate the stability of a system, described by an Nth-order difference equation, to the values of the roots of the characteristic polynomial. Indeed, since BIBO stability requires that the impulse response be absolutely summable, then, for a causal system, we have

$$\sum_{n=0}^{\infty} |h(n)| = \sum_{n=0}^{\infty} \left| \sum_{k=1}^{N} C_k \lambda^n k \right| \leq \sum_{k=1}^{N} |C_k| \sum_{n=0}^{\infty} |\lambda_k|^n$$

Now if $|\lambda_k| < 1$ for all k, then

$$\sum_{n=0}^{\infty} |\lambda_k|^n < \infty$$

and hence

$$\sum_{n=0}^{\infty} |h(n)| < \infty$$

On the other hand, if one or more of the $|\lambda_k| \geq 1$, $h(n)$ is no longer absolutely summable, and consequently, the system is unstable. Therefore, a necessary and sufficient condition for the stability of a causal IIR system described by a linear constant-coefficient difference equation is that all roots of the characteristic polynomial be less than unity in magnitude. The reader may verify that this condition carries over to the case where the system has roots of multiplicity m.

Finally we note that any recursive system described by a linear constant-coefficient difference equation is an IIR system. The converse is not true, however. That is, not every linear time-invariant IIR system can be described by a linear constant-coefficient difference equation. In other words, recursive systems described by linear constant-coefficient difference equations are a subclass of linear time-invariant IIR systems.

2.5 Implementation of Discrete-Time Systems

Our treatment of discrete-time systems has been focused on the time-domain characterization and analysis of linear time-invariant systems described by constant-coefficient linear difference equations. Additional analytical methods are developed in the next two chapters, where we characterize and analyze LTI systems in the frequency domain. Two other important topics that will be treated later are the design and implementation of these systems.

In practice, system design and implementation are usually treated jointly rather than separately. Often, the system design is driven by the method of implementation and by implementation constraints, such as cost, hardware limitations, size limitations, and power requirements. At this point, we have not as yet developed the necessary analysis and design tools to treat such complex issues. However, we have developed sufficient background to consider some basic implementation methods for realizations of LTI systems described by linear constant-coefficient difference equations.

2.5.1 Structures for the Realization of Linear Time-Invariant Systems

In this subsection we describe structures for the realization of systems described by linear constant-coefficient difference equations. Additional structures for these systems are introduced in Chapter 9.

As a beginning, let us consider the first-order system

$$y(n) = -a_1 y(n-1) + b_0 x(n) + b_1 x(n-1) \tag{2.5.1}$$

which is realized as in Fig. 2.5.1(a). This realization uses separate delays (memory) for both the input and output signal samples and it is called a *direct form I structure*. Note that this system can be viewed as two linear time-invariant systems in cascade. The first is a nonrecursive system described by the equation

$$v(n) = b_0 x(n) + b_1 x(n-1) \tag{2.5.2}$$

whereas the second is a recursive system described by the equation

$$y(n) = -a_1 y(n-1) + v(n) \tag{2.5.3}$$

However, as we have seen in Section 2.3.4, if we interchange the order of the cascaded linear time-invariant systems, the overall system response remains the same. Thus if we interchange the order of the recursive and nonrecursive systems, we obtain an alternative structure for the realization of the system described by (2.5.1). The resulting system is shown in Fig. 2.5.1(b). From this figure we obtain the two difference equations

$$w(n) = -a_1 w(n-1) + x(n) \tag{2.5.4}$$

$$y(n) = b_0 w(n) + b_1 w(n-1) \tag{2.5.5}$$

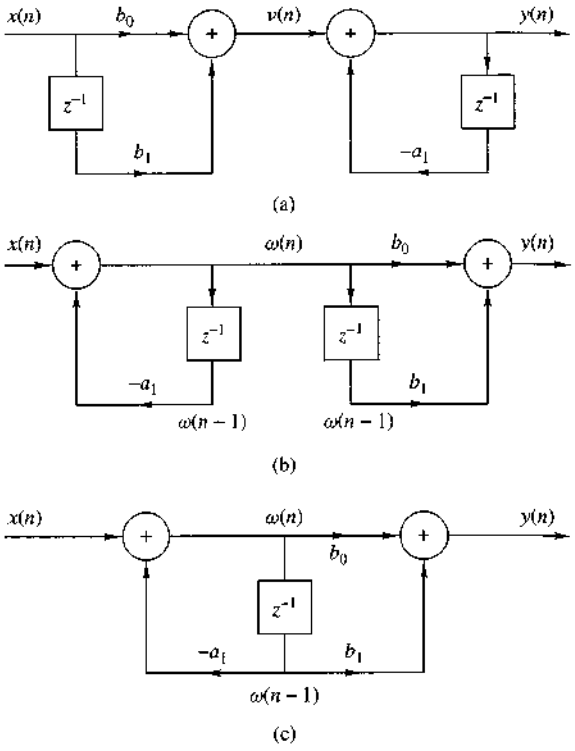


Figure 2.5.1
Steps in converting from the direct form I realization in (a) to the direct form II realization in (c)

which provide an alternative algorithm for computing the output of the system described by the single difference equation given in (2.5.1). In other words, the two difference equations (2.5.4) and (2.5.5) are equivalent to the single difference equation (2.5.1).

A close observation of Fig. 2.5.1 reveals that the two delay elements contain the same input $w(n)$ and hence the same output $w(n-1)$. Consequently, these two elements can be merged into one delay, as shown in Fig. 2.5.1(c). In contrast to the direct form I structure, this new realization requires only one delay for the auxiliary quantity $w(n)$, and hence it is more efficient in terms of memory requirements. It is called the *direct form II structure* and it is used extensively in practical applications.

These structures can readily be generalized for the general linear time-invariant recursive system described by the difference equation

$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k) \tag{2.5.6}$$

Figure 2.5.2 illustrates the direct form I structure for this system. This structure requires $M + N$ delays and $N + M + 1$ multiplications. It can be viewed as the

cascade of a nonrecursive system

$$v(n) = \sum_{k=0}^{M} b_k x(n-k) \tag{2.5.7}$$

and a recursive system

$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + v(n) \tag{2.5.8}$$

By reversing the order of these two systems, as was previously done for the first-order system, we obtain the direct form II structure shown in Fig. 2.5.3 for $N > M$. This structure is the cascade of a recursive system

$$w(n) = -\sum_{k=1}^{N} a_k w(n-k) + x(n) \tag{2.5.9}$$

followed by a nonrecursive system

$$y(n) = \sum_{k=0}^{M} b_k w(n-k) \tag{2.5.10}$$

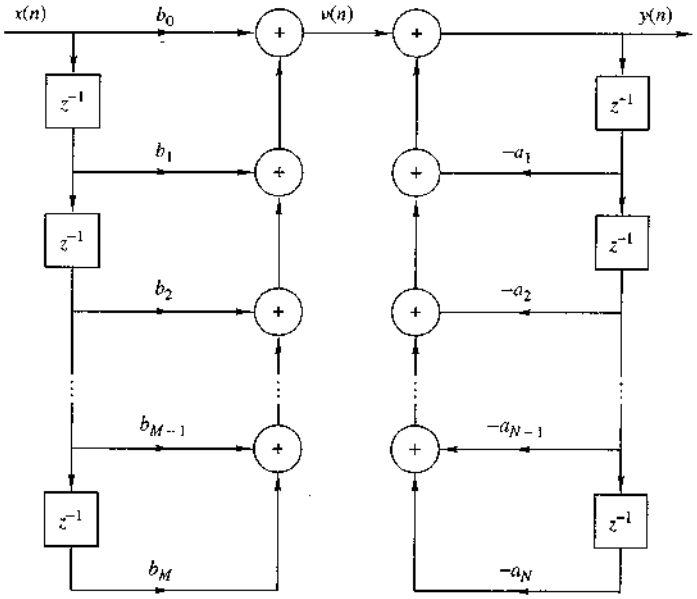


Figure 2.5.2 Direct form I structure of the system described by (2.5.6).

We observe that if $N \geq M$, this structure requires a number of delays equal the order N of the system. However, if $M > N$, the required memory is specifi by M. Figure 2.5.3 can easily be modified to handle this case. Thus the direct for II structure requires $M + N + 1$ multiplications and $\max\{M, N\}$ delays. Because requires the minimum number of delays for the realization of the system describ by (2.5.6), it is sometimes called a *canonic form*.

A special case of (2.5.6) occurs if we set the system parameters $a_k = 0$, k $1, \ldots, N$. Then the input–output relationship for the system reduces to

$$y(n) = \sum_{k=0}^{M} b_k x(n-k) \qquad (2.5.1$$

which is a nonrecursive linear time-invariant system. This system views only the mo recent $M + 1$ input signal samples and, prior to addition, weights each sample b the appropriate coefficient b_k from the set $\{b_k\}$. In other words, the system outp is basically a *weighted moving average* of the input signal. For this reason it sometimes called a *moving average (MA) system*. Such a system is an FIR syste

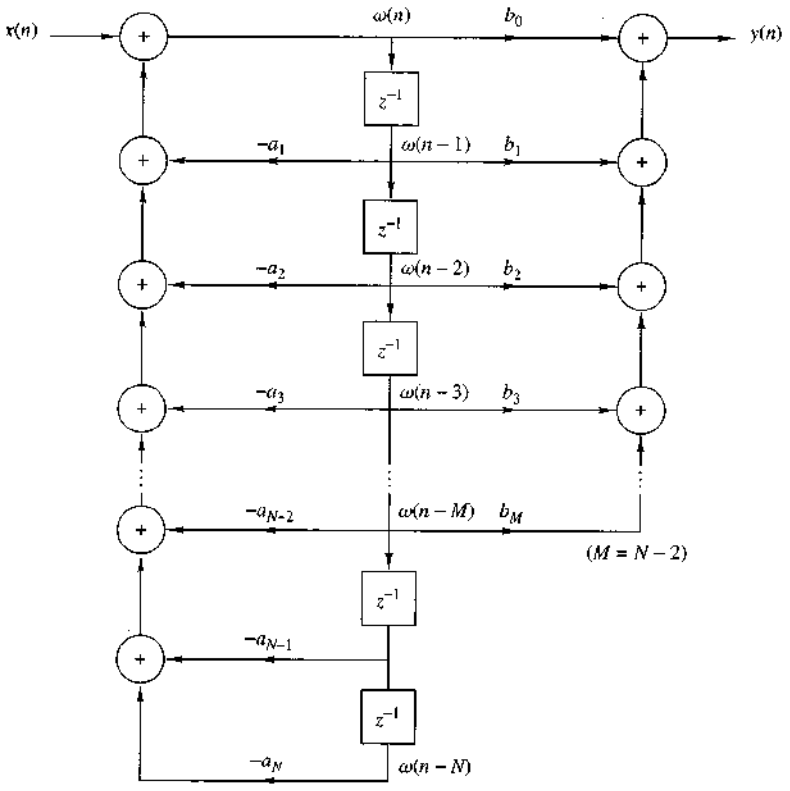


Figure 2.5.3 Direct form II structure for the system described by (2.5.6).

with an impulse response $h(k)$ equal to the coefficients b_k, that is,

$$h(k) = \begin{cases} b_k, & 0 \le k \le M \\ 0, & \text{otherwise} \end{cases} \tag{2.5.12}$$

If we return to (2.5.6) and set $M = 0$, the general linear time-invariant system reduces to a "purely recursive" system described by the difference equation

$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + b_0 x(n) \tag{2.5.13}$$

In this case the system output is a weighted linear combination of N past outputs and the present input.

Linear time-invariant systems described by a second-order difference equation are an important subclass of the more general systems described by (2.5.6) or (2.5.10) or (2.5.13). The reason for their importance will be explained later when we discuss quantization effects. Suffice to say at this point that second-order systems are usually used as basic building blocks for realizing higher-order systems.

The most general second-order system is described by the difference equation

$$\begin{aligned} y(n) = &- a_1 y(n-1) - a_2 y(n-2) + b_0 x(n) \\ &+ b_1 x(n-1) + b_2 x(n-2) \end{aligned} \tag{2.5.14}$$

which is obtained from (2.5.6) by setting $N = 2$ and $M = 2$. The direct form II structure for realizing this system is shown in Fig. 2.5.4(a). If we set $a_1 = a_2 = 0$, then (2.5.14) reduces to

$$y(n) = b_0 x(n) + b_1 x(n-1) + b_2 x(n-2) \tag{2.5.15}$$

which is a special case of the FIR system described by (2.5.11). The structure for realizing this system is shown in Fig. 2.5.4(b). Finally, if we set $b_1 = b_2 = 0$ in (2.5.14), we obtain the purely recursive second-order system described by the difference equation

$$y(n) = -a_1 y(n-1) - a_2 y(n-2) + b_0 x(n) \tag{2.5.16}$$

which is a special case of (2.5.13). The structure for realizing this system is shown in Fig. 2.5.4(c).

2.5.2 Recursive and Nonrecursive Realizations of FIR Systems

We have already made the distinction between FIR and IIR systems, based on whether the impulse response $h(n)$ of the system has a finite duration, or an infinite duration. We have also made the distinction between recursive and nonrecursive systems. Basically, a causal recursive system is described by an input–output equation of the form

$$y(n) = F[y(n-1), \ldots, y(n-N), x(n), \ldots, x(n-M)] \tag{2.5.17}$$

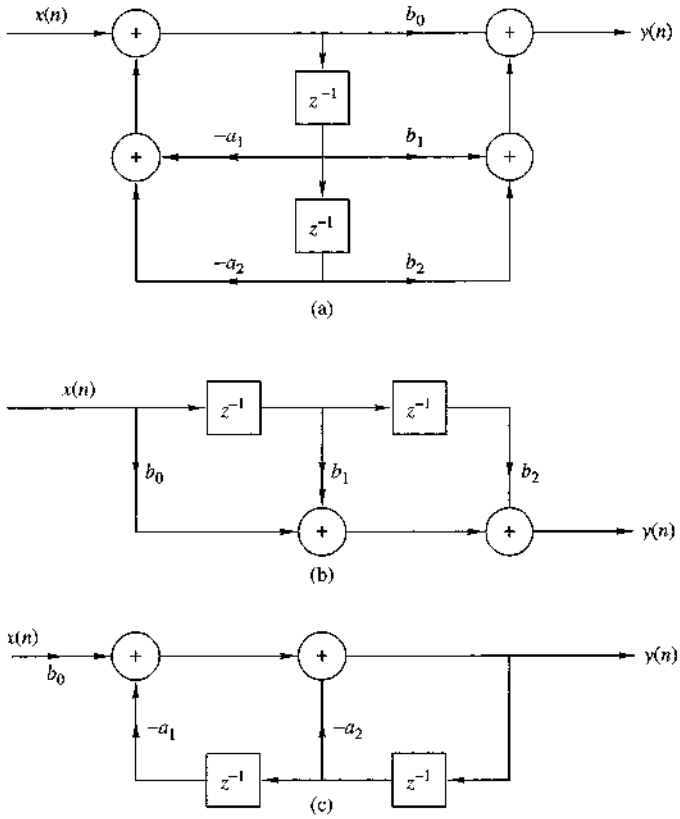


Figure 2.5.4 Structures for the realization of second-order systems: (a) general second-order system; (b) FIR system; (c) "purely recursive system."

and for a linear time-invariant system specifically, by the difference equation

$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k) \tag{2.5.18}$$

On the other hand, causal nonrecursive systems do not depend on past values of the output and hence are described by an input–output equation of the form

$$y(n) = F[x(n), x(n-1), \ldots, x(n-M)] \tag{2.5.19}$$

and for linear time-invariant systems specifically, by the difference equation in (2.5.18) with $a_k = 0$ for $k = 1, 2, \ldots, N$.

In the case of FIR systems, we have already observed that it is always possible to realize such systems nonrecursively. In fact, with $a_k = 0, k = 1, 2, \ldots, N$, in (2.5.18)

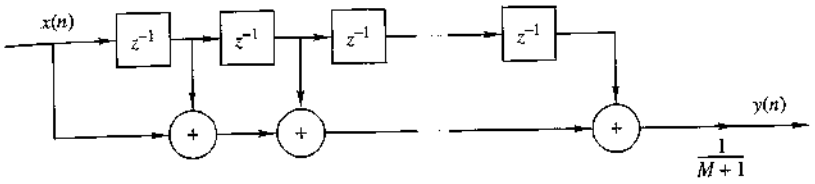


Figure 2.5.5 Nonrecursive realization of an FIR moving average system.

we have a system with an input–output equation

$$y(n) = \sum_{k=0}^{M} b_k x(n-k) \tag{2.5.20}$$

This is a nonrecursive and FIR system. As indicated in (2.5.12), the impulse response of the system is simply equal to the coefficients $\{b_k\}$. Hence every FIR system can be realized nonrecursively. On the other hand, any FIR system can also be realized recursively. Although the general proof of this statement is given later, we shall give a simple example to illustrate the point.

Suppose that we have an FIR system of the form

$$y(n) = \frac{1}{M+1} \sum_{k=0}^{M} x(n-k) \tag{2.5.21}$$

for computing the *moving average* of a signal $x(n)$. Clearly, this system is FIR with impulse response

$$h(n) = \frac{1}{M+1}, \qquad 0 \le n \le M$$

Figure 2.5.5 illustrates the structure of the nonrecursive realization of the system. Now, suppose that we express (2.5.21) as

$$\begin{aligned} y(n) &= \frac{1}{M+1} \sum_{k=0}^{M} x(n-1-k) \\ &\quad + \frac{1}{M+1}[x(n) - x(n-1-M)] \\ &= y(n-1) + \frac{1}{M+1}[x(n) - x(n-1-M)] \end{aligned} \tag{2.5.22}$$

Now, (2.5.22) represents a recursive realization of the FIR system. The structure of this recursive realization of the moving average system is illustrated in Fig. 2.5.6.

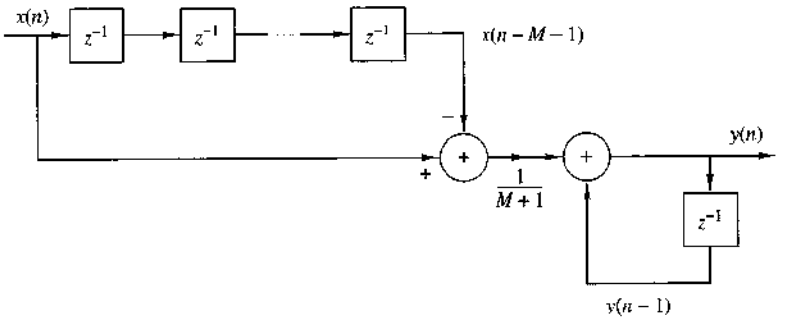


Figure 2.5.6 Recursive realization of an FIR moving average system.

In summary, we can think of the terms FIR and IIR as general characteristics that distinguish a type of linear time-invariant system, and of the terms *recursive* and *nonrecursive* as descriptions of the structures for realizing or implementing the system.

2.6 Correlation of Discrete-Time Signals

A mathematical operation that closely resembles convolution is correlation. Just as in the case of convolution, two signal sequences are involved in correlation. In contrast to convolution, however, our objective in computing the correlation between the two signals is to measure the degree to which the two signals are similar and thus to extract some information that depends to a large extent on the application. Correlation of signals is often encountered in radar, sonar, digital communications, geology, and other areas in science and engineering.

To be specific, let us suppose that we have two signal sequences $x(n)$ and $y(n)$ that we wish to compare. In radar and active sonar applications, $x(n)$ can represent the sampled version of the transmitted signal and $y(n)$ can represent the sampled version of the received signal at the output of the analog-to-digital (A/D) converter. If a target is present in the space being searched by the radar or sonar, the received signal $y(n)$ consists of a delayed version of the transmitted signal, reflected from the target, and corrupted by additive noise. Figure 2.6.1 depicts the radar signal reception problem.

We can represent the received signal sequence as

$$y(n) = \alpha x(n - D) + w(n) \tag{2.6.1}$$

where α is some attenuation factor representing the signal loss involved in the round-trip transmission of the signal $x(n)$, D is the round-trip delay, which is assumed to be an integer multiple of the sampling interval, and $w(n)$ represents the additive noise that is picked up by the antenna and any noise generated by the electronic components and amplifiers contained in the front end of the receiver. On the other hand, if there is no target in the space searched by the radar and sonar, the received signal $y(n)$ consists of noise alone.

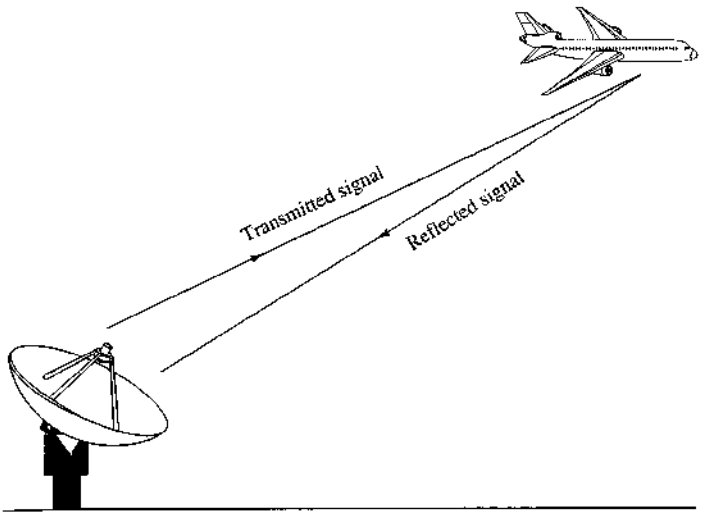


Figure 2.6.1 Radar target detection.

Having the two signal sequences, $x(n)$, which is called the reference signal or transmitted signal, and $y(n)$, the received signal, the problem in radar and sonar detection is to compare $y(n)$ and $x(n)$ to determine if a target is present and, if so, to determine the time delay D and compute the distance to the target. In practice, the signal $x(n - D)$ is heavily corrupted by the additive noise to the point where a visual inspection of $y(n)$ does not reveal the presence or absence of the desired signal reflected from the target. Correlation provides us with a means for extracting this important information from $y(n)$.

Digital communications is another area where correlation is often used. In digital communications the information to be transmitted from one point to another is usually converted to binary form, that is, a sequence of zeros and ones, which are then transmitted to the intended receiver. To transmit a 0 we can transmit the signal sequence $x_0(n)$ for $0 \leq n \leq L - 1$, and to transmit a 1 we can transmit the signal sequence $x_1(n)$ for $0 \leq n \leq L-1$, where L is some integer that denotes the number of samples in each of the two sequences. Very often, $x_1(n)$ is selected to be the negative of $x_0(n)$. The signal received by the intended receiver may be represented as

$$y(n) = x_i(n) + w(n), \qquad i = 0, 1, \qquad 0 \leq n \leq L - 1 \tag{2.6.2}$$

where now the uncertainty is whether $x_0(n)$ or $x_1(n)$ is the signal component in $y(n)$, and $w(n)$ represents the additive noise and other interference inherent in any communication system. Again, such noise has its origin in the electronic components contained in the front end of the receiver. In any case, the receiver knows the possible transmitted sequences $x_0(n)$ and $x_1(n)$ and is faced with the task of comparing the received signal $y(n)$ with both $x_0(n)$ and $x_1(n)$ to determine which of the two signals better matches $y(n)$. This comparison process is performed by means of the correlation operation described in the following subsection.

2.6.1 Crosscorrelation and Autocorrelation Sequences

Suppose that we have two real signal sequences $x(n)$ and $y(n)$ each of which has finite energy. The *crosscorrelation* of $x(n)$ and $y(n)$ is a sequence $r_{xy}(l)$, which is defined as

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n)y(n-l), \qquad l = 0, \pm 1, \pm 2, \ldots \tag{2.6.3}$$

or, equivalently, as

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n+l)y(n), \qquad l = 0, \pm 1, \pm 2, \ldots \tag{2.6.4}$$

The index l is the (time) shift (or *lag*) parameter and the subscripts xy on the crosscorrelation sequence $r_{xy}(l)$ indicate the sequences being correlated. The order of the subscripts, with x preceding y, indicates the direction in which one sequence is shifted, relative to the other. To elaborate, in (2.6.3), the sequence $x(n)$ is left unshifted and $y(n)$ is shifted by l units in time, to the right for l positive and to the left for l negative. Equivalently, in (2.6.4), the sequence $y(n)$ is left unshifted and $x(n)$ is shifted by l units in time, to the left for l positive and to the right for l negative. But shifting $x(n)$ to the left by l units relative to $y(n)$ is equivalent to shifting $y(n)$ to the right by l units relative to $x(n)$. Hence the computations (2.6.3) and (2.6.4) yield identical crosscorrelation sequences.

If we reverse the roles of $x(n)$ and $y(n)$ in (2.6.3) and (2.6.4) and therefore reverse the order of the indices xy, we obtain the crosscorrelation sequence

$$r_{yx}(l) = \sum_{n=-\infty}^{\infty} y(n)x(n-l) \tag{2.6.5}$$

or, equivalently,

$$r_{yx}(l) = \sum_{n=-\infty}^{\infty} y(n+l)x(n) \tag{2.6.6}$$

By comparing (2.6.3) with (2.6.6) or (2.6.4) with (2.6.5), we conclude that

$$r_{xy}(l) = r_{yx}(-l) \tag{2.6.7}$$

Therefore, $r_{yx}(l)$ is simply the folded version of $r_{xy}(l)$, where the folding is done with respect to $l = 0$. Hence, $r_{yx}(l)$ provides exactly the same information as $r_{xy}(l)$, with respect to the similarity of $x(n)$ to $y(n)$.

EXAMPLE 2.6.1

Determine the crosscorrelation sequence $r_{xy}(l)$ of the sequences

$$x(n) = \{\ldots, 0, 0, 2, -1, 3, 7, \underset{\uparrow}{1}, 2, -3, 0, 0, \ldots\}$$

$$y(n) = \{\ldots, 0, 0, 1, -1, 2, -2, \underset{\uparrow}{4}, 1, -2, 5, 0, 0, \ldots\}$$

Solution. Let us use the definition in (2.6.3) to compute $r_{xy}(l)$. For $l = 0$ we have

$$r_{xy}(0) = \sum_{n=-\infty}^{\infty} x(n)y(n)$$

The product sequence $v_0(n) = x(n)y(n)$ is

$$v_0(n) = \{\ldots, 0, 0, 2, 1, 6, -14, \underset{\uparrow}{4}, 2, 6, 0, 0, \ldots\}$$

and hence the sum over all values of n is

$$r_{xy}(0) = 7$$

For $l > 0$, we simply shift $y(n)$ to the right relative to $x(n)$ by l units, compute the product sequence $v_l(n) = x(n)y(n-l)$, and finally, sum over all values of the product sequence. Thus we obtain

$$r_{xy}(1) = 13, \quad r_{xy}(2) = -18, \quad r_{xy}(3) = 16, \quad r_{xy}(4) = -7$$
$$r_{xy}(5) = 5, \quad r_{xy}(6) = -3, \quad r_{xy}(l) = 0, \quad l \geq 7$$

For $l < 0$, we shift $y(n)$ to the left relative to $x(n)$ by l units, compute the product sequence $v_l(n) = x(n)y(n-l)$, and sum over all values of the product sequence. Thus we obtain the values of the crosscorrelation sequence

$$r_{xy}(-1) = 0, \quad r_{xy}(-2) = 33, \quad r_{xy}(-3) = -14, \quad r_{xy}(-4) = 36$$
$$r_{xy}(-5) = 19, \quad r_{xy}(-6) = -9, \quad r_{xy}(-7) = 10, \quad r_{xy}(l) = 0, \ l \leq -8$$

Therefore, the crosscorrelation sequence of $x(n)$ and $y(n)$ is

$$r_{xy}(l) = \{10, -9, 19, 36, -14, 33, 0, \underset{\uparrow}{7}, 13, -18, 16, -7, 5, -3\}$$

The similarities between the computation of the crosscorrelation of two sequences and the convolution of two sequences is apparent. In the computation of convolution, one of the sequences is folded, then shifted, then multiplied by the other sequence to form the product sequence for that shift, and finally, the values of the product sequence are summed. Except for the folding operation, the computation of the crosscorrelation sequence involves the same operations: shifting one of the sequences, multiplying the two sequences, and summing over all values of the product sequence. Consequently, if we have a computer program that performs convolution, we can use it to perform crosscorrelation by providing as inputs to the program the sequence $x(n)$ and the folded sequence $y(-n)$. Then the convolution of $x(n)$ with $y(-n)$ yields the crosscorrelation $r_{xy}(l)$, that is,

$$r_{xy}(l) = x(l) * y(-l) \tag{2.6.8}$$

We note that the absence of folding makes crosscorrelation a noncommutative operation. In the special case where $y(n) = x(n)$, we have the *autocorrelation* of $x(n)$, which is defined as the sequence

$$r_{xx}(l) = \sum_{n=-\infty}^{\infty} x(n)x(n-l) \tag{2.6.9}$$

or, equivalently, as

$$r_{xx}(l) = \sum_{n=-\infty}^{\infty} x(n+l)x(n) \tag{2.6.10}$$

In dealing with finite-duration sequences, it is customary to express the autocorrelation and crosscorrelation in terms of the finite limits on the summation. In particular, if $x(n)$ and $y(n)$ are causal sequences of length N [i.e., $x(n) = y(n) = 0$ for $n < 0$ and $n \geq N$], the crosscorrelation and autocorrelation sequences may be expressed as

$$r_{xy}(l) = \sum_{n=l}^{N-|k|-1} x(n)y(n-l) \tag{2.6.11}$$

and

$$r_{xx}(l) = \sum_{n=i}^{N-|k|-1} x(n)x(n-l) \tag{2.6.12}$$

where $i = l$, $k = 0$ for $l \geq 0$, and $i = 0$, $k = l$ for $l < 0$.

2.6.2 Properties of the Autocorrelation and Crosscorrelation Sequences

The autocorrelation and crosscorrelation sequences have a number of important properties that we now present. To develop these properties, let us assume that we have two sequences $x(n)$ and $y(n)$ with finite energy from which we form the linear combination,

$$ax(n) + by(n-l)$$

where a and b are arbitrary constants and l is some time shift. The energy in this signal is

$$\begin{aligned}\sum_{n=-\infty}^{\infty} [ax(n) + by(n-l)]^2 &= a^2 \sum_{n=-\infty}^{\infty} x^2(n) + b^2 \sum_{n=-\infty}^{\infty} y^2(n-l) \\ &\quad + 2ab \sum_{n=-\infty}^{\infty} x(n)y(n-l) \\ &= a^2 r_{xx}(0) + b^2 r_{yy}(0) + 2abr_{xy}(l)\end{aligned} \tag{2.6.13}$$

First, we note that $r_{xx}(0) = E_x$ and $r_{yy}(0) = E_y$, which are the energies of $x(n)$ and $y(n)$, respectively. It is obvious that

$$a^2 r_{xx}(0) + b^2 r_{yy}(0) + 2abr_{xy}(l) \geq 0 \tag{2.6.14}$$

Now, assuming that $b \neq 0$, we can divide (2.6.14) by b^2 to obtain

$$r_{xx}(0)\left(\frac{a}{b}\right)2 + 2r_{xy}(l)\left(\frac{a}{b}\right) + r_{yy}(0) \geq 0$$

We view this equation as a quadratic with coefficients $r_{xx}(0)$, $2r_{xy}(l)$, and $r_{yy}(0)$. Since the quadratic is nonnegative, it follows that the discriminant of this quadratic must be nonpositive, that is,

$$4[r_{xy}^2(l) - r_{xx}(0)r_{yy}(0)] \leq 0$$

Therefore, the crosscorrelation sequence satisfies the condition that

$$|r_{xy}(l)| \leq \sqrt{r_{xx}(0)r_{yy}(0)} = \sqrt{E_x E_y} \tag{2.6.15}$$

In the special case where $y(n) = x(n)$, (2.6.15) reduces to

$$|r_{xx}(l)| \leq r_{xx}(0) = E_x \tag{2.6.16}$$

This means that the autocorrelation sequence of a signal attains its maximum value at zero lag. This result is consistent with the notion that a signal matches perfectly with itself at zero shift. In the case of the crosscorrelation sequence, the upper bound on its values is given in (2.6.15).

Note that if any one or both of the signals involved in the crosscorrelation are scaled, the shape of the crosscorrelation sequence does not change; only the amplitudes of the crosscorrelation sequence are scaled accordingly. Since scaling is unimportant, it is often desirable, in practice, to normalize the autocorrelation and crosscorrelation sequences to the range from -1 to 1. In the case of the autocorrelation sequence, we can simply divide by $r_{xx}(0)$. Thus the normalized autocorrelation sequence is defined as

$$\rho_{xx}(l) = \frac{r_{xx}(l)}{r_{xx}(0)} \tag{2.6.17}$$

Similarly, we define the normalized crosscorrelation sequence

$$\rho_{xy}(l) = \frac{r_{xy}(l)}{\sqrt{r_{xx}(0)r_{yy}(0)}} \tag{2.6.18}$$

Now $|\rho_{xx}(l)| \leq 1$ and $|\rho_{xy}(l)| \leq 1$, and hence these sequences are independent of signal scaling.

Finally, as we have already demonstrated, the crosscorrelation sequence satisfies the property

$$r_{xy}(l) = r_{yx}(-l)$$

With $y(n) = x(n)$, this relation results in the following important property for the autocorrelation sequence

$$r_{xx}(l) = r_{xx}(-l) \tag{2.6.19}$$

Hence the autocorrelation function is an even function. Consequently, it suffices to compute $r_{xx}(l)$ for $l \geq 0$.

EXAMPLE 2.6.2

Compute the autocorrelation of the signal

$$x(n) = a^n u(n), \, 0 < a < 1$$

Solution. Since $x(n)$ is an infinite-duration signal, its autocorrelation also has infinite duration. We distinguish two cases.

If $l \geq 0$, from Fig. 2.6.2 we observe that

$$r_{xx}(l) = \sum_{n=1}^{\infty} x(n)x(n-l) = \sum_{n=1}^{\infty} a^n a^{n-l} = a^{-l} \sum_{n=1}^{\infty} (a^2)^n$$

Since $a < 1$, the infinite series *converges* and we obtain

$$r_{xx}(l) = \frac{1}{1-a^2} a^{|l|}, \qquad l \geq 0$$

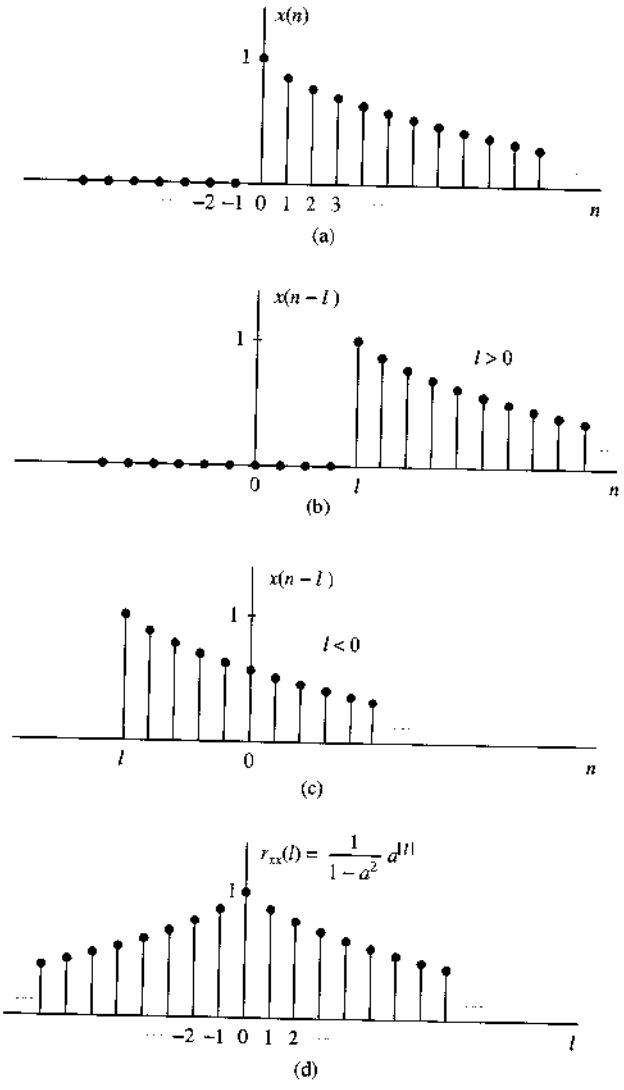


Figure 2.6.2 Computation of the autocorrelation of the signal $x(n) = a^n$, $0 < a < 1$.

For $l < 0$ we have

$$r_{xx}(l) = \sum_{n=0}^{\infty} x(n)x(n-l) = a^{-l}\sum_{n=0}^{\infty}(a^2)^n = \frac{1}{1-a^2}a^{-l}, \qquad l < 0$$

But when l is negative, $a^{-l} = a^{|l|}$. Thus the two relations for $r_{xx}(l)$ can be combined into the following expression:

$$r_{xx}(l) = \frac{1}{1-a^2}a^{|l|}, \qquad -\infty < l < \infty \tag{2.6.20}$$

The sequence $r_{xx}(l)$ is shown in Fig. 2.6.2(d). We observe that

$$r_{xx}(-l) = r_{xx}(l)$$

and

$$r_{xx}(0) = \frac{1}{1-a^2}$$

Therefore, the normalized autocorrelation sequence is

$$\rho_{xx}(l) = \frac{r_{xx}(l)}{r_{xx}(0)} = a^{|l|}, \qquad -\infty < l < \infty \tag{2.6.21}$$

2.6.3 Correlation of Periodic Sequences

In Section 2.6.1 we defined the crosscorrelation and autocorrelation sequences of energy signals. In this section we consider the correlation sequences of power signals and, in particular, periodic signals.

Let $x(n)$ and $y(n)$ be two power signals. Their crosscorrelation sequence is defined as

$$r_{xy}(l) = \lim_{M\to\infty}\frac{1}{2M+1}\sum_{n=-M}^{M} x(n)y(n-l) \tag{2.6.22}$$

If $x(n) = y(n)$, we have the definition of the autocorrelation sequence of a power signal as

$$r_{xx}(l) = \lim_{M\to\infty}\frac{1}{2M+1}\sum_{n=-M}^{M} x(n)x(n-l) \tag{2.6.23}$$

In particular, if $x(n)$ and $y(n)$ are two periodic sequences, each with period N, the averages indicated in (2.6.22) and (2.6.23) over the infinite interval are identical to the averages over a single period, so that (2.6.22) and (2.6.23) reduce to

$$r_{xy}(l) = \frac{1}{N}\sum_{n=0}^{N-1} x(n)y(n-l) \tag{2.6.24}$$

and

$$r_{xx}(l) = \frac{1}{N}\sum_{n=0}^{N-1} x(n)x(n-l) \tag{2.6.25}$$

It is clear that $r_{xy}(l)$ and $r_{xx}(l)$ are periodic correlation sequences with period N. The factor $1/N$ can be viewed as a normalization scale factor.

In some practical applications, correlation is used to identify periodicities in an observed physical signal which may be corrupted by random interference. For example, consider a signal sequence $y(n)$ of the form

$$y(n) = x(n) + w(n) \tag{2.6.26}$$

where $x(n)$ is a periodic sequence of some unknown period N and $w(n)$ represents an additive random interference. Suppose that we observe M samples of $y(n)$, say, $0 \le n \le M-1$, where $M >> N$. For all practical purposes, we can assume that $y(n) = 0$ for $n < 0$ and $n \ge M$. Now the autocorrelation sequence of $y(n)$, using the normalization factor of $1/M$, is

$$r_{yy}(l) = \frac{1}{M}\sum_{n=0}^{M-1} y(n)y(n-l) \tag{2.6.27}$$

If we substitute for $y(n)$ from (2.6.26) into (2.6.27) we obtain

$$\begin{aligned} r_{yy}(l) &= \frac{1}{M}\sum_{n=0}^{M-1}[x(n)+w(n)][x(n-l)+w(n-l)] \\ &= \frac{1}{M}\sum_{n=0}^{M-1} x(n)x(n-l) \\ &\quad + \frac{1}{M}\sum_{n=0}^{M-1}[x(n)w(n-l)+w(n)x(n-l)] \\ &\quad + \frac{1}{M}\sum_{n=0}^{M-1} w(n)w(n-l) \\ &= r_{xx}(l) + r_{xw}(l) + r_{wx}(l) + r_{ww}(l) \end{aligned} \tag{2.6.28}$$

The first factor on the right-hand side of (2.6.28) is the autocorrelation sequence of $x(n)$. Since $x(n)$ is periodic, its autocorrelation sequence exhibits the same periodicity, thus containing relatively large peaks at $l = 0, N, 2N$, and so on. However, as the shift l approaches M, the peaks are reduced in amplitude due to the fact that we have a finite data record of M samples so that many of the products $x(n)x(n-l)$ are zero. Consequently, we should avoid computing $r_{yy}(l)$ for large lags, say, $l > M/2$.

The crosscorrelations $r_{xw}(l)$ and $r_{wx}(l)$ between the signal $x(n)$ and the additive random interference are expected to be relatively small as a result of the expectation that $x(n)$ and $w(n)$ will be totally unrelated. Finally, the last term on the right-hand side of (2.6.28) is the autocorrelation sequence of the random sequence $w(n)$. This correlation sequence will certainly contain a peak at $l = 0$, but because of its random characteristics, $r_{ww}(l)$ is expected to decay rapidly toward zero. Consequently, only $r_{xx}(l)$ is expected to have large peaks for $l > 0$. This behavior allows us to detect the presence of the periodic signal $x(n)$ buried in the interference $w(n)$ and to identify its period.

An example that illustrates the use of autocorrelation to identify a hidden periodicity in an observed physical signal is shown in Fig. 2.6.3. This figure illustrates the autocorrelation (normalized) sequence for the Wölfer sunspot numbers in the 100-year period 1770–1869 for $0 \le l \le 20$, where any value of l corresponds to one year. There is clear evidence in this figure that a periodic trend exists, with a period of 10 to 11 years.

EXAMPLE 2.6.3

Suppose that a signal sequence $x(n) = \sin(\pi/5)n$, for $0 \le n \le 99$ is corrupted by an additive noise sequence $w(n)$, where the values of the additive noise are selected independently from sample to sample, from a uniform distribution over the range $(-\Delta/2, \Delta/2)$, where Δ is a parameter of the distribution. The observed sequence is $y(n) = x(n) + w(n)$. Determine the autocorrelation sequence $r_{yy}(l)$ and thus determine the period of the signal $x(n)$.

Solution. The assumption is that the signal sequence $x(n)$ has some unknown period that we are attempting to determine from the noise-corrupted observations $\{y(n)\}$. Although $x(n)$ is periodic with period 10, we have only a finite-duration sequence of length $M = 100$ [i.e., 10 periods of $x(n)$]. The noise power level P_w in the sequence $w(n)$ is determined by the parameter Δ. We simply state that $P_w = \Delta^2/12$. The signal power level is $P_x = \frac{1}{2}$. Therefore, the signal-to-noise ratio (SNR) is defined as

$$\frac{P_x}{P_w} = \frac{\frac{1}{2}}{\Delta^2/12} = \frac{6}{\Delta^2}$$

Usually, the SNR is expressed on a logarithmic scale in decibels (dB) as $10 \log_{10} (P_x/P_w)$.

Figure 2.6.4 illustrates a sample of a noise sequence $w(n)$, and the observed sequence $y(n) = x(n) + w(n)$ when the SNR = 1 dB. The autocorrelation sequence $r_{yy}(l)$ is illustrated in Fig. 2.6.4(c). We observe that the periodic signal $x(n)$, embedded in $y(n)$, results in a periodic autocorrelation function $r_{xx}(l)$ with period $N = 10$. The effect of the additive noise is to add to the peak value at $l = 0$, but for $l \neq 0$, the correlation sequence $r_{ww}(l) \approx 0$ as a result of the fact that values of $w(n)$ were generated independently. Such noise is usually called *white noise*. The presence of this noise explains the reason for the large peak at $l = 0$. The smaller, nearly equal peaks at $l = \pm 10, \pm 20, \ldots$ are due to the periodic characteristics of $x(n)$.

2.6.4 Input–Output Correlation Sequences

In this section we derive two input–output relationships for LTI systems in the "correlation domain." Let us assume that a signal $x(n)$ with known autocorrelation $r_{xx}(l)$

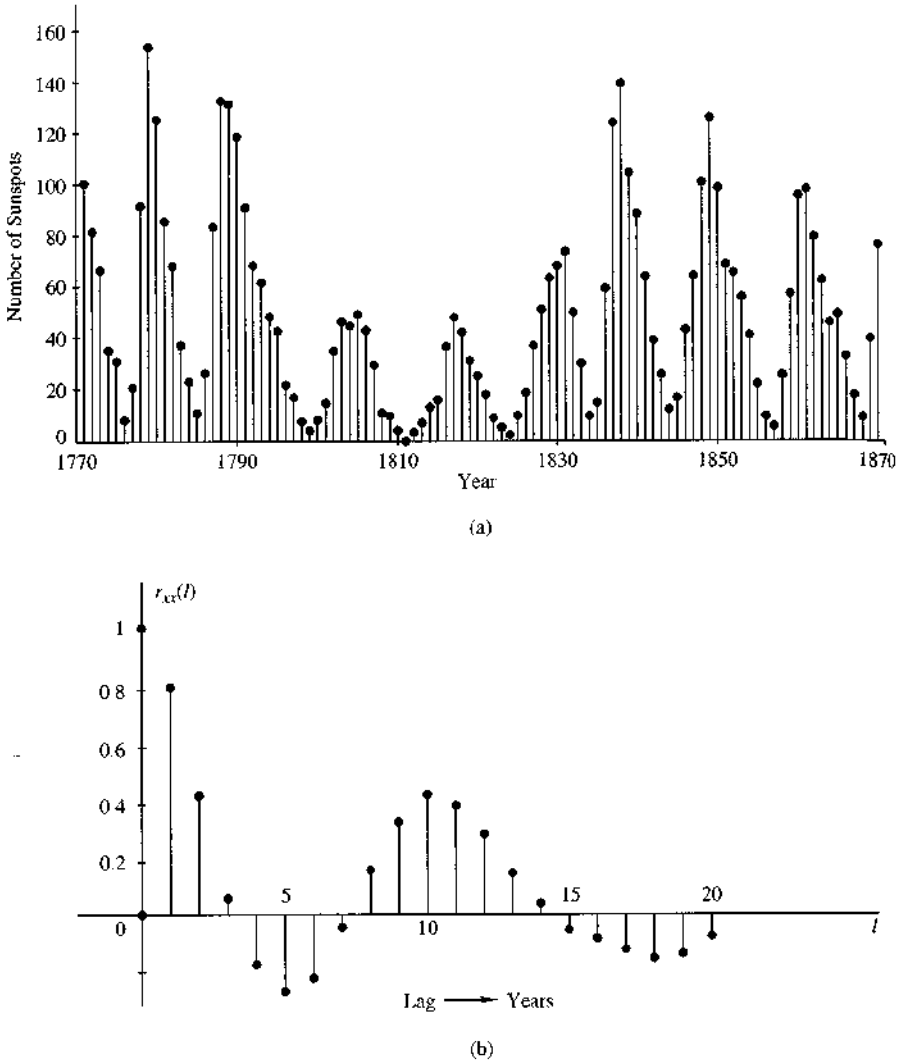


Figure 2.6.3 Identification of periodicity in the Wölfer sunspot numbers: (a) annual Wölfer sunspot numbers; (b) normalized autocorrelation sequence.

is applied to an LTI system with impulse response $h(n)$, producing the output signal

$$y(n) = h(n) * x(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k)$$

The crosscorrelation between the output and the input signal is

$$r_{yx}(l) = y(l) * x(-l) = h(l) * [x(l) * x(-l)]$$

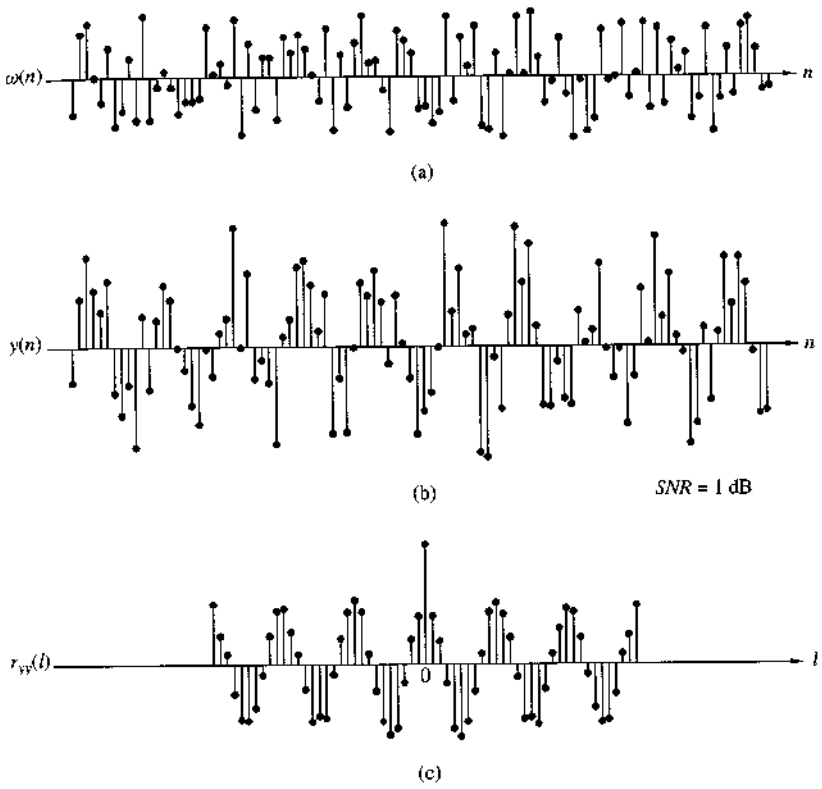


Figure 2.6.4 Use of autocorrelation to detect the presence of a periodic signal corrupted by noise.

or

$$r_{yx}(l) = h(l) * r_{xx}(l) \tag{2.6.29}$$

where we have used (2.6.8) and the properties of convolution. Hence the cross-correlation between the input and the output of the system is the convolution of the impulse response with the autocorrelation of the input sequence. Alternatively, $r_{yx}(l)$ may be viewed as the output of the LTI system when the input sequence is $r_{xx}(l)$. This is illustrated in Fig. 2.6.5. If we replace l by $-l$ in (2.6.29), we obtain

$$r_{xy}(l) = h(-l) * r_{xx}(l)$$

The autocorrelation of the output signal can be obtained by using (2.6.8) with $x(n) = y(n)$ and the properties of convolution. Thus we have

$$\begin{aligned} r_{yy}(l) &= y(l) * y(-l) \\ &= [h(l) * x(l)] * [h(-l) * x(-l)] \\ &= [h(l) * h(-l)] * [x(l) * x(-l)] \\ &= r_{hh}(l) * r_{xx}(l) \end{aligned} \tag{2.6.30}$$

The autocorrelation $r_{hh}(l)$ of the impulse response $h(n)$ exists if the system is stable. Furthermore, the stability insures that the system does not change the type (energy or power) of the input signal. By evaluating (2.6.30) for $l = 0$ we obtain

$$r_{yy}(0) = \sum_{k=-\infty}^{\infty} r_{hh}(k) r_{xx}(k) \tag{2.6.31}$$

which provides the energy (or power) of the output signal in terms of autocorrelations. These relationships hold for both energy and power signals. The direct derivation of these relationships for energy and power signals, and their extensions to complex signals, are left as exercises for the student.

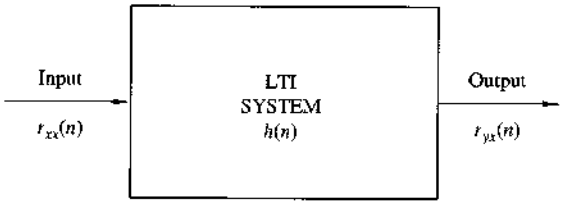


Figure 2.6.5
Input–output relation for crosscorrelation $r_{yx}(n)$.

2.7 Summary and References

The major theme of this chapter is the characterization of discrete-time signals and systems in the time domain. Of particular importance is the class of linear time-invariant (LTI) systems which are widely used in the design and implementation of digital signal processing systems. We characterized LTI systems by their unit sample response $h(n)$ and derived the convolution summation, which is a formula for determining the response $y(n)$ of the system characterized by $h(n)$ to any given input sequence $x(n)$.

The class of LTI systems characterized by linear difference equations with constant coefficients is by far the most important of the LTI systems in the theory and application of digital signal processing. The general solution of a linear difference equation with constant coefficients was derived in this chapter and shown to consist of two components: the solution of the homogeneous equation, which represents the natural response of the system when the input is zero, and the particular solution, which represents the response of the system to the input signal. From the difference equation, we also demonstrated how to derive the unit sample response of the LTI system.

Linear time-invariant systems were generally subdivided into FIR (finite-duration impulse response) and IIR (infinite-duration impulse response) depending on whether $h(n)$ has finite duration or infinite duration, respectively. The realizations of such systems were briefly described. Furthermore, in the realization of FIR systems, we made the distinction between recursive and nonrecursive realizations. On the other hand, we observed that IIR systems can be implemented recursively, only.

There are a number of texts on discrete-time signals and systems. We mention as examples the books by McGillem and Cooper (1984), Oppenheim and Willsky (1983), and Siebert (1986). Linear constant-coefficient difference equations are treated in depth in the books by Hildebrand (1952) and Levy and Lessman (1961).

The last topic in this chapter, on correlation of discrete-time signals, plays an important role in digital signal processing, especially in applications dealing with digital communications, radar detection and estimation, sonar, and geophysics. In our treatment of correlation sequences, we avoided the use of statistical concepts. Correlation is simply defined as a mathematical operation between two sequences, which produces another sequence, called either the *crosscorrelation sequence* when the two sequences are different, or the *autocorrelation sequence* when the two sequences are identical.

In practical applications in which correlation is used, one (or both) of the sequences is (are) contaminated by noise and, perhaps, by other forms of interference. In such a case, the noisy sequence is called a *random sequence* and is characterized in statistical terms. The corresponding correlation sequence becomes a function of the statistical characteristics of the noise and any other interference.

The statistical characterization of sequences and their correlation is treated in Chapter 12. Supplementary reading on probabilistic and statistical concepts dealing with correlation can be found in the books by Davenport (1970), Helstrom (1990), Peebles (1987), and Stark and Woods (1994).

Problems

2.1 A discrete-time signal $x(n)$ is defined as

$$x(n) = \begin{cases} 1 + \frac{n}{3}, & -3 \le n \le -1 \\ 1, & 0 \le n \le 3 \\ 0, & \text{elsewhere} \end{cases}$$

(a) Determine its values and sketch the signal $x(n)$.

(b) Sketch the signals that result if we:

1. First fold $x(n)$ and then delay the resulting signal by four samples.
2. First delay $x(n)$ by four samples and then fold the resulting signal.

(c) Sketch the signal $x(-n + 4)$.

(d) Compare the results in parts (b) and (c) and derive a rule for obtaining the signal $x(-n + k)$ from $x(n)$.

(e) Can you express the signal $x(n)$ in terms of signals $\delta(n)$ and $u(n)$?

2.2 A discrete-time signal $x(n)$ is shown in Fig. P2.2. Sketch and label carefully each of the following signals.

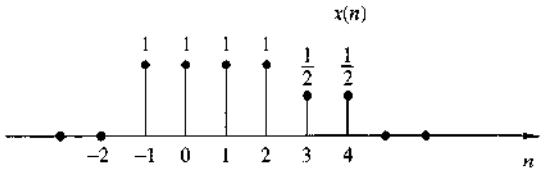


Figure P2.2

(a) $x(n-2)$ **(b)** $x(4-n)$ **(c)** $x(n+2)$ **(d)** $x(n)u(2-n)$ **(e)** $x(n-1)\delta(n-3)$ **(f)** $x(n^2)$ **(g)** even part of $x(n)$ **(h)** odd part of $x(n)$

2.3 Show that

(a) $\delta(n) = u(n) - u(n-1)$

(b) $u(n) = \sum_{k=-\infty}^{n} \delta(k) = \sum_{k=0}^{\infty} \delta(n-k)$

2.4 Show that any signal can be decomposed into an even and an odd component. Is the decomposition unique? Illustrate your arguments using the signal

$$x(n) = \{2, 3, \underset{\uparrow}{4}, 5, 6\}$$

2.5 Show that the energy (power) of a real-valued energy (power) signal is equal to the sum of the energies (powers) of its even and odd components.

2.6 Consider the system

$$y(n) = \mathcal{T}[x(n)] = x(n^2)$$

(a) Determine if the system is time invariant.

(b) To clarify the result in part (a) assume that the signal

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

is applied into the system.

(1) Sketch the signal $x(n)$.

(2) Determine and sketch the signal $y(n) = \mathcal{T}[x(n)]$.

(3) Sketch the signal $y_2'(n) = y(n-2)$.

(4) Determine and sketch the signal $x_2(n) = x(n-2)$.

(5) Determine and sketch the signal $y_2(n) = \mathcal{T}[x_2(n)]$.

(6) Compare the signals $y_2(n)$ and $y(n-2)$. What is your conclusion?

(c) Repeat part (b) for the system

$$y(n) = x(n) - x(n-1)$$

Can you use this result to make any statement about the time invariance of this system? Why?

(d) Repeat parts (b) and (c) for the system

$$y(n) = \mathcal{T}[x(n)] = nx(n)$$

2.7 A discrete-time system can be

(1) Static or dynamic
(2) Linear or nonlinear
(3) Time invariant or time varying
(4) Causal or noncausal
(5) Stable or unstable

Examine the following systems with respect to the properties above.

(a) $y(n) = \cos[x(n)]$

(b) $y(n) = \sum_{k=-\infty}^{n+1} x(k)$

(c) $y(n) = x(n)\cos(\omega_0 n)$

(d) $y(n) = x(-n+2)$

(e) $y(n) = \text{Trun}[x(n)]$, where $\text{Trun}[x(n)]$ denotes the integer part of $x(n)$, obtained by truncation

(f) $y(n) = \text{Round}[x(n)]$, where $\text{Round}[x(n)]$ denotes the integer part of $x(n)$ obtained by rounding

Remark: The systems in parts (e) and (f) are quantizers that perform truncation and rounding, respectively.

(g) $y(n) = |x(n)|$

(h) $y(n) = x(n)u(n)$

(i) $y(n) = x(n) + nx(n+1)$

(j) $y(n) = x(2n)$

(k) $y(n) = \begin{cases} x(n), & \text{if } x(n) \geq 0 \\ 0, & \text{if } x(n) < 0 \end{cases}$

(l) $y(n) = x(-n)$

(m) $y(n) = \text{sign}[x(n)]$

(n) The ideal sampling system with input $x_a(t)$ and output $x(n) = x_a(nT)$, $-\infty < n < \infty$

2.8 Two discrete-time systems $\mathcal{T}_1$ and $\mathcal{T}_2$ are connected in cascade to form a new system $\mathcal{T}$ as shown in Fig. P2.8. Prove or disprove the following statements.

(a) If $\mathcal{T}_1$ and $\mathcal{T}_2$ are linear, then $\mathcal{T}$ is linear (i.e., the cascade connection of two linear systems is linear).

(b) If $\mathcal{T}_1$ and $\mathcal{T}_2$ are time invariant, then $\mathcal{T}$ is time invariant.

(c) If $\mathcal{T}_1$ and $\mathcal{T}_2$ are causal, then $\mathcal{T}$ is causal.

(d) If $\mathcal{T}_1$ and $\mathcal{T}_2$ are linear and time invariant, the same holds for $\mathcal{T}$.

(e) If $\mathcal{T}_1$ and $\mathcal{T}_2$ are linear and time invariant, then interchanging their order does not change the system $\mathcal{T}$.

(f) As in part (e) except that $\mathcal{T}_1$, $\mathcal{T}_2$ are now time varying. (*Hint:* Use an example.)

(g) If $\mathcal{T}_1$ and $\mathcal{T}_2$ are nonlinear, then $\mathcal{T}$ is nonlinear.

(h) If $\mathcal{T}_1$ and $\mathcal{T}_2$ are stable, then $\mathcal{T}$ is stable.

(i) Show by an example that the inverses of parts (c) and (h) do not hold in general.

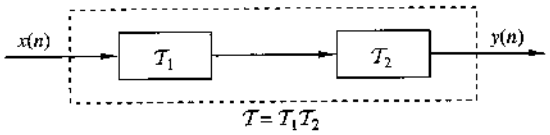


Figure P2.8

2.9 Let $\mathcal{T}$ be an LTI, relaxed, and BIBO stable system with input $x(n)$ and output $y(n)$. Show that:

(a) If $x(n)$ is periodic with period N [i.e., $x(n) = x(n+N)$ for all $n \geq 0$], the output $y(n)$ tends to a periodic signal with the same period.

(b) If $x(n)$ is bounded and tends to a constant, the output will also tend to a constant.

(c) If $x(n)$ is an energy signal, the output $y(n)$ will also be an energy signal.

2.10 The following input–output pairs have been observed during the operation of a time-invariant system:

$$x_1(n) = \{\underset{\uparrow}{1}, 0, 2\} \xleftrightarrow{\mathcal{T}} y_1(n) = \{\underset{\uparrow}{0}, 1, 2\}$$

$$x_2(n) = \{\underset{\uparrow}{0}, 0, 3\} \xleftrightarrow{\mathcal{T}} y_2(n) = \{\underset{\uparrow}{0}, 1, 0, 2\}$$

$$x_3(n) = \{\underset{\uparrow}{0}, 0, 0, 1\} \xleftrightarrow{\mathcal{T}} y_3(n) = \{1, \underset{\uparrow}{2}, 1\}$$

Can you draw any conclusions regarding the linearity of the system. What is the impulse response of the system?

2.11 The following input–output pairs have been observed during the operation of a linear system:

$$x_1(n) = \{-1, \underset{\uparrow}{2}, 1\} \xleftrightarrow{\mathcal{T}} y_1(n) = \{1, \underset{\uparrow}{2}, -1, 0, 1\}$$

$$x_2(n) = \{1, \underset{\uparrow}{-1}, -1\} \xleftrightarrow{\mathcal{T}} y_2(n) = \{-1, \underset{\uparrow}{1}, 0, 2\}$$

$$x_3(n) = \{0, \underset{\uparrow}{1}, 1\} \xleftrightarrow{\mathcal{T}} y_3(n) = \{1, \underset{\uparrow}{2}, 1\}$$

Can you draw any conclusions about the time invariance of this system?

2.12 The only available information about a system consists of N input–output pairs of signals $y_i(n) = \mathcal{T}[x_i(n)]$, $i = 1, 2, \ldots, N$.

(a) What is the class of input signals for which we can determine the output, using the information above, if the system is known to be linear?

(b) The same as above, if the system is known to be time invariant.

2.13 Show that the necessary and sufficient condition for a relaxed LTI system to be BIBO stable is

$$\sum_{n=-\infty}^{\infty} |h(n)| \leq M_h < \infty$$

for some constant M_h.

2.14 Show that:

(a) A relaxed linear system is causal if and only if for any input $x(n)$ such that

$$x(n) = 0 \text{ for } n < n_0 \Rightarrow y(n) = 0 \qquad \text{for } n < n_0$$

(b) A relaxed LTI system is causal if and only if

$$h(n) = 0 \qquad \text{for } n < 0$$

2.15

(a) Show that for any real or complex constant a, and any finite integer numbers M and N, we have

$$\sum_{n}^{N} a^n = Ma^n = \begin{cases} \dfrac{a^M - a^{N+1}}{1-a}, & \text{if } a \neq 1 \\ N - M + 1, & \text{if } a = 1 \end{cases}$$

(b) Show that if $|a| < 1$, then

$$\sum_{n=0}^{\infty} a^n = \frac{1}{1-a}$$

2.16 **(a)** If $y(n) = x(n) * h(n)$, show that $\sum_y = \sum_x \sum_h$, where $\sum_x = \sum_{n=-\infty}^{\infty} x(n)$.

(b) Compute the convolution $y(n) = x(n) * h(n)$ of the following signals and check the correctness of the results by using the test in (a).

(1) $x(n) = \{1, 2, 4\}, h(n) = \{1, 1, 1, 1, 1\}$

(2) $x(n) = \{1, 2, -1\}, h(n) = x(n)$

(3) $x(n) = \{0, 1, -2, 3, -4\}, h(n) = \{\frac{1}{2}, \frac{1}{2}, 1, \frac{1}{2}\}$

(4) $x(n) = \{1, 2, 3, 4, 5\}, h(n) = \{1\}$

(5) $x(n) = \{\underset{\uparrow}{1}, -2, 3\}, h(n) = \{\underset{\uparrow}{0}, 0, 1, 1, 1, 1\}$

(6) $x(n) = \{\underset{\uparrow}{0}, 0, 1, 1, 1, 1\}, h(n) = \{1, \underset{\uparrow}{-2}, 3\}$

(7) $x(n) = \{\underset{\uparrow}{0}, 1, 4, -3\}, h(n) = \{\underset{\uparrow}{1}, 0, -1, -1\}$

(8) $x(n) = \{\underset{\uparrow}{1}, 1, 2\}, h(n) = u(n)$

(9) $x(n) = \{1, 1, \underset{\uparrow}{0}, 1, 1\}, h(n) = \{1, -2, -3, \underset{\uparrow}{4}\}$

(10) $x(n) = \{1, 2, \underset{\uparrow}{0}, 2, 1\} h(n) = x(n)$

(11) $x(n) = (\frac{1}{2})^n u(n), h(n) = (\frac{1}{4})^n u(n)$

2.17 Compute and plot the convolutions $x(n) * h(n)$ and $h(n) * x(n)$ for the pairs of signal shown in Fig. P2.17.

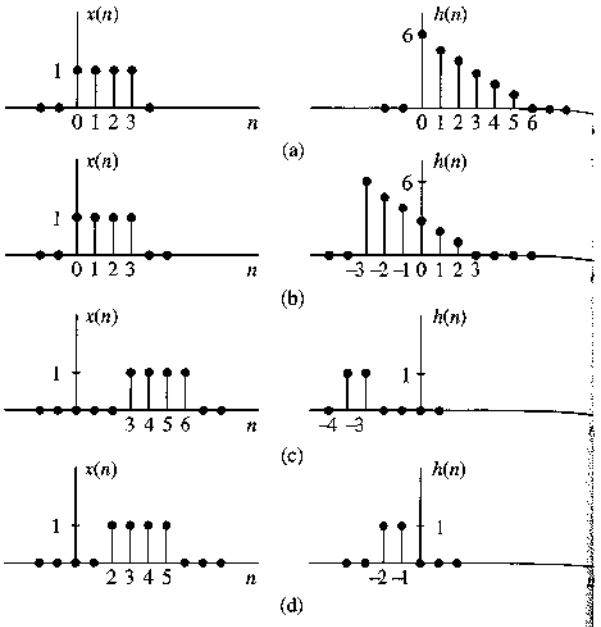


Figure P2.17

2.18 Determine and sketch the convolution $y(n)$ of the signals

$$x(n) = \begin{cases} \frac{1}{3}n, & 0 \le n \le 6 \\ 0, & \text{elsewhere} \end{cases}$$

$$h(n) = \begin{cases} 1, & -2 \le n \le 2 \\ 0, & \text{elsewhere} \end{cases}$$

(a) Graphically

(b) Analytically

2.19 Compute the convolution $y(n)$ of the signals

$$x(n) = \begin{cases} \alpha^n, & -3 \le n \le 5 \\ 0, & \text{elsewhere} \end{cases}$$

$$h(n) = \begin{cases} 1, & 0 \le n \le 4 \\ 0, & \text{elsewhere} \end{cases}$$

2.20 Consider the following three operations.

(a) Multiply the integer numbers: 131 and 122.

(b) Compute the convolution of signals: $\{1, 3, 1\} * \{1, 2, 2\}$.

(c) Multiply the polynomials: $1 + 3z + z^2$ and $1 + 2z + 2z^2$.

(d) Repeat part (a) for the numbers 1.31 and 12.2.

(e) Comment on your results.

2.21 Compute the convolution $y(n) = x(n) * h(n)$ of the following pairs of signals.

(a) $x(n) = a^n u(n)$, $h(n) = b^n u(n)$ when $a \neq b$ and when $a = b$

(b) $x(n) = \begin{cases} 1, & n = -2, 0, 1 \\ 2, & n = -1 \\ 0, & \text{elsewhere} \end{cases}$ $h(n) = \delta(n) - \delta(n-1) + \delta(n-4) + \delta(n-5)$

(c) $x(n) = u(n+1) - u(n-4) - \delta(n-5)$; $h(n) = [u(n+2) - u(n-3)] \cdot (3 - |n|)$

(d) $x(n) = u(n) - u(n-5)$; $h(n) = u(n-2) - u(n-8) + u(n-11) - u(n-17)$

2.22 Let $x(n)$ be the input signal to a discrete-time filter with impulse response $h_i(n)$ and let $y_i(n)$ be the corresponding output.

(a) Compute and sketch $x(n)$ and $y_i(n)$ in the following cases, using the same scale in all figures.

$$x(n) = \{1, 4, 2, 3, 5, 3, 3, 4, 5, 7, 6, 9\}$$

$$h_1(n) = \{1, 1\}$$

$$h_2(n) = \{1, 2, 1\}$$

$$h_3(n) = \{\frac{1}{2}, \frac{1}{2}\}$$

$$h_4(n) = \{\frac{1}{4}, \frac{1}{2}, \frac{1}{4}\}$$

$$h_5(n) = \{\frac{1}{4}, -\frac{1}{2}, \frac{1}{4}\}$$

Sketch $x(n)$, $y_1(n)$, $y_2(n)$ on one graph and $x(n)$, $y_3(n)$, $y_4(n)$, $y_5(n)$ on another graph.

(b) What is the difference between $y_1(n)$ and $y_2(n)$, and between $y_3(n)$ and $y_4(n)$?

(c) Comment on the smoothness of $y_2(n)$ and $y_4(n)$. Which factors affect the smoothness?

(d) Compare $y_4(n)$ with $y_5(n)$. What is the difference? Can you explain it?

(e) Let $h_6(n) = \{\frac{1}{2}, -\frac{1}{2}\}$. Compute $y_6(n)$. Sketch $x(n)$, $y_2(n)$, and $y_6(n)$ on the same figure and comment on the results.

2.23 Express the output $y(n)$ of a linear time-invariant system with impulse response $h(n)$ in terms of its step response $s(n) = h(n) * u(n)$ and the input $x(n)$.

2.24 The discrete-time system

$$y(n) = ny(n-1) + x(n), \qquad n \geq 0$$

is at rest [i.e., $y(-1) = 0$]. Check if the system is linear time invariant and BIBO stable.

2.25 Consider the signal $y(n) = a^n u(n)$, $0 < a < 1$.

(a) Show that any sequence $x(n)$ can be decomposed as

$$x(n) = \sum_{n=-\infty}^{\infty} c_k y(n-k)$$

and express c_k in terms of $x(n)$.

(b) Use the properties of linearity and time invariance to express the output $y(n) = \mathcal{T}[x(n)]$ in terms of the input $x(n)$ and the signal $g(n) = \mathcal{T}[y(n)]$, where $\mathcal{T}[\cdot]$ is an LTI system.

(c) Express the impulse response $h(n) = \mathcal{T}[\delta(n)]$ in terms of $g(n)$.

2.26 Determine the zero-input response of the system described by the second-order difference equation

$$x(n) - 3y(n-1) - 4y(n-2) = 0$$

2.27 Determine the particular solution of the difference equation

$$y(n) = \frac{5}{6}y(n-1) - \frac{1}{6}y(n-2) + x(n)$$

when the forcing function is $x(n) = 2^n u(n)$

2.28 In Example 2.4.8, equation (2.4.30), separate the output sequence $y(n)$ into the transient response and the steady-state response. Plot these two responses for $a_1 = -0.9$.

2.29 Determine the impulse response for the cascade of two linear time-invariant systems having impulse responses.

$$h_1(n) = a^n[u(n) - u(n-N)] \text{ and } h_2(n) = [u(n) - u(n-M)]$$

2.30 Determine the response $y(n)$, $n \geq 0$, of the system described by the second-order difference equation

$$y(n) - 3y(n-1) - 4y(n-2) = x(n) + 2x(n-1)$$

to the input $x(n) = 4^n u(n)$.

2.31 Determine the impulse response of the following causal system:

$$y(n) - 3y(n-1) - 4y(n-2) = x(n) + 2x(n-1)$$

2.32 Let $x(n)$, $N_1 \leq n \leq N_2$ and $h(n)$, $M_1 \leq n \leq M_2$ be two finite-duration signals.

(a) Determine the range $L_1 \leq n \leq L_2$ of their convolution, in terms of N_1, N_2, M_1 and M_2.

(b) Determine the limits of the cases of partial overlap from the left, full overlap, and partial overlap from the right. For convenience, assume that $h(n)$ has shorter duration than $x(n)$.

(c) Illustrate the validity of your results by computing the convolution of the signals

$$x(n) = \begin{cases} 1, & -2 \leq n \leq 4 \\ 0, & \text{elsewhere} \end{cases}$$

$$h(n) = \begin{cases} 2, & -1 \leq n \leq 2 \\ 0, & \text{elsewhere} \end{cases}$$

2.33 Determine the impulse response and the unit step response of the systems described by the difference equation

(a) $y(n) = 0.6y(n-1) - 0.08y(n-2) + x(n)$

(b) $y(n) = 0.7y(n-1) - 0.1y(n-2) + 2x(n) - x(n-2)$

2.34 Consider a system with impulse response

$$h(n) = \begin{cases} (\frac{1}{2})^n, & 0 \le n \le 4 \\ 0, & \text{elsewhere} \end{cases}$$

Determine the input $x(n)$ for $0 \le n \le 8$ that will generate the output sequence

$$y(n) = \{\underset{\uparrow}{1}, 2, 2.5, 3, 3, 3, 2, 1, 0, \ldots\}$$

2.35 Consider the interconnection of LTI systems as shown in Fig. P2.35.

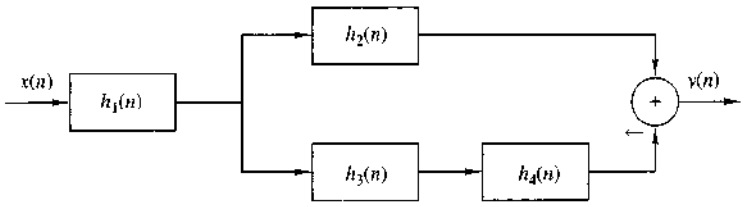


Figure P2.35

(a) Express the overall impulse response in terms of $h_1(n)$, $h_2(n)$, $h_3(n)$, and $h_4(n)$.

(b) Determine $h(n)$ when

$$h_1(n) = \{\frac{1}{2}, \frac{1}{4}, \frac{1}{2}\}$$

$$h_2(n) = h_3(n) = (n+1)u(n)$$

$$h_4(n) = \delta(n-2)$$

(c) Determine the response of the system in part (b) if

$$x(n) = \delta(n+2) + 3\delta(n-1) - 4\delta(n-3)$$

2.36 Consider the system in Fig. P2.36 with $h(n) = a^n u(n)$, $-1 < a < 1$. Determine the response $y(n)$ of the system to the excitation

$$x(n) = u(n+5) - u(n-10)$$

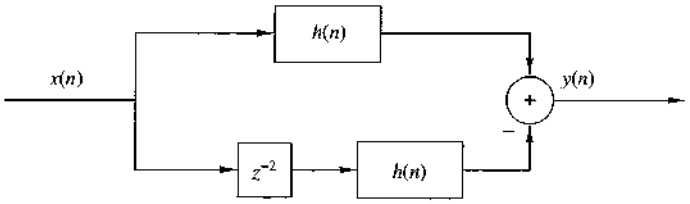


Figure P2.36

2.37 Compute and sketch the step response of the system

$$y(n) = \frac{1}{M}\sum_{k=0}^{M-1} x(n-k)$$

2.38 Determine the range of values of the parameter a for which the linear time-invariant system with impulse response

$$h(n) = \begin{cases} a^n, & n \geq 0, n \text{ even} \\ 0, & \text{otherwise} \end{cases}$$

is stable.

2.39 Determine the response of the system with impulse response

$$h(n) = a^n u(n)$$

to the input signal

$$x(n) = u(n) - u(n-10)$$

(*Hint*: The solution can be obtained easily and quickly by applying the linearity and time-invariance properties to the result in Example 2.3.5.)

2.40 Determine the response of the (relaxed) system characterized by the impulse response

$$h(n) = (\frac{1}{2})^n u(n)$$

to the input signal

$$x(n) = \begin{cases} 1, & 0 \leq n < 10 \\ 0, & \text{otherwise} \end{cases}$$

2.41 Determine the response of the (relaxed) system characterized by the impulse response

$$h(n) = (\frac{1}{2})^n u(n)$$

to the input signals

(a) $x(n) = 2^n u(n)$

(b) $x(n) = u(-n)$

2.42 Three systems with impulse responses $h_1(n) = \delta(n) - \delta(n-1)$, $h_2(n) = h(n)$, and $h_3(n) = u(n)$, are connected in cascade.

(a) What is the impulse response, $h_c(n)$, of the overall system?

(b) Does the order of the interconnection affect the overall system?

2.43 **(a)** Prove and explain graphically the difference between the relations

$$x(n)\delta(n-n_0) = x(n_0)\delta(n-n_0) \qquad \text{and} \qquad x(n) * \delta(n-n_0) = x(n-n_0)$$

(b) Show that a discrete-time system, which is described by a convolution summation, is LTI and relaxed.

(c) What is the impulse response of the system described by $y(n) = x(n-n_0)$?

2.44 Two signals $s(n)$ and $v(n)$ are related through the following difference equations.

$$s(n) + a_1 s(n-1) + \cdots + a_N s(n-N) = b_0 v(n)$$

Design the block diagram realization of:

(a) The system that generates $s(n)$ when excited by $v(n)$.

(b) The system that generates $v(n)$ when excited by $s(n)$.

(c) What is the impulse response of the cascade interconnection of systems in parts (a) and (b)?

2.45 Compute the zero-state response of the system described by the difference equation

$$y(n) + \frac{1}{2}y(n-1) = x(n) + 2x(n-2)$$

to the input

$$x(n) = \{1, 2, \underset{\uparrow}{3}, 4, 2, 1\}$$

by solving the difference equation recursively.

2.46 Determine the direct form II realization for each of the following LTI systems:

(a) $2y(n) + y(n-1) - 4y(n-3) = x(n) + 3x(n-5)$

(b) $y(n) = x(n) - x(n-1) + 2x(n-2) - 3x(n-4)$

2.47 Consider the discrete-time system shown in Fig. P2.47.

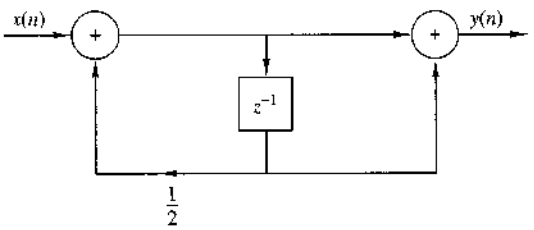


Figure P2.47

(a) Compute the 10 first samples of its impulse response.

(b) Find the input–output relation.

(c) Apply the input $x(n) = \{\underset{\uparrow}{1}, 1, 1, \ldots\}$ and compute the first 10 samples of the output.

(d) Compute the first 10 samples of the output for the input given in part (c) by using convolution.

(e) Is the system causal? Is it stable?

2.48 Consider the system described by the difference equation

$$y(n) = ay(n-1) + bx(n)$$

(a) Determine b in terms of a so that

$$\sum_{n=-\infty}^{\infty} h(n) = 1$$

(b) Compute the zero-state step response $s(n)$ of the system and choose b so that $s(\infty) = 1$.

(c) Compare the values of b obtained in parts (a) and (b). What did you notice?

2.49 A discrete-time system is realized by the structure shown in Fig. P2.49.

(a) Determine the impulse response.

(b) Determine a realization for its inverse system, that is, the system which produces $x(n)$ as an output when $y(n)$ is used as an input.

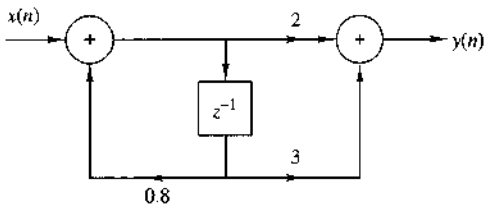


Figure P2.49

2.50 Consider the discrete-time system shown in Fig. P2.50.

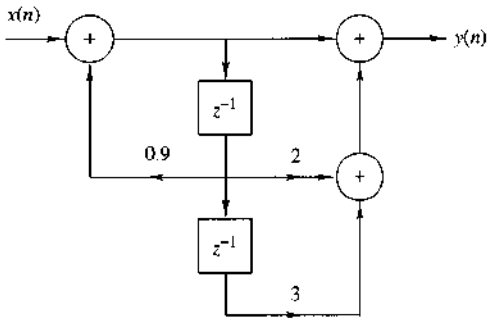


Figure P2.50

(a) Compute the first six values of the impulse response of the system.

(b) Compute the first six values of the zero-state step response of the system.

(c) Determine an analytical expression for the impulse response of the system.

2.51 Determine and sketch the impulse response of the following systems for $n = 0, 1, \ldots, 9$.

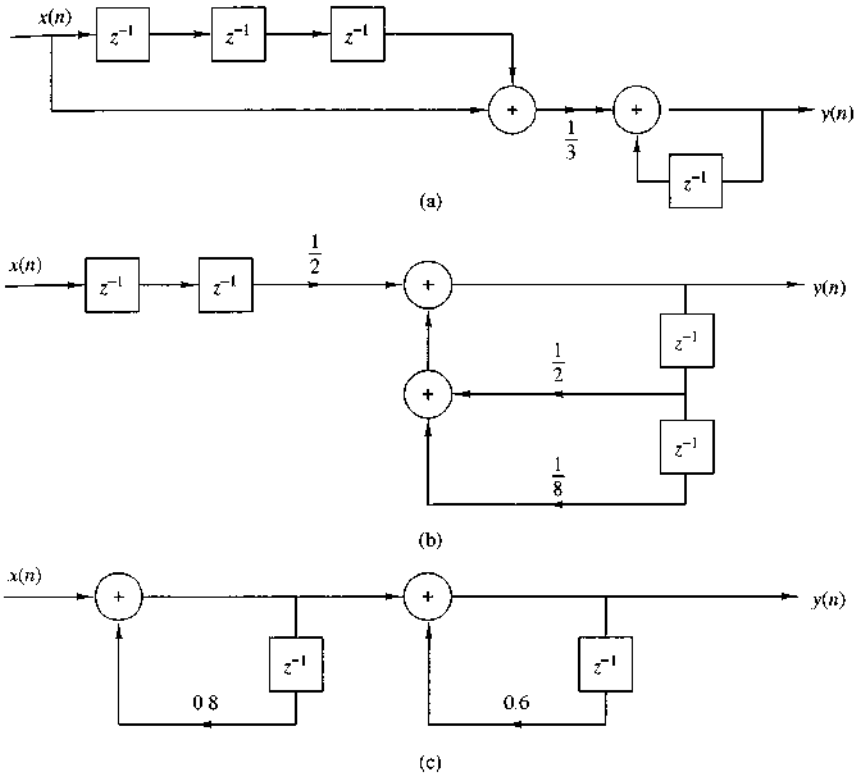


Figure P2.51

(a) Fig. P2.51(a).

(b) Fig. P2.51(b).

(c) Fig. P2.51(c).

(d) Classify the systems above as FIR or IIR.

(e) Find an explicit expression for the impulse response of the system in part (c).

2.52 Consider the systems shown in Fig. P2.52.

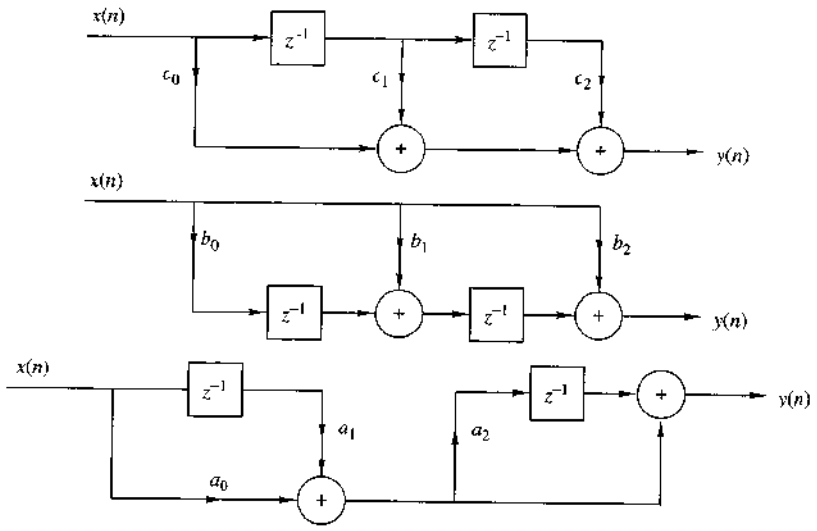


Figure P2.52

(a) Determine and sketch their impulse responses $h_1(n)$, $h_2(n)$, and $h_3(n)$.

(b) Is it possible to choose the coefficients of these systems in such a way that

$$h_1(n) = h_2(n) = h_3(n)$$

2.53 Consider the system shown in Fig. P2.53.

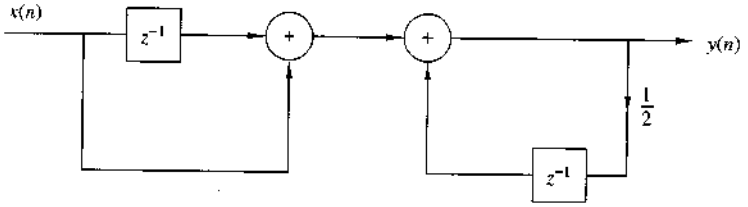


Figure P2.53

(a) Determine its impulse response $h(n)$.

(b) Show that $h(n)$ is equal to the convolution of the following signals:

$$h_1(n) = \delta(n) + \delta(n-1)$$

$$h_2(n) = (\frac{1}{2})^n u(n)$$

2.54 Compute and sketch the convolution $y_i(n)$ and correlation $r_i(n)$ sequences for th following pair of signals and comment on the results obtained.

(a) $x_1(n) = \{\underset{\uparrow}{1}, 2, 4\}$ $\quad h_1(n) = \{\underset{\uparrow}{1}, 1, 1, 1, 1\}$

(b) $x_2(n) = \{\underset{\uparrow}{0}, 1, -2, 3, -4\}$ $\quad h_2(n) = \{\frac{1}{2}, 1, \underset{\uparrow}{2}, 1, \frac{1}{2}\}$

(c) $x_3(n) = \{\underset{\uparrow}{1}, 2, 3, 4\}$ $\quad h_3(n) = \{\underset{\uparrow}{4}, 3, 2, 1\}$

(d) $x_4(n) = \{\underset{\uparrow}{1}, 2, 3, 4\}$ $\quad h_4(n) = \{\underset{\uparrow}{1}, 2, 3, 4\}$

2.55 The zero-state response of a causal LTI system to the input $x(n) = \{\underset{\uparrow}{1}, 3, 3, 1\}$ is $y(n) = \{\underset{\uparrow}{1}, 4, 6, 4, 1\}$. Determine its impulse response.

2.56 Prove by direct substitution the equivalence of equations (2.5.9) and (2.5.10), which describe the direct form II structure, to the relation (2.5.6), which describes the direct form I structure.

2.57 Determine the response $y(n)$, $n \geq 0$ of the system described by the second-order difference equation

$$y(n) - 4y(n-1) + 4y(n-2) = x(n) - x(n-1)$$

when the input is

$$x(n) = (-1)^n u(n)$$

and the initial conditions are $y(-1) = y(-2) = 0$.

2.58 Determine the impulse response $h(n)$ for the system described by the second-order difference equation

$$y(n) - 4y(n-1) + 4y(n-2) = x(n) - x(n-1)$$

2.59 Show that any discrete-time signal $x(n)$ can be expressed as

$$x(n) = \sum_{k=-\infty}^{\infty} [x(k) - x(k-1)]u(n-k)$$

where $u(n-k)$ is a unit step delayed by k units in time, that is,

$$u(n-k) = \begin{cases} 1, & n \geq k \\ 0, & \text{otherwise} \end{cases}$$

2.60 Show that the output of an LTI system can be expressed in terms of its unit step response $s(n)$ as follows.

$$y(n) = \sum_{k=-\infty}^{\infty} [s(k) - s(k-1)]x(n-k)$$

$$= \sum_{k=-\infty}^{\infty} [x(k) - x(k-1)]s(n-k)$$

2.61 Compute the correlation sequences $r_{xx}(l)$ and $r_{xy}(l)$ for the following signal sequences.

$$x(n) = \begin{cases} 1, & n_0 - N \leq n \leq n_0 + N \\ 0, & \text{otherwise} \end{cases}$$

$$y(n) = \begin{cases} 1, & -N \leq n \leq N \\ 0, & \text{otherwise} \end{cases}$$

2.62 Determine the autocorrelation sequences of the following signals.

(a) $x(n) = \{\underset{\uparrow}{1}, 2, 1, 1\}$

(b) $y(n) = \{\underset{\uparrow}{1}, 1, 2, 1\}$

What is your conclusion?

2.63 What is the normalized autocorrelation sequence of the signal $x(n)$ given by

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

2.64 An audio signal $s(t)$ generated by a loudspeaker is reflected at two different walls with reflection coefficients r_1 and r_2. The signal $x(t)$ recorded by a microphone close to the loudspeaker, after sampling, is

$$x(n) = s(n) + r_1 s(n - k_1) + r_2 s(n - k_2)$$

where k_1 and k_2 are the delays of the two echoes.

(a) Determine the autocorrelation $r_{xx}(l)$ of the signal $x(n)$.

(b) Can we obtain r_1, r_2, k_1, and k_2 by observing $r_{xx}(l)$?

(c) What happens if $r_2 = 0$?

2.65 *Time-delay estimation in radar* Let $x_a(t)$ be the transmitted signal and $y_a(t)$ be the received signal in a radar system, where

$$y_a(t) = a x_a(t - t_d) + v_a(t)$$

and $v_a(t)$ is additive random noise. The signals $x_a(t)$ and $y_a(t)$ are sampled in the receiver, according to the sampling theorem, and are processed digitally to determine the time delay and hence the distance of the object. The resulting discrete-time signals are

$$\begin{aligned} x(n) &= x_a(nT) \\ y(n) &= y_a(nT) = a x_a(nT - DT) + v_a(nT) \\ &\triangleq a x(n - D) + v(n) \end{aligned}$$

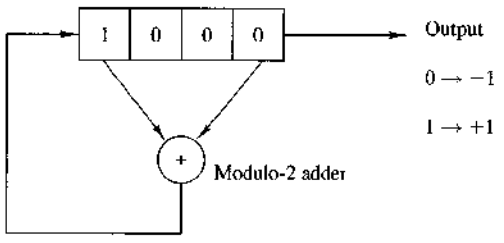


Figure P2.65 Linear feedback shift register.

(a) Explain how we can measure the delay D by computing the crosscorrelation $r_{xy}(l)$.

(b) Let $x(n)$ be the 13-point *Barker sequence*

$$x(n) = \{+1, +1, +1, +1, +1, -1, -1, +1, +1, -1, +1, -1, +1\}$$

and $v(n)$ be a Gaussian random sequence with zero mean and variance $\sigma^2 = 0.01$. Write a program that generates the sequence $y(n)$, $0 \le n \le 199$ for $a = 0.9$ and $D = 20$. Plot the signals $x(n)$, $y(n)$, $0 \le n \le 199$.

(c) Compute and plot the crosscorrelation $r_{xy}(l)$, $0 \le l \le 59$. Use the plot to estimate the value of the delay D.

(d) Repeat parts (b) and (c) for $\sigma^2 = 0.1$ and $\sigma^2 = 1$.

(e) Repeat parts (b) and (c) for the signal sequence

$$x(n) = \{-1, -1, -1, +1, +1, +1, +1, -1, +1, -1, +1, +1, -1, -1, +1\}$$

which is obtained from the four-stage feedback shift register shown in Fig. P2.65. Note that $x(n)$ is just one period of the periodic sequence obtained from the feedback shift register.

(f) Repeat parts (b) and (c) for a sequence of period $N = 2^7 - 1$, which is obtained from a seven-stage feedback shift register. Table 2.2 gives the stages connected to the modulo-2 adder for (maximal-length) shift-register sequences of length $N = 2^m - 1$.

TABLE 2.2 Shift-Register Connections for Generating Maximal-Length Sequences

m	Stages Connected to Modulo-2 Adder
1	1
2	1, 2
3	1, 3
4	1, 4
5	1, 4
6	1, 6
7	1, 7
8	1, 5, 6, 7
9	1, 6
10	1, 8
11	1, 10
12	1, 7, 9, 12
13	1, 10, 11, 13
14	1, 5, 9, 14
15	1, 15
16	1, 5, 14, 16
17	1, 15

2.66 *Implementation of LTI systems* Consider the recursive discrete-time system described by the difference equation

$$y(n) = -a_1 y(n-1) - a_2 y(n-2) + b_0 x(n)$$

where $a_1 = -0.8$, $a_2 = 0.64$, and $b_0 = 0.866$.

(a) Write a program to compute and plot the impulse response $h(n)$ of the system for $0 \le n \le 49$.

(b) Write a program to compute and plot the zero-state step response $s(n)$ of the system for $0 \le n \le 100$.

(c) Define an FIR system with impulse response $h_{\text{FIR}}(n)$ given by

$$h_{\text{FIR}}(n) = \begin{cases} h(n), & 0 \le n \le 19 \\ 0, & \text{elsewhere} \end{cases}$$

where $h(n)$ is the impulse response computed in part (a). Write a program to compute and plot its step response.

(d) Compare the results obtained in parts (b) and (c) and explain their similarities and differences.

2.67 Write a computer program that computes the overall impulse response $h(n)$ of the system shown in Fig. P2.67 for $0 \le n \le 99$. The systems $\mathcal{T}_1$, $\mathcal{T}_2$, $\mathcal{T}_3$, and $\mathcal{T}_4$ are specified by

$$\mathcal{T}_1 : h_1(n) = \{1, \underset{\uparrow}{\frac{1}{2}}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}\}$$

$$\mathcal{T}_2 : h_2(n) = \{\underset{\uparrow}{1}, 1, 1, 1, 1\}$$

$$\mathcal{T}_3 : y_3(n) = \frac{1}{4}x(n) + \frac{1}{2}x(n-1) + \frac{1}{4}x(n-2)$$

$$\mathcal{T}_4 : y(n) = 0.9y(n-1) - 0.81y(n-2) + v(n) + v(n-1)$$

Plot $h(n)$ for $0 \le n \le 99$.

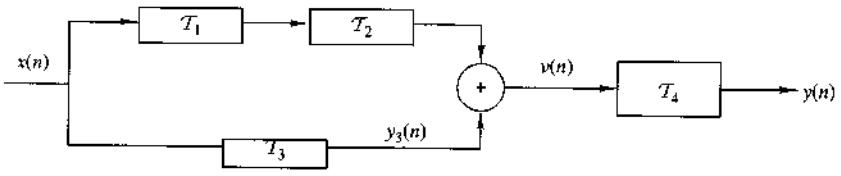


Figure P2.67

CHAPTER

3

The z-Transform and Its Application to the Analysis of LTI Systems

Transform techniques are an important tool in the analysis of signals and linear time-invariant (LTI) systems. In this chapter we introduce the z-transform, develop its properties, and demonstrate its importance in the analysis and characterization of linear time-invariant systems.

The z-transform plays the same role in the analysis of discrete-time signals and LTI systems as the Laplace transform does in the analysis of continuous-time signals and LTI systems. For example, we shall see that in the z-domain (complex z-plane) the convolution of two time-domain signals is equivalent to multiplication of their corresponding z-transforms. This property greatly simplifies the analysis of the response of an LTI system to various signals. In addition, the z-transform provides us with a means of characterizing an LTI system, and its response to various signals, by its pole–zero locations.

We begin this chapter by defining the z-transform. Its important properties are presented in Section 3.2. In Section 3.3 the transform is used to characterize signals in terms of their pole–zero patterns. Section 3.4 describes methods for inverting the z-transform of a signal so as to obtain the time-domain representation of the signal. Section 3.5 is focused on the use of the z-transform in the analysis of LTI systems. Finally, in Section 3.6, we treat the one-sided z-transform and use it to solve linear difference equations with nonzero initial conditions.

3.1 The z-Transform

In this section we introduce the z-transform of a discrete-time signal, investigate its convergence properties, and briefly discuss the inverse z-transform.

3.1.1 The Direct z-Transform

The z-transform of a discrete-time signal $x(n)$ is defined as the power series

$$X(z) \equiv \sum_{n=-\infty}^{\infty} x(n)z^{-n} \tag{3.1.1}$$

where z is a complex variable. The relation (3.1.1) is sometimes called the *direct z-transform* because it transforms the time-domain signal $x(n)$ into its complex-plane representation $X(z)$. The inverse procedure [i.e., obtaining $x(n)$ from $X(z)$] is called the *inverse z-transform* and is examined briefly in Section 3.1.2 and in more detail in Section 3.4.

For convenience, the z-transform of a signal $x(n)$ is denoted by

$$X(z) \equiv Z\{x(n)\} \tag{3.1.2}$$

whereas the relationship between $x(n)$ and $X(z)$ is indicated by

$$x(n) \overset{z}{\longleftrightarrow} X(z) \tag{3.1.3}$$

Since the z-transform is an infinite power series, it exists only for those values of z for which this series converges. The *region of convergence* (ROC) of $X(z)$ is the set of all values of z for which $X(z)$ attains a finite value. Thus any time we cite a z-transform we should also indicate its ROC.

We illustrate these concepts by some simple examples.

EXAMPLE 3.1.1

Determine the z-transforms of the following *finite-duration* signals.

(a) $x_1(n) = \{\underset{\uparrow}{1}, 2, 5, 7, 0, 1\}$

(b) $x_2(n) = \{1, 2, \underset{\uparrow}{5}, 7, 0, 1\}$

(c) $x_3(n) = \{\underset{\uparrow}{0}, 0, 1, 2, 5, 7, 0, 1\}$

(d) $x_4(n) = \{2, 4, \underset{\uparrow}{5}, 7, 0, 1\}$

(e) $x_5(n) = \delta(n)$

(f) $x_6(n) = \delta(n-k), k > 0$

(g) $x_7(n) = \delta(n+k), k > 0$

Solution. From definition (3.1.1), we have

(a) $X_1(z) = 1 + 2z^{-1} + 5z^{-2} + 7z^{-3} + z^{-5}$, ROC: entire z-plane except $z = 0$

(b) $X_2(z) = z^2 + 2z + 5 + 7z^{-1} + z^{-3}$, ROC: entire z-plane except $z = 0$ and $z = \infty$

(c) $X_3(z) = z^{-2} + 2z^{-3} + 5z^{-4} + 7z^{-5} + z^{-7}$, ROC: entire z-plane except $z = 0$

(d) $X_4(z) = 2z^2 + 4z + 5 + 7z^{-1} + z^{-3}$, ROC: entire z-plane except $z = 0$ and $z = \infty$

(e) $X_5(z) = 1$ [i.e., $\delta(n) \overset{z}{\longleftrightarrow} 1$], ROC: entire z-plane

(f) $X_6(z) = z^{-k}$ [i.e., $\delta(n-k) \overset{z}{\longleftrightarrow} z^{-k}$], $k > 0$, ROC: entire z-plane except $z = 0$

(g) $X_7(z) = z^k$ [i.e., $\delta(n+k) \overset{z}{\longleftrightarrow} z^k$], $k > 0$, ROC: entire z-plane except $z = \infty$

From this example it is easily seen that the ROC of a *finite-duration signal* is the entire z-plane, except possibly the points $z = 0$ and/or $z = \infty$. These points are excluded, because z^k $(k > 0)$ becomes unbounded for $z = \infty$ and z^{-k} $(k > 0)$ becomes unbounded for $z = 0$.

From a mathematical point of view the z-transform is simply an alternative representation of a signal. This is nicely illustrated in Example 3.1.1, where we see that the coefficient of z^{-n}, in a given transform, is the value of the signal at time n. In other words, the exponent of z contains the time information we need to identify the samples of the signal.

In many cases we can express the sum of the finite or infinite series for the z-transform in a closed-form expression. In such cases the z-transform offers a compact alternative representation of the signal.

EXAMPLE 3.1.2

Determine the z-transform of the signal

$$x(n) = (\frac{1}{2})^n u(n)$$

Solution. The signal $x(n)$ consists of an infinite number of nonzero values

$$x(n) = \{1, (\frac{1}{2}), (\frac{1}{2})^2, (\frac{1}{2})^3, \ldots, (\frac{1}{2})^n, \ldots\}$$

The z-transform of $x(n)$ is the infinite power series

$$X(z) = 1 + \frac{1}{2}z^{-1} + (\frac{1}{2})^2 z^{-2} + (\frac{1}{2})^n z^{-n} + \cdots$$

$$= \sum_{n=0}^{\infty} (\frac{1}{2})^n z^{-n} = \sum_{n=0}^{\infty} (\frac{1}{2}z^{-1})^n$$

This is an infinite geometric series. We recall that

$$1 + A + A^2 + A^3 + \cdots = \frac{1}{1 - A} \qquad \text{if } |A| < 1$$

Consequently, for $|\frac{1}{2}z^{-1}| < 1$, or equivalently, for $|z| > \frac{1}{2}$, $X(z)$ converges to

$$X(z) = \frac{1}{1 - \frac{1}{2}z^{-1}}, \qquad \text{ROC: } |z| > \frac{1}{2}$$

We see that in this case, the z-transform provides a compact alternative representation of the signal $x(n)$.

Let us express the complex variable z in polar form as

$$z = re^{j\theta} \tag{3.1.4}$$

where $r = |z|$ and $\theta = \measuredangle z$. Then $X(z)$ can be expressed as

$$X(z)|_{z=re^{j\theta}} = \sum_{n=-\infty}^{\infty} x(n)r^{-n}e^{-j\theta n}$$

In the ROC of $X(z)$, $|X(z)| < \infty$. But

$$\begin{aligned} |X(z)| &= \left| \sum_{n=-\infty}^{\infty} x(n)r^{-n}e^{-j\theta n} \right| \\ &\leq \sum_{n=-\infty}^{\infty} |x(n)r^{-n}e^{-j\theta n}| = \sum_{n=-\infty}^{\infty} |x(n)r^{-n}| \end{aligned} \tag{3.1.5}$$

Hence $|X(z)|$ is finite if the sequence $x(n)r^{-n}$ is absolutely summable.

The problem of finding the ROC for $X(z)$ is equivalent to determining the range of values of r for which the sequence $x(n)r^{-n}$ is absolutely summable. To elaborate, let us express (3.1.5) as

$$\begin{aligned} |X(z)| &\leq \sum_{n=-\infty}^{-1} |x(n)r^{-n}| + \sum_{n=0}^{\infty} \left| \frac{x(n)}{r^n} \right| \\ &\leq \sum_{n=1}^{\infty} |x(-n)r^{n}| + \sum_{n=0}^{\infty} \left| \frac{x(n)}{r^n} \right| \end{aligned} \tag{3.1.6}$$

If $X(z)$ converges in some region of the complex plane, both summations in (3.1.6) must be finite in that region. If the first sum in (3.1.6) converges, there must exist values of r small enough such that the product sequence $x(-n)r^n$, $1 \leq n < \infty$, is absolutely summable. Therefore, the ROC for the first sum consists of all points in a circle of some radius r_1, where $r_1 < \infty$, as illustrated in Fig. 3.1.1(a). On the other hand, if the second sum in (3.1.6) converges, there must exist values of r large enough such that the product sequence $x(n)/r^n$, $0 \leq n < \infty$, is absolutely summable. Hence the ROC for the second sum in (3.1.6) consists of all points outside a circle of radius $r > r_2$, as illustrated in Fig. 3.1.1(b).

Since the convergence of $X(z)$ requires that both sums in (3.1.6) be finite, it follows that the ROC of $X(z)$ is generally specified as the annular region in the z-plane, $r_2 < r < r_1$, which is the common region where both sums are finite. This region is illustrated in Fig. 3.1.1(c). On the other hand, if $r_2 > r_1$, there is no common region of convergence for the two sums and hence $X(z)$ does not exist.

The following examples illustrate these important concepts.

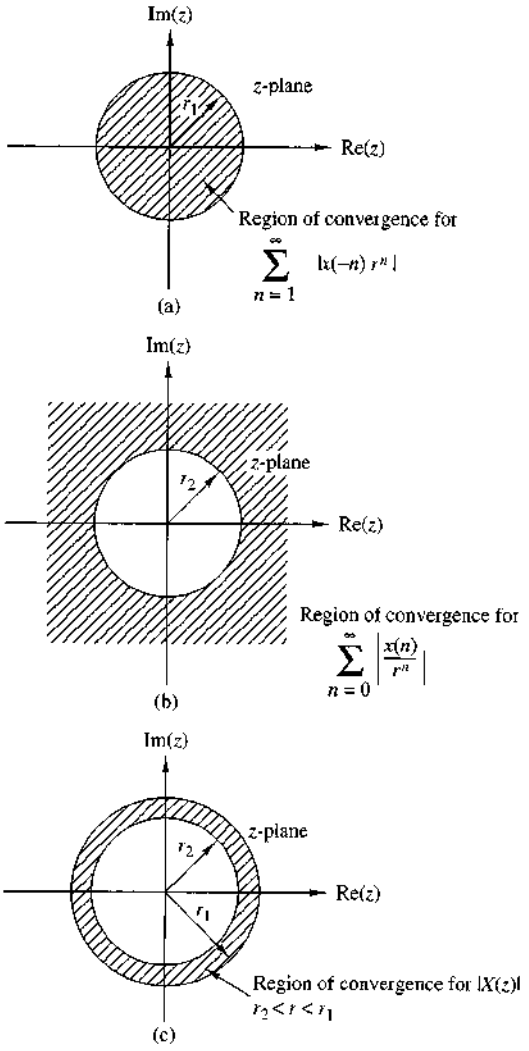


Figure 3.1.1
Region of convergence for $X(z)$ and its corresponding causal and anticausal components

EXAMPLE 3.1.3

Determine the z-transform of the signal

$$x(n) = \alpha^n u(n) = \begin{cases} \alpha^n, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

Solution. From the definition (3.1.1) we have

$$X(z) = \sum_{n=0}^{\infty} \alpha^n z^{-n} = \sum_{n=0}^{\infty} (\alpha z^{-1})^n$$

If $|\alpha z^{-1}| < 1$ or equivalently, $|z| > |\alpha|$, this power series converges to $1/(1 - \alpha z^{-1})$ Thus we have the z-transform pair

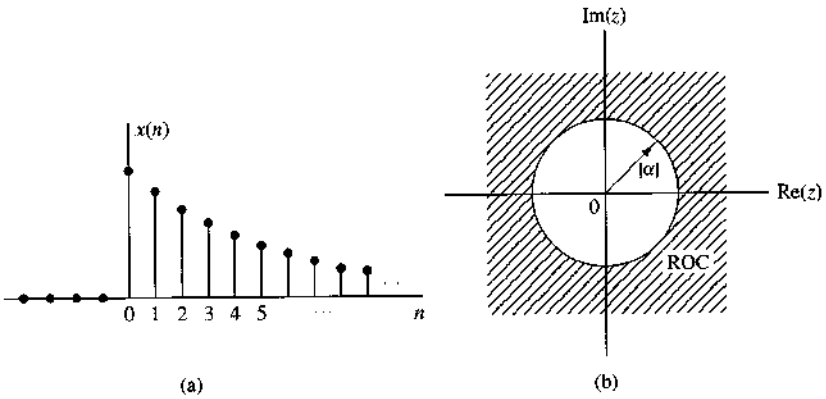


Figure 3.1.2 The exponential signal $x(n) = \alpha^n u(n)$ (a), and the ROC of its z-transform (b).

$$x(n) = \alpha^n u(n) \overset{z}{\longleftrightarrow} X(z) = \frac{1}{1 - \alpha z^{-1}}, \qquad \text{ROC: } |z| > |\alpha| \tag{3.1.7}$$

The ROC is the exterior of a circle having radius $|\alpha|$. Figure 3.1.2 shows a graph of the signal $x(n)$ and its corresponding ROC. Note that, in general, α need not be real.

If we set $\alpha = 1$ in (3.1.7), we obtain the z-transform of the unit step signal

$$x(n) = u(n) \overset{z}{\longleftrightarrow} X(z) = \frac{1}{1 - z^{-1}}, \qquad \text{ROC: } |z| > 1 \tag{3.1.8}$$

EXAMPLE 3.1.4

Determine the z-transform of the signal

$$x(n) = -\alpha^n u(-n-1) = \begin{cases} 0, & n \geq 0 \\ -\alpha^n, & n \leq -1 \end{cases}$$

Solution. From the definition (3.1.1) we have

$$X(z) = \sum_{n=-\infty}^{-1} (-\alpha^n) z^{-n} = -\sum_{l=1}^{\infty} (\alpha^{-1} z)^l$$

where $l = -n$. Using the formula

$$A + A^2 + A^3 + \cdots = A(1 + A + A^2 + \cdots) = \frac{A}{1 - A}$$

when $|A| < 1$ gives

$$X(z) = -\frac{\alpha^{-1} z}{1 - \alpha^{-1} z} = \frac{1}{1 - \alpha z^{-1}}$$

provided that $|\alpha^{-1} z| < 1$ or, equivalently, $|z| < |\alpha|$. Thus

$$x(n) = -\alpha^n u(-n-1) \overset{z}{\longleftrightarrow} X(z) = -\frac{1}{1 - \alpha z^{-1}}, \qquad \text{ROC: } |z| < |\alpha| \tag{3.1.9}$$

The ROC is now the interior of a circle having radius $|\alpha|$. This is shown in Fig. 3.1.3.

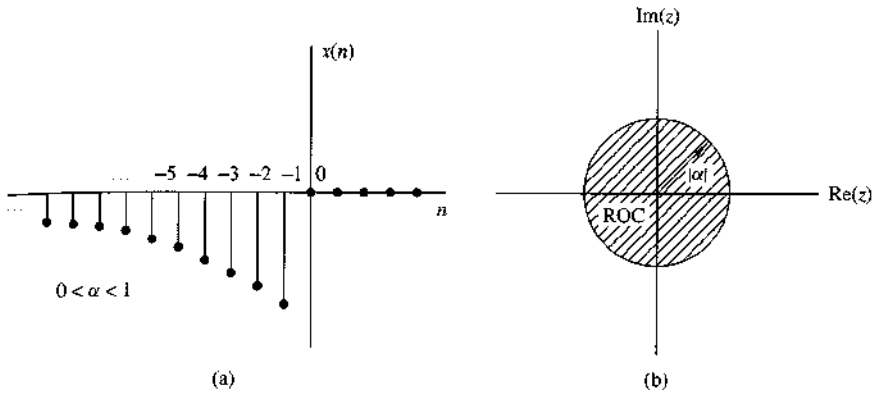


Figure 3.1.3 Anticausal signal $x(n) = -\alpha^n u(-n-1)$ (a), and the ROC of its z-transform (b).

Examples 3.1.3 and 3.1.4 illustrate two very important issues. The first concerns the uniqueness of the z-transform. From (3.1.7) and (3.1.9) we see that the causal signal $\alpha^n u(n)$ and the anticausal signal $-\alpha^n u(-n-1)$ have identical closed-form expressions for the z-transform, that is,

$$Z\{\alpha^n u(n)\} = Z\{-\alpha n u(-n-1)\} = \frac{1}{1-\alpha z^{-1}}$$

This implies that a closed-form expression for the z-transform does not uniquely specify the signal in the time domain. The ambiguity can be resolved only if in addition to the closed-form expression, the ROC is specified. In summary, *a discrete-time signal $x(n)$ is uniquely determined by its z-transform $X(z)$ and the region of convergence of $X(z)$.* In this text the term "z-transform" is used to refer to both the closed-form expression and the corresponding ROC. Example 3.1.3 also illustrates the point that *the ROC of a causal signal is the exterior of a circle of some radius r_2 while the ROC of an anticausal signal is the interior of a circle of some radius r_1.* The following example considers a sequence that is nonzero for $-\infty < n < \infty$.

EXAMPLE 3.1.5

Determine the z-transform of the signal

$$x(n) = \alpha^n u(n) + b^n u(-n-1)$$

Solution. From definition (3.1.1) we have

$$X(z) = \sum_{n=0}^{\infty} \alpha^n z^{-n} + \sum_{n=-\infty}^{-1} b^n z^{-n} = \sum_{n=0}^{\infty} (\alpha z^{-1})^n + \sum_{l=1}^{\infty} (b^{-1} z)^l$$

The first power series converges if $|\alpha z^{-1}| < 1$ or $|z| > |\alpha|$. The second power series converges if $|b^{-1}z| < 1$ or $|z| < |b|$.

In determining the convergence of $X(z)$, we consider two different cases.

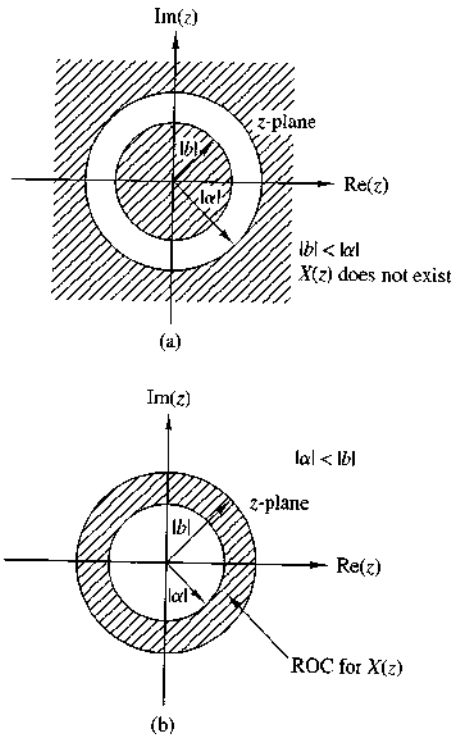


Figure 3.1.4
ROC for z-transform in Example 3.1.5.

Case 1 $|b| < |\alpha|$: In this case the two ROC above do not overlap, as shown in Fig. 3.1.4(a). Consequently, we cannot find values of z for which both power series converge simultaneously. Clearly, in this case, $X(z)$ does not exist.

Case 2 $|b| > |\alpha|$: In this case there is a ring in the z-plane where both power series converge simultaneously, as shown in Fig. 3.1.4(b). Then we obtain

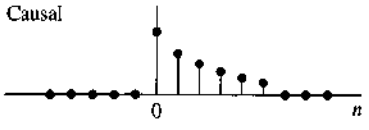
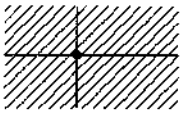
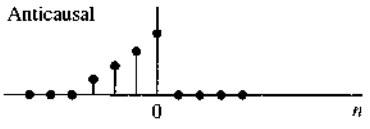
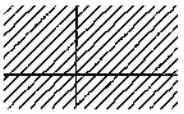
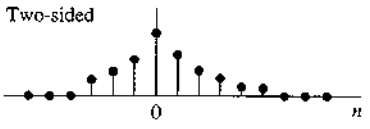
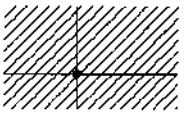
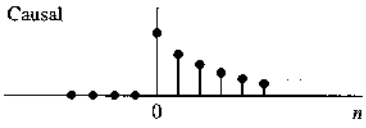
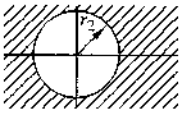
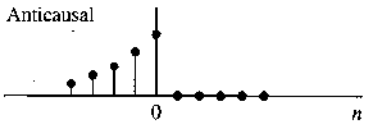
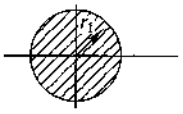
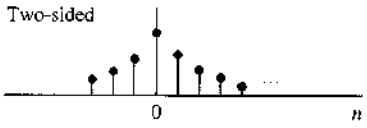
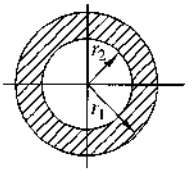
$$\begin{aligned} X(z) &= \frac{1}{1-\alpha z^{-1}} - \frac{1}{1-bz^{-1}} \\ &= \frac{b-\alpha}{\alpha + b - z - \alpha b z^{-1}} \end{aligned} \tag{3.1.10}$$

The ROC of $X(z)$ is $|\alpha| < |z| < |b|$.

This example shows that *if there is a ROC for an infinite-duration two-sided signal, it is a ring (annular region) in the z-plane.* From Examples 3.1.1, 3.1.3, 3.1.4, and 3.1.5, we see that the ROC of a signal depends both on its duration (finite or infinite) and on whether it is causal, anticausal, or two-sided. These facts are summarized in Table 3.1.

One special case of a two-sided signal is a signal that has infinite duration on the right side but not on the left [i.e., $x(n) = 0$ for $n < n_0 < 0$]. A second case is

TABLE 3.1 Characteristic Families of Signals with Their Corresponding ROCs

Signal	ROC
Finite-Duration Signals	
Causal 	 Entire z-plane except $z = 0$
Anticausal 	 Entire z-plane except $z = \infty$
Two-sided 	 Entire z-plane except $z = 0$ and $z = \infty$
Infinite-Duration Signals	
Causal 	 $\lvert z\rvert > r_2$
Anticausal 	 $\lvert z\rvert < r_1$
Two-sided 	 $r_2 < \lvert z\rvert < r_1$

a signal that has infinite duration on the left side but not on the right [i.e., $x(n) = 0$ for $n > n_1 > 0$]. A third special case is a signal that has finite duration on both the left and right sides [i.e., $x(n) = 0$ for $n < n_0 < 0$ and $n > n_1 > 0$]. These types of signals are sometimes called *right-sided, left-sided,* and *finite-duration two-sided* signals, respectively. The determination of the ROC for these three types of signals is left as an exercise for the reader (Problem 3.5).

Finally, we note that the z-transform defined by (3.1.1) is sometimes referred to as the *two-sided* or *bilateral z-transform*, to distinguish it from the *one-sided* or

unilateral z-transform given by

$$X^+(z) = \sum_{n=0}^{\infty} x(n)z^{-n} \tag{3.1.11}$$

The one-sided z-transform is examined in Section 3.6. In this text we use the expression z-transform exclusively to mean the two-sided z-transform defined by (3.1.1). The term "two-sided" will be used only in cases where we want to resolve any ambiguities. Clearly, if $x(n)$ is causal [i.e., $x(n) = 0$ for $n < 0$], the one-sided and two-sided z-transforms are identical. In any other case, they are different.

3.1.2 The Inverse z-Transform

Often, we have the z-transform $X(z)$ of a signal and we must determine the signal sequence. The procedure for transforming from the z-domain to the time domain is called the *inverse z-transform*. An inversion formula for obtaining $x(n)$ from $X(z)$ can be derived by using the *Cauchy integral theorem*, which is an important theorem in the theory of complex variables.

To begin, we have the z-transform defined by (3.1.1) as

$$X(z) = \sum_{k=-\infty}^{\infty} x(k)z^{-k} \tag{3.1.12}$$

Suppose that we multiply both sides of (3.1.12) by z^{n-1} and integrate both sides over a closed contour within the ROC of $X(z)$ which encloses the origin. Such a contour is illustrated in Fig. 3.1.5. Thus we have

$$\oint_C X(z)z^{n-1}\,dz = \oint_C \sum_{k=-\infty}^{\infty} x(k)z^{n-1-k}\,dz \tag{3.1.13}$$

where C denotes the closed contour in the ROC of $X(z)$, taken in a counterclockwise direction. Since the series converges on this contour, we can interchange the order of

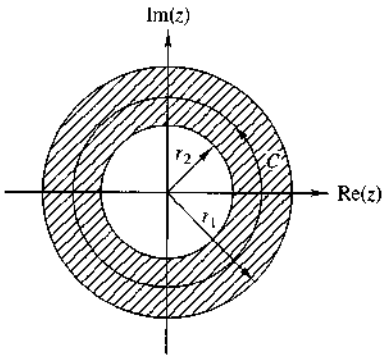


Figure 3.1.5
Contour C for integral in (3.1.13).

integration and summation on the right-hand side of (3.1.13). Thus (3.1.13) becomes

$$\oint_C X(z)z^{n-1}dz = \sum_{k=-\infty}^{\infty} x(k) \oint_C z^{n-1-k}dz \tag{3.1.14}$$

Now we can invoke the Cauchy integral theorem, which states that

$$\frac{1}{2\pi j} \oint_C z^{n-1-k}\, dz = \begin{cases} 1, & k = n \\ 0, & k \neq n \end{cases} \tag{3.1.15}$$

where C is any contour that encloses the origin. By applying (3.1.15), the right-hand side of (3.1.14) reduces to $2\pi j x(n)$ and hence the desired inversion formula

$$x(n) = \frac{1}{2\pi j} \oint_C X(z)z^{n-1}\, dz \tag{3.1.16}$$

Although the contour integral in (3.1.16) provides the desired inversion formula for determining the sequence $x(n)$ from the z-transform, we shall not use (3.1.16) directly in our evaluation of inverse z-transforms. In our treatment we deal with signals and systems in the z-domain which have rational z-transforms (i.e., z-transforms that are a ratio of two polynomials). For such z-transforms we develop a simpler method for inversion that stems from (3.1.16) and employs a table lookup.

3.2 Properties of the z-Transform

The z-transform is a very powerful tool for the study of discrete-time signals and systems. The power of this transform is a consequence of some very important properties that the transform possesses. In this section we examine some of these properties.

In the treatment that follows, it should be remembered that when we combine several z-transforms, the ROC of the overall transform is, at least, the intersection of the ROC of the individual transforms. This will become more apparent later, when we discuss specific examples.

Linearity. If

$$x_1(n) \overset{z}{\longleftrightarrow} X_1(z)$$

and

$$x_2(n) \overset{z}{\longleftrightarrow} X_2(z)$$

then

$$x(n) = a_1x_1(n) + a_2x_2(n) \overset{z}{\longleftrightarrow} X(z) = a_1X_1(z) + a_2X_2(z) \tag{3.2.1}$$

for any constants a_1 and a_2. The proof of this property follows immediately from the definition of linearity and is left as an exercise for the reader.

The linearity property can easily be generalized for an arbitrary number of signals. Basically, it implies that the z-transform of a linear combination of signals is the same linear combination of their z-transforms. Thus the linearity property helps us to find the z-transform of a signal by expressing the signal as a sum of elementary signals, for each of which, the z-transform is already known.

EXAMPLE 3.2.1

Determine the z-transform and the ROC of the signal

$$x(n) = [3(2^n) - 4(3^n)]u(n)$$

Solution. If we define the signals

$$x_1(n) = 2^n u(n)$$

and

$$x_2(n) = 3^n u(n)$$

then $x(n)$ can be written as

$$x(n) = 3x_1(n) - 4x_2(n)$$

According to (3.2.1), its z-transform is

$$X(z) = 3X_1(z) - 4X_2(z)$$

From (3.1.7) we recall that

$$\alpha^n u(n) \overset{z}{\longleftrightarrow} \frac{1}{1 - \alpha z^{-1}}, \qquad \text{ROC: } |z| > |\alpha| \tag{3.2.2}$$

By setting $\alpha = 2$ and $\alpha = 3$ in (3.2.2), we obtain

$$x_1(n) = 2^n u(n) \overset{z}{\longleftrightarrow} X_1(z) = \frac{1}{1 - 2z^{-1}}, \qquad \text{ROC: } |z| > 2$$

$$x_2(n) = 3^n u(n) \overset{z}{\longleftrightarrow} X_2(z) = \frac{1}{1 - 3z^{-1}}, \qquad \text{ROC: } |z| > 3$$

The intersection of the ROC of $X_1(z)$ and $X_2(z)$ is $|z| > 3$. Thus the overall transform $X(z)$ is

$$X(z) = \frac{3}{1 - 2z^{-1}} - \frac{4}{1 - 3z^{-1}}, \qquad \text{ROC: } |z| > 3$$

EXAMPLE 3.2.2

Determine the z-transform of the signals

(a) $x(n) = (\cos \omega_0 n)u(n)$

(b) $x(n) = (\sin \omega_0 n)u(n)$

Solution.

(a) By using Euler's identity, the signal $x(n)$ can be expressed as

$$x(n) = (\cos \omega_0 n)u(n) = \frac{1}{2}e^{j\omega_0 n}u(n) + \frac{1}{2}e^{-j\omega_0 n}u(n)$$

Thus (3.2.1) implies that

$$X(z) = \frac{1}{2}Z\{e^{j\omega_0 n}u(n)\} + \frac{1}{2}Z\{e^{-j\omega_0 n}u(n)\}$$

If we set $\alpha = e^{\pm j\omega_0}(|\alpha| = |e^{\pm j\omega_0}| = 1)$ in (3.2.2), we obtain

$$e^{j\omega_0 n}u(n) \overset{z}{\longleftrightarrow} \frac{1}{1 - e^{j\omega_0}z^{-1}}, \qquad \text{ROC: } |z| > 1$$

and

$$e^{-j\omega_0 n}u(n) \overset{z}{\longleftrightarrow} \frac{1}{1 - e^{-j\omega_0}z^{-1}}, \qquad \text{ROC: } |z| > 1$$

Thus

$$X(z) = \frac{1}{2}\frac{1}{1 - e^{j\omega_0}z^{-1}} + \frac{1}{2}\frac{1}{1 - e^{-j\omega_0}z^{-1}}, \qquad \text{ROC: } |z| > 1$$

After some simple algebraic manipulations we obtain the desired result, namely,

$$(\cos \omega_0 n)u(n) \overset{z}{\longleftrightarrow} \frac{1 - z^{-1}\cos\omega_0}{1 - 2z^{-1}\cos\omega_0 + z^{-2}}, \qquad \text{ROC: } |z| > 1 \tag{3.2.3}$$

(b) From Euler's identity,

$$x(n) = (\sin \omega 0n)u(n) = \frac{1}{2j}[e^{j\omega_0 n}u(n) - e^{-j\omega_0 n}u(n)]$$

Thus

$$X(z) = \frac{1}{2j}\left(\frac{1}{1 - e^{j\omega_0}z^{-1}} - \frac{1}{1 - e^{-j\omega_0}z^{-1}}\right), \qquad \text{ROC: } |z| > 1$$

and finally,

$$(\sin \omega_0 n)u(n) \overset{z}{\longleftrightarrow} \frac{z^{-1}\sin\omega_0}{1 - 2z^{-1}\cos\omega_0 + z^{-2}}, \qquad \text{ROC: } |z| > 1 \tag{3.2.4}$$

Time shifting. If

$$x(n) \overset{z}{\longleftrightarrow} X(z)$$

then

$$x(n - k) \overset{z}{\longleftrightarrow} z^{-k}X(z) \tag{3.2.5}$$

The ROC of $z^{-k}X(z)$ is the same as that of $X(z)$ except for $z = 0$ if $k > 0$ and $z = \infty$ if $k < 0$. The proof of this property follows immediately from the definition of the z-transform given in (3.1.1)

The properties of linearity and time shifting are the key features that make the z-transform extremely useful for the analysis of discrete-time LTI systems.

EXAMPLE 3.2.3

By applying the time-shifting property, determine the z-transform of the signals $x_2(n)$ and $x_3(n)$ in Example 3.1.1 from the z-transform of $x_1(n)$.

Solution. It can easily be seen that

$$x_2(n) = x_1(n+2)$$

and

$$x_3(n) = x_1(n-2)$$

Thus from (3.2.5) we obtain

$$X_2(z) = z^2 X_1(z) = z^2 + 2z + 5 + 7z^{-1} + z^{-3}$$

and

$$X_3(z) = z^{-2} X_1(z) = z^{-2} + 2z^{-3} + 5z^{-4} + 7z^{-5} + z^{-7}$$

Note that because of the multiplication by z^2, the ROC of $X_2(z)$ does not include the point $z = \infty$, even if it is contained in the ROC of $X_1(z)$.

Example 3.2.3 provides additional insight in understanding the meaning of the shifting property. Indeed, if we recall that the coefficient of z^{-n} is the sample value at time n, it is immediately seen that delaying a signal by $k (k > 0)$ samples [i.e., $x(n) \rightarrow x(n-k)$] corresponds to multiplying all terms of the z-transform by z^{-k}. The coefficient of z^{-n} becomes the coefficient of $z^{-(n+k)}$.

EXAMPLE 3.2.4

Determine the transform of the signal

$$x(n) = \begin{cases} 1, & 0 \le n \le N-1 \\ 0, & \text{elsewhere} \end{cases} \tag{3.2.6}$$

Solution. We can determine the z-transform of this signal by using the definition (3.1.1). Indeed,

$$X(z) = \sum_{n=0}^{N-1} 1 \cdot z^{-n} = 1 + z^{-1} + \cdots + z^{-(N-1)} = \begin{cases} N, & \text{if } z = 1 \\ \dfrac{1 - z^{-N}}{1 - z^{-1}}, & \text{if } z \neq 1 \end{cases} \tag{3.2.7}$$

Since $x(n)$ has finite duration, its ROC is the entire z-plane, except $z = 0$.

Let us also derive this transform by using the linearity and time-shifting properties. Note that $x(n)$ can be expressed in terms of two unit step signals

$$x(n) = u(n) - u(n-N)$$

By using (3.2.1) and (3.2.5) we have

$$X(z) = Z\{u(n)\} - Z\{u(n-N)\} = (1 - z^{-N}) Z\{u(n)\} \tag{3.2.8}$$

However, from (3.1.8) we have

$$Z\{u(n)\} = \frac{1}{1 - z^{-1}}, \qquad \text{ROC: } |z| > 1$$

which, when combined with (3.2.8), leads to (3.2.7).

Example 3.2.4 helps to clarify a very important issue regarding the ROC of the combination of several z-transforms. If the linear combination of several signals has finite duration, the ROC of its z-transform is exclusively dictated by the finite-duration nature of this signal, not by the ROC of the individual transforms.

Scaling in the z-domain. If

$$x(n) \overset{z}{\longleftrightarrow} X(z), \qquad \text{ROC: } r_1 < |z| < r_2$$

then

$$a^n x(n) \overset{z}{\longleftrightarrow} X(a^{-1}z), \qquad \text{ROC: } |a|r_1 < |z| < |a|r_2 \tag{3.2.9}$$

for any constant a, real or complex.

Proof From the definition (3.1.1)

$$Z\{a^n x(n)\} = \sum_{n=-\infty}^{\infty} a^n x(n) z^{-n} = \sum_{n=-\infty}^{\infty} x(n)(a^{-1}z)^{-n}$$

$$= X(a^{-1}z)$$

Since the ROC of $X(z)$ is $r_1 < |z| < r_2$, the ROC of $X(a^{-1}z)$ is

$$r_1 < |a^{-1}z| < r_2$$

or

$$|a|r_1 < |z| < |a|r_2$$

To better understand the meaning and implications of the scaling property, we express a and z in polar form as $a = r_0 e^{j\omega_0}$, $z = re^{j\omega}$, and we introduce a new complex variable $w = a^{-1}z$. Thus $Z\{x(n)\} = X(z)$ and $Z\{a^n x(n)\} = X(w)$. It can easily be seen that

$$w = a^{-1}z = \left(\frac{1}{r_0} r\right) e^{j(\omega - \omega_0)}$$

This change of variables results in either shrinking (if $r_0 > 1$) or expanding (if $r_0 < 1$) the z-plane in combination with a rotation (if $\omega_0 \neq 2k\pi$) of the z-plane (see Fig. 3.2.1). This explains why we have a change in the ROC of the new transform where $|a| < 1$. The case $|a| = 1$, that is, $a = e^{j\omega_0}$ is of special interest because it corresponds only to rotation of the z-plane.

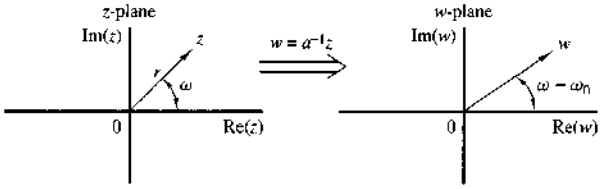


Figure 3.2.1 Mapping of the z-plane to the w-plane via the transformation $w = a^{-1}z$, $a = r_0 e^{j\omega_0}$.

EXAMPLE 3.2.5

Determine the z-transforms of the signals

(a) $x(n) = a^n(\cos \omega_0 n)u(n)$

(b) $x(n) = a^n(\sin \omega_0 n)u(n)$

Solution.

(a) From (3.2.3) and (3.2.9) we easily obtain

$$a^n(\cos \omega_0 n)u(n) \overset{z}{\longleftrightarrow} \frac{1 - az^{-1}\cos \omega_0}{1 - 2az^{-1}\cos \omega_0 + a^2 z^{-2}}, \qquad |z| > |a| \tag{3.2.10}$$

(b) Similarly, (3.2.4) and (3.2.9) yield

$$a^n(\sin \omega_0 n)u(n) \overset{z}{\longleftrightarrow} \frac{az^{-1}\sin \omega_0}{1 - 2az^{-1}\cos \omega_0 + a^2 z^{-2}}, \qquad |z| > |a| \tag{3.2.11}$$

Time reversal. If

$$x(n) \overset{z}{\longleftrightarrow} X(z), \qquad \text{ROC: } r_1 < |z| < r_2$$

then

$$x(-n) \overset{z}{\longleftrightarrow} X(z^{-1}), \qquad \text{ROC: } \frac{1}{r_2} < |z| < \frac{1}{r_1} \tag{3.2.12}$$

Proof From the definition (3.1.1), we have

$$Z\{x(-n)\} = \sum_{n=-\infty}^{\infty} x(-n)z^{-n} = \sum_{l=-\infty}^{\infty} x(l)(z^{-1})^{-l} = X(z^{-1})$$

where the change of variable $l = -n$ is made. The ROC of $X(z^{-1})$ is

$$r_1 < |z^{-1}| < r_2 \quad \text{or equivalently} \quad \frac{1}{r_2} < |z| < \frac{1}{r_1}$$

Note that the ROC for $x(n)$ is the inverse of that for $x(-n)$. This means that if z_0 belongs to the ROC of $x(n)$, then $1/z_0$ is in the ROC for $x(-n)$.

An intuitive proof of (3.2.12) is the following. When we fold a signal, the coefficient of z^{-n} becomes the coefficient of z^n. Thus, folding a signal is equivalent to replacing z by z^{-1} in the z-transform formula. In other words, reflection in the time domain corresponds to inversion in the z-domain.

EXAMPLE 3.2.6

Determine the z-transform of the signal

$$x(n) = u(-n)$$

Solution. It is known from (3.1.8) that

$$u(n) \overset{z}{\longleftrightarrow} \frac{1}{1-z^{-1}}, \qquad \text{ROC: } |z| > 1$$

By using (3.2.12), we easily obtain

$$u(-n) \overset{z}{\longleftrightarrow} \frac{1}{1-z}, \qquad \text{ROC: } |z| < 1 \tag{3.2.13}$$

Differentiation in the z-domain. If

$$x(n) \overset{z}{\longleftrightarrow} X(z)$$

then

$$nx(n) \overset{z}{\longleftrightarrow} -z\frac{dX(z)}{dz} \tag{3.2.14}$$

Proof By differentiating both sides of (3.1.1), we have

$$\frac{dX(z)}{dz} = \sum_{n=-\infty}^{\infty} x(n)(-n)z^{-n-1} = -z^{-1} \sum_{n=-\infty}^{\infty} [nx(n)]z^{-n}$$

$$= -z^{-1}Z\{nx(n)\}$$

Note that both transforms have the same ROC.

EXAMPLE 3.2.7

Determine the z-transform of the signal

$$x(n) = na^n u(n)$$

Solution. The signal $x(n)$ can be expressed as $nx_1(n)$, where $x_1(n) = a^n u(n)$. From (3.2.2) we have that

$$x_1(n) = a^n u(n) \overset{z}{\longleftrightarrow} X_1(z) = \frac{1}{1-az^{-1}}, \qquad \text{ROC: } |z| > |a|$$

Thus, by using (3.2.14), we obtain

$$na^n u(n) \overset{z}{\longleftrightarrow} X(z) = -z\frac{dX_1(z)}{dz} = \frac{az^{-1}}{(1-az^{-1})^2}, \qquad \text{ROC: } |z| > |a| \tag{3.2.15}$$

If we set $a = 1$ in (3.2.15), we find the z-transform of the unit ramp signal

$$nu(n) \overset{z}{\longleftrightarrow} \frac{z^{-1}}{(1-z^{-1})^2}, \qquad \text{ROC: } |z| > 1 \tag{3.2.16}$$

EXAMPLE 3.2.8

Determine the signal $x(n)$ whose z-transform is given by

$$X(z) = \log(1 + az^{-1}), \qquad |z| > |a|$$

Solution. By taking the first derivative of $X(z)$, we obtain

$$\frac{dX(z)}{dz} = \frac{-az^{-2}}{1+az^{-1}}$$

Thus

$$-z\frac{dX(z)}{dz} = az^{-1}\left[\frac{1}{1-(-a)z^{-1}}\right], \qquad |z| > |a|$$

The inverse z-transform of the term in brackets is $(-a)^n$. The multiplication by z^{-1} implies a time delay by one sample (time-shifting property), which results in $(-a)^{n-1}u(n-1)$. Finally, from the differentiation property we have

$$nx(n) = a(-a)^{n-1}u(n-1)$$

or

$$x(n) = (-1)^{n+1}\frac{a^n}{n}u(n-1)$$

Convolution of two sequences. If

$$x_1(n) \overset{z}{\longleftrightarrow} X_1(z)$$

$$x_2(n) \overset{z}{\longleftrightarrow} X_2(z)$$

then

$$x(n) = x_1(n) * x_2(n) \overset{z}{\longleftrightarrow} X(z) = X_1(z)X_2(z) \tag{3.2.17}$$

The ROC of $X(z)$ is, at least, the intersection of that for $X_1(z)$ and $X_2(z)$.

Proof The convolution of $x_1(n)$ and $x_2(n)$ is defined as

$$x(n) = \sum_{k=-\infty}^{\infty} x_1(k)x_2(n-k)$$

The z-transform of $x(n)$ is

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n} = \sum_{n=-\infty}^{\infty}\left[\sum_{k=-\infty}^{\infty} x_1(k)x_2(n-k)\right]z^{-n}$$

Upon interchanging the order of the summations and applying the time-shifting property in (3.2.5), we obtain

$$X(z) = \sum_{k=-\infty}^{\infty} x_1(k)\left[\sum_{n=-\infty}^{\infty} x_2(n-k)z^{-n}\right]$$

$$= X_2(z)\sum_{k=-\infty}^{\infty} x_1(k)z^{-k} = X_2(z)X_1(z)$$

EXAMPLE 3.2.9

Compute the convolution $x(n)$ of the signals

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

$$x_2(n) = \begin{cases} 1, & 0 \le n \le 5 \\ 0, & \text{elsewhere} \end{cases}$$

Solution. From (3.1.1), we have

$$X_1(z) = 1 - 2z^{-1} + z^{-2}$$

$$X_2(z) = 1 + z^{-1} + z^{-2} + z^{-3} + z^{-4} + z^{-5}$$

According to (3.2.17), we carry out the multiplication of $X_1(z)$ and $X_2(z)$. Thus

$$X(z) = X_1(z)X_2(z) = 1 - z^{-1} - z^{-6} + z^{-7}$$

Hence

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

The same result can also be obtained by noting that

$$X_1(z) = (1 - z^{-1})^2$$

$$X_2(z) = \frac{1 - z^{-6}}{1 - z^{-1}}$$

Then

$$X(z) = (1 - z^{-1})(1 - z^{-6}) = 1 - z^{-1} - z^{-6} + z^{-7}$$

The reader is encouraged to obtain the same result explicitly by using the convolution summation formula (time-domain approach).

The convolution property is one of the most powerful properties of the z-transform because it converts the convolution of two signals (time domain) to multiplication of their transforms. Computation of the convolution of two signals, using the z-transform, requires the following steps:

1. Compute the z-transforms of the signals to be convolved.

$$X_1(z) = Z\{x_1(n)\}$$

(time domain $\longrightarrow$ z-domain)

$$X_2(z) = Z\{x_2(n)\}$$

2. Multiply the two z-transforms.

$$X(z) = X_1(z)X_2(z), \qquad (z\text{-domain})$$

3. Find the inverse z-transform of $X(z)$.

$$x(n) = Z^{-1}\{X(z)\}, \qquad (z\text{-domain} \longrightarrow \text{time domain})$$

This procedure is, in many cases, computationally easier than the direct evaluation of the convolution summation.

Correlation of two sequences. If

$$x_1(n) \overset{z}{\longleftrightarrow} X_1(z)$$

$$x_2(n) \overset{z}{\longleftrightarrow} X_2(z)$$

then

$$r_{x_1x_2}(l) = \sum_{n=-\infty}^{\infty} x_1(n)x_2(n-l) \overset{z}{\longleftrightarrow} R_{x_1x_2}(z) = X_1(z)X_2(z^{-1}) \tag{3.2.18}$$

Proof We recall that

$$r_{x_1x_2}(l) = x_1(l) * x_2(-l)$$

Using the convolution and time-reversal properties, we easily obtain

$$R_{x_1x_2}(z) = Z\{x_1(l)\}Z\{x_2(-l)\} = X_1(z)X_2(z^{-1})$$

The ROC of $R_{x_1x_2}(z)$ is at least the intersection of that for $X_1(z)$ and $X_2(z^{-1})$.

As in the case of convolution, the crosscorrelation of two signals is more easily done via polynomial multiplication according to (3.2.18) and then inverse transforming the result.

EXAMPLE 3.2.10

Determine the autocorrelation sequence of the signal

$$x(n) = a^n u(n), \qquad -1 < a < 1$$

Solution. Since the autocorrelation sequence of a signal is its correlation with itself, (3.2.18) gives

$$R_{xx}(z) = Z\{r_{xx}(l)\} = X(z)X(z^{-1})$$

From (3.2.2) we have

$$X(z) = \frac{1}{1-az^{-1}}, \qquad \text{ROC: } |z| > |a| \qquad \text{(causal signal)}$$

and by using (3.2.15), we obtain

$$X(z^{-1}) = \frac{1}{1-az}, \qquad \text{ROC: } |z| < \frac{1}{|a|} \qquad \text{(anticausal signal)}$$

Thus

$$R_{xx}(z) = \frac{1}{1-az^{-1}}\frac{1}{1-az} = \frac{1}{1-a(z+z^{-1})+a^2}, \qquad \text{ROC: } |a| < |z| < \frac{1}{|a|}$$

Since the ROC of $R_{xx}(z)$ is a ring, $r_{xx}(l)$ is a two-sided signal, even if $x(n)$ is causal.

To obtain $r_{xx}(l)$, we observe that the z-transform of the sequence in Example 3.1.5 with $b = 1/a$ is simply $(1 - a^2)R_{xx}(z)$. Hence it follows that

$$r_{xx}(l) = \frac{1}{1 - a^2} a^{|l|}, \qquad -\infty < l < \infty$$

The reader is encouraged to compare this approach with the time-domain solution of the same problem given in Section 2.6.

Multiplication of two sequences. If

$$x_1(n) \overset{z}{\longleftrightarrow} X_1(z)$$

$$x_2(n) \overset{z}{\longleftrightarrow} X_2(z)$$

then

$$x(n) = x_1(n)x_2(n) \overset{z}{\longleftrightarrow} X(z) = \frac{1}{2\pi j} \oint_C X_1(v) X_2\left(\frac{z}{v}\right) v^{-1}\, dv \tag{3.2.19}$$

where C is a closed contour that encloses the origin and lies within the region of convergence common to both $X_1(v)$ and $X_2(1/v)$.

Proof The z-transform of $x_3(n)$ is

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n} = \sum_{n=-\infty}^{\infty} x_1(n)x_2(n)z^{-n}$$

Let us substitute the inverse transform

$$x_1(n) = \frac{1}{2\pi j} \oint_C X_1(v) v^{n-1} dv$$

for $x_1(n)$ in the z-transform $X(z)$ and interchange the order of summation and integration. Thus we obtain

$$X(z) = \frac{1}{2\pi j} \oint_C X_1(v) \left[\sum_{n=-\infty}^{\infty} x_2(n) \left(\frac{z}{v}\right)^{-n} \right] v^{-1} dv$$

The sum in the brackets is simply the transform $X_2(z)$ evaluated at z/v. Therefore,

$$X(z) = \frac{1}{2\pi j} \oint_C X_1(v) X_2\left(\frac{z}{v}\right) v^{-1}\, dv$$

which is the desired result.

To obtain the ROC of $X(z)$ we note that if $X_1(v)$ converges for $r_{1l} < |v| < r_{1u}$ and $X_2(z)$ converges for $r_{2l} < |z| < r_{2u}$, then the ROC of $X_2(z/v)$ is

$$r_{2l} < \left|\frac{z}{v}\right| < r_{2u}$$

Hence the ROC for $X(z)$ is at least

$$r_{1l}r_{2l} < |z| < r_{1u}r_{2u} \tag{3.2.20}$$

Although this property will not be used immediately, it will prove useful later, especially in our treatment of filter design based on the window technique, where we multiply the impulse response of an IIR system by a finite-duration "window" which serves to truncate the impulse response of the IIR system.

For complex-valued sequences $x_1(n)$ and $x_2(n)$ we can define the product sequence as $x(n) = x_1(n)x_2^*(n)$. Then the corresponding complex convolution integral becomes

$$x(n) = x_1(n)x_2^*(n) \overset{z}{\longleftrightarrow} X(z) = \frac{1}{2\pi j}\oint_C X_1(v)X_2^*\left(\frac{z^*}{v^*}\right)v^{-1}\,dv \tag{3.2.21}$$

The proof of (3.2.21) is left as an exercise for the reader.

Parseval's relation. If $x_1(n)$ and $x_2(n)$ are complex-valued sequences, then

$$\sum_{n=-\infty}^{\infty} x_1(n)x_2^*(n) = \frac{1}{2\pi j}\oint_C X_1(v)X_2^*\left(\frac{1}{v^*}\right)v^{-1}\,dv \tag{3.2.22}$$

provided that $r_{1l}r_{2l} < 1 < r_{1u}r_{2u}$, where $r_{1l} < |z| < r_{1u}$ and $r_{2l} < |z| < r_2u$ are the ROC of $X_1(z)$ and $X_2(z)$. The proof of (3.2.22) follows immediately by evaluating $X(z)$ in (3.2.21) at $z = 1$.

The Initial Value Theorem. If $x(n)$ is *causal* [i.e., $x(n) = 0$ for $n < 0$], then

$$x(0) = \lim_{z\to\infty} X(z) \tag{3.2.23}$$

Proof Since $x(n)$ is causal, (3.1.1) gives

$$X(z) = \sum_{n=0}^{\infty} x(n)z^{-n} = x(0) + x(1)z^{-1} + x(2)z^{-2} + \cdots$$

Obviously, as $z \to \infty$, $z^{-n} \to 0$ since $n > 0$, and (3.2.23) follows.

TABLE 3.2 Properties of the z-Transform

Property	Time Domain	z-Domain	ROC
Notation	$x(n)$	$X(z)$	ROC: $r_2 < \|z\| < r_1$
	$x_1(n)$	$X_1(z)$	ROC_1
	$x_2(n)$	$X_2(z)$	ROC_2
Linearity	$a_1x_1(n) + a_2x_2(n)$	$a_1X_1(z) + a_2X_2(z)$	At least the intersection of ROC_1 and ROC_2
Time shifting	$x(n-k)$	$z^{-k}X(z)$	That of $X(z)$, except $z = 0$ if $k > 0$ and $z = \infty$ if $k < 0$
Scaling in the z-domain	$a^n x(n)$	$X(a^{-1}z)$	$\|a\|r_2 < \|z\| < \|a\|r_1$
Time reversal	$x(-n)$	$X(z^{-1})$	$\frac{1}{r_1} < \|z\| < \frac{1}{r_2}$
Conjugation	$x^*(n)$	$X^*(z^*)$	ROC
Real part	$\text{Re}\{x(n)\}$	$\frac{1}{2}[X(z) + X^*(z^*)]$	Includes ROC
Imaginary part	$\text{Im}\{x(n)\}$	$\frac{1}{2}j[X(z) - X^*(z^*)]$	Includes ROC
Differentiation in the z-domain	$nx(n)$	$-z\frac{dX(z)}{dz}$	$r_2 < \|z\| < r_1$
Convolution	$x_1(n) * x_2(n)$	$X_1(z)X_2(z)$	At least, the intersection of ROC_1 and ROC_2
Correlation	$r_{x_1x_2}(l) = x_1(l) * x_2(-l)$	$R_{x_1x_2}(z) = X_1(z)X_2(z^{-1})$	At least, the intersection of ROC of $X_1(z)$ and $X_2(z^{-1})$
Initial value theorem	If $x(n)$ causal	$x(0) = \lim_{z\to\infty} X(z)$	
Multiplication	$x_1(n)x_2(n)$	$\frac{1}{2\pi j}\oint_C X_1(v)X_2\left(\frac{z}{v}\right)v^{-1}\,dv$	At least, $r_{1l}r_{2l} < \|z\| < r_{1u}r_{2u}$
Parseval's relation	$\sum_{n=-\infty}^{\infty} x_1(n)x_2^*(n) = \frac{1}{2\pi j}\oint_C X_1(v)X_2^*(1/v^*)v^{-1}dv$		

All the properties of the z-transform presented in this section are summarized in Table 3.2 for easy reference. They are listed in the same order as they have been introduced in the text. The conjugation properties and Parseval's relation are left as exercises for the reader.

We have now derived most of the z-transforms that are encountered in many practical applications. These z-transform pairs are summarized in Table 3.3 for easy reference. A simple inspection of this table shows that these z-transforms are all *rational functions* (i.e., ratios of polynomials in z^{-1}). As will soon become apparent, rational z-transforms are encountered not only as the z-transforms of various important signals but also in the characterization of discrete-time linear time-invariant systems described by constant-coefficient difference equations.

TABLE 3.3 Some Common z-Transform Pairs

	Signal, $x(n)$	z-Transform, $X(z)$	ROC
1	$\delta(n)$	1	All z
2	$u(n)$	$\frac{1}{1-z^{-1}}$	$\|z\| > 1$
3	$a^n u(n)$	$\frac{1}{1-az^{-1}}$	$\|z\| > \|a\|$
4	$na^n u(n)$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$\|z\| > \|a\|$
5	$-a^n u(-n-1)$	$\frac{1}{1-az^{-1}}$	$\|z\| < \|a\|$
6	$-na^n u(-n-1)$	$\frac{az^{-1}}{(1-az^{-1})^2}$	$\|z\| < \|a\|$
7	$(\cos \omega_0 n)u(n)$	$\frac{1-z^{-1}\cos\omega_0}{1-2z^{-1}\cos\omega_0+z^{-2}}$	$\|z\| > 1$
8	$(\sin \omega_0 n)u(n)$	$\frac{z^{-1}\sin\omega_0}{1-2z^{-1}\cos\omega_0+z^{-2}}$	$\|z\| > 1$
9	$(a^n \cos \omega_0 n)u(n)$	$\frac{1-az^{-1}\cos\omega_0}{1-2az^{-1}\cos\omega_0+a^2z^{-2}}$	$\|z\| > \|a\|$
10	$(a^n \sin \omega_0 n)u(n)$	$\frac{az^{-1}\sin\omega_0}{1-2az^{-1}\cos\omega_0+a^2z^{-2}}$	$\|z\| > \|a\|$

3.3 Rational z-Transforms

As indicated in Section 3.2, an important family of z-transforms are those for which $X(z)$ is a rational function, that is, a ratio of two polynomials in z^{-1} (or z). In this section we discuss some very important issues regarding the class of rational z-transforms.

3.3.1 Poles and Zeros

The *zeros* of a z-transform $X(z)$ are the values of z for which $X(z) = 0$. The *poles* of a z-transform are the values of z for which $X(z) = \infty$. If $X(z)$ is a rational function, then

$$X(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z^{-1} + \cdots + b_M z^{-M}}{a_0 + a_1 z^{-1} + \cdots + a_N z^{-N}} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{k=0}^{N} a_k z^{-k}} \tag{3.3.1}$$

If $a_0 \neq 0$ and $b_0 \neq 0$, we can avoid the negative powers of z by factoring out the terms $b_0 z^{-M}$ and $a_0 z^{-N}$ as follows:

$$X(z) = \frac{B(z)}{A(z)} = \frac{b_0 z^{-M}}{a_0 z^{-N}} \frac{z^M + (b_1/b_0)z^{M-1} + \cdots + b_M/b_0}{z^N + (a_1/a_0)z^{N-1} + \cdots + a_N/a_0}$$

Since $B(z)$ and $A(z)$ are polynomials in z, they can be expressed in factored form as

$$X(z) = \frac{B(z)}{A(z)} = \frac{b_0}{a_0} z^{-M+N} \frac{(z - z_1)(z - z_2) \cdots (z - z_M)}{(z - p_1)(z - p_2) \cdots (z - p_N)}$$

$$X(z) = G z^{N-M} \frac{\prod_{k=1}^{M} (z - z_k)}{\prod_{k=1}^{N} (z - p_k)} \tag{3.3.2}$$

where $G \equiv b_0/a_0$. Thus $X(z)$ has M finite zeros at $z = z_1, z_2, \ldots, z_M$ (the roots of the numerator polynomial), N finite poles at $z = p_1, p_2, \ldots, p_N$ (the roots of the denominator polynomial), and $|N - M|$ zeros (if $N > M$) or poles (if $N < M$) at the origin $z = 0$. Poles or zeros may also occur at $z = \infty$. A zero exists at $z = \infty$ if $X(\infty) = 0$ and a pole exists at $z = \infty$ if $X(\infty) = \infty$. If we count the poles and zeros at zero and infinity, we find that $X(z)$ has exactly the same number of poles as zeros.

We can represent $X(z)$ graphically by a *pole–zero plot* (or *pattern*) in the complex plane, which shows the location of poles by crosses ($\times$) and the location of zeros by circles ($\circ$). The multiplicity of multiple-order poles or zeros is indicated by a number close to the corresponding cross or circle. Obviously, by definition, the ROC of a z-transform should not contain any poles.

EXAMPLE 3.3.1

Determine the pole–zero plot for the signal

$$x(n) = a^n u(n), \qquad a > 0$$

Solution. From Table 3.3 we find that

$$X(z) = \frac{1}{1 - az^{-1}} = \frac{z}{z - a}, \qquad \text{ROC: } |z| > a$$

Thus $X(z)$ has one zero at $z_1 = 0$ and one pole at $p_1 = a$. The pole–zero plot is shown in Fig. 3.3.1. Note that the pole $p_1 = a$ is not included in the ROC since the z-transform does not converge at a pole.

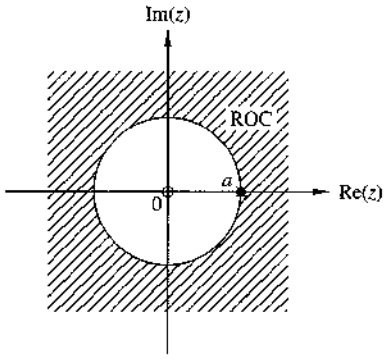


Figure 3.3.1
Pole–zero plot for the causal exponential signal $x(n) = a^n u(n)$.

EXAMPLE 3.3.2

Determine the pole–zero plot for the signal

$$x(n) = \begin{cases} a^n, & 0 \le n \le M-1 \\ 0, & \text{elsewhere} \end{cases}$$

where $a > 0$.

Solution. From the definition (3.1.1) we obtain

$$X(z) = \sum_{n=0}^{M-1} (az^{-1})^n = \frac{1-(az^{-1})^M}{1-az^{-1}} = \frac{z^M - a^M}{z^{M-1}(z-a)}$$

Since $a > 0$, the equation $z^M = a^M$ has M roots at

$$z_k = ae^{j2\pi k/M} \qquad k = 0, 1, \ldots, M-1$$

The zero $z_0 = a$ cancels the pole at $z = a$. Thus

$$X(z) = \frac{(z-z_1)(z-z_2)\cdots(z-z_{M-1})}{z^{M-1}}$$

which has $M-1$ zeros and $M-1$ poles, located as shown in Fig. 3.3.2 for $M = 8$. Note that the ROC is the entire z-plane except $z = 0$ because of the $M-1$ poles located at the origin

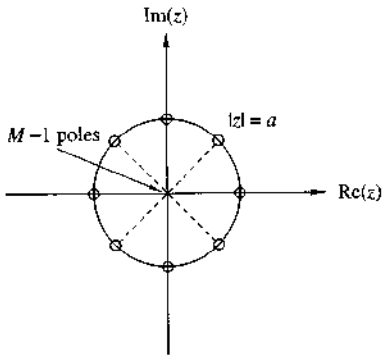


Figure 3.3.2
Pole–zero pattern for the finite-duration signal $x(n) = a^n$, $0 \le n \le M-1 (a > 0)$, for $M = 8$.

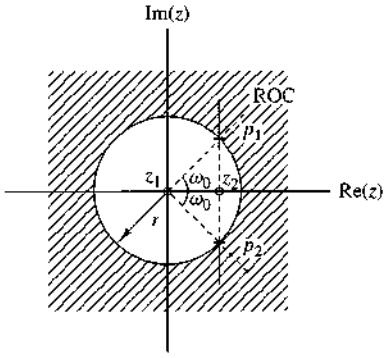


Figure 3.3.3
Pole-zero pattern for Example 3.3.3.

Clearly, if we are given a pole–zero plot, we can determine $X(z)$, by using (3.3.2), to within a scaling factor G. This is illustrated in the following example.

EXAMPLE 3.3.3

Determine the z-transform and the signal that corresponds to the pole–zero plot of Fig. 3.3.3.

Solution. There are two zeros ($M = 2$) at $z_1 = 0$, $z_2 = r\cos\omega_0$ and two poles ($N = 2$) at $p_1 = re^{j\omega_0}$, $p_2 = re^{-j\omega_0}$. By substitution of these relations into (3.3.2), we obtain

$$X(z) = G\frac{(z - z_1)(z - z_2)}{(z - p_1)(z - p_2)} = G\frac{z(z - r\cos\omega 0)}{(z - re^{j\omega_0})(z - re^{-j\omega_0})}, \qquad \text{ROC: } |z| > r$$

After some simple algebraic manipulations, we obtain

$$X(z) = G\frac{1 - rz^{-1}\cos\omega 0}{1 - 2rz^{-1}\cos\omega_0 + r^2z^{-2}}, \qquad \text{ROC: } |z| > r$$

From Table 3.3 we find that

$$x(n) = G(r^n\cos\omega_0 n)u(n)$$

From Example 3.3.3, we see that the product $(z - p_1)(z - p_2)$ results in a polynomial with real coefficients, when p_1 and p_2 are complex conjugates. In general, if a polynomial has real coefficients, its roots are either real or occur in complex-conjugate pairs.

As we have seen, the z-transform $X(z)$ is a complex function of the complex variable $z = \Re(z) + j\Im(z)$. Obviously, $|X(z)|$, the magnitude of $X(z)$, is a real and positive function of z. Since z represents a point in the complex plane, $|X(z)|$ is a two-dimensional function and describes a "surface." This is illustrated in Fig. 3.3.4 for the z-transform

$$X(z) = \frac{z^{-1} - z^{-2}}{1 - 1.2732z^{-1} + 0.81z^{-2}} \tag{3.3.3}$$

which has one zero at $z_1 = 1$ and two poles at p_1, $p_2 = 0.9e^{\pm j\pi/4}$. Note the high peaks near the singularities (poles) and the deep valley close to the zero.

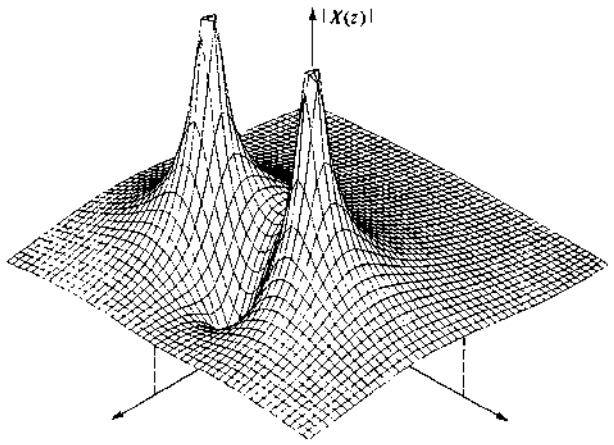


Figure 3.3.4 Graph of $|X(z)|$ for the z-transform in (3.3.3).

3.3.2 Pole Location and Time-Domain Behavior for Causal Signals

In this subsection we consider the relation between the z-plane location of a pole pair and the form (shape) of the corresponding signal in the time domain. The discussion is based generally on the collection of z-transform pairs given in Table 3.3 and the results in the preceding subsection. We deal exclusively with real, causal signals. In particular, we see that the characteristic behavior of causal signals depends on whether the poles of the transform are contained in the region $|z| < 1$, or in the region $|z| > 1$, or on the circle $|z| = 1$. Since the circle $|z| = 1$ has a radius of 1, it is called the *unit circle*.

If a real signal has a z-transform with one pole, this pole has to be real. The only such signal is the real exponential

$$x(n) = a^n u(n) \overset{z}{\longleftrightarrow} X(z) = \frac{1}{1 - az^{-1}}, \qquad \text{ROC: } |z| > |a|$$

having one zero at $z_1 = 0$ and one pole at $p_1 = a$ on the real axis. Figure 3.3.5 illustrates the behavior of the signal with respect to the location of the pole relative to the unit circle. The signal is decaying if the pole is inside the unit circle, fixed if the pole is on the unit circle, and growing if the pole is outside the unit circle. In addition, a negative pole results in a signal that alternates in sign. Obviously, causal signals with poles outside the unit circle become unbounded, cause overflow in digital systems, and in general, should be avoided.

A causal real signal with a double real pole has the form

$$x(n) = na^n u(n)$$

(see Table 3.3) and its behavior is illustrated in Fig. 3.3.6. Note that in contrast to the single-pole signal, a double real pole on the unit circle results in an unbounded signal.

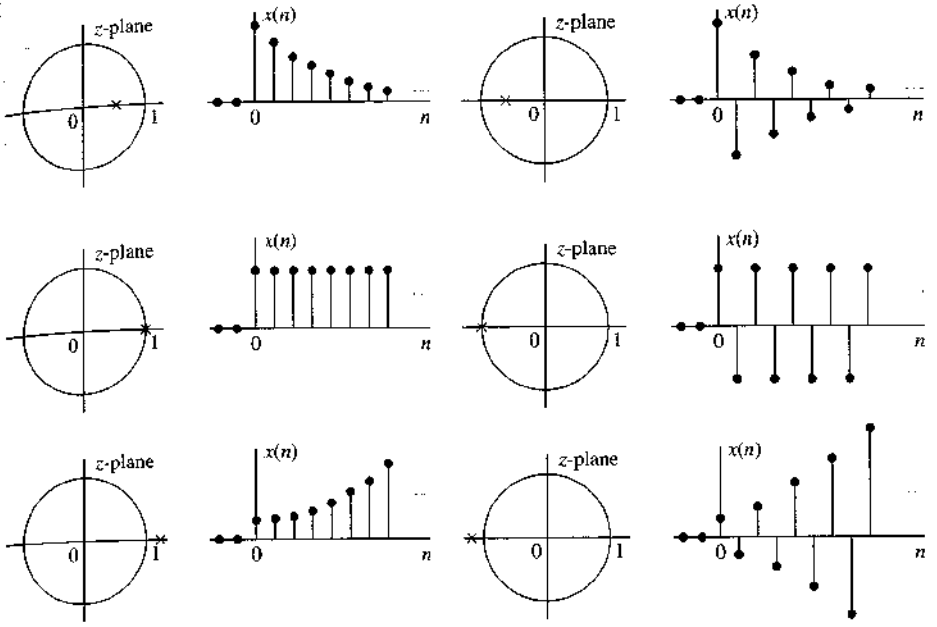


Figure 3.3.5 Time-domain behavior of a single-real-pole causal signal as a function of the location of the pole with respect to the unit circle.

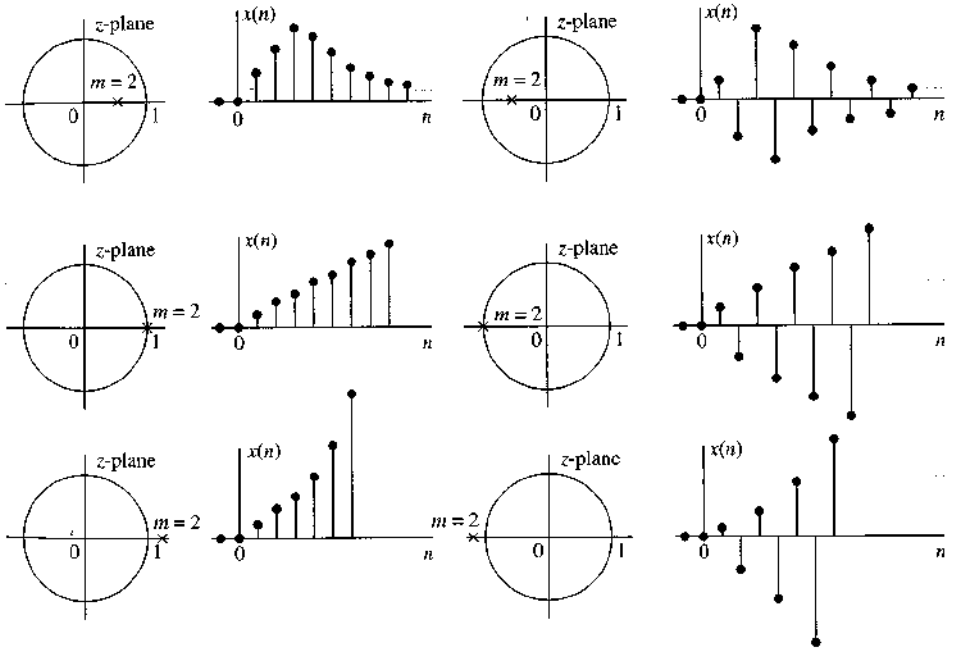


Figure 3.3.6 Time-domain behavior of causal signals corresponding to a double ($m = 2$) real pole, as a function of the pole location.

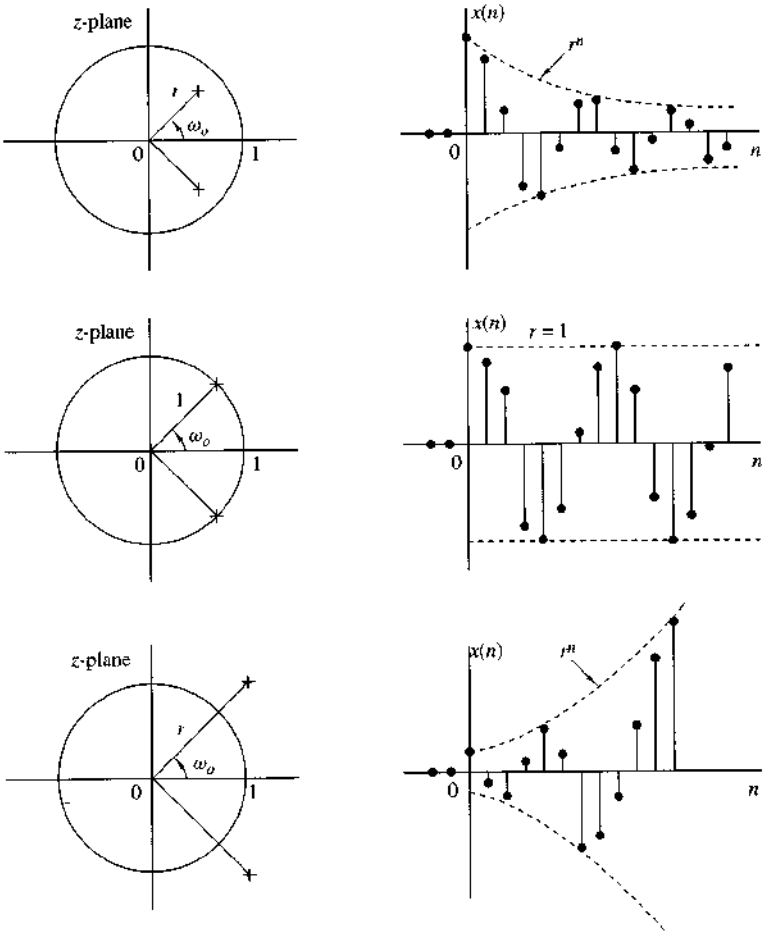


Figure 3.3.7 A pair of complex-conjugate poles corresponds to causal signals with oscillatory behavior.

Figure 3.3.7 illustrates the case of a pair of complex-conjugate poles. According to Table 3.3, this configuration of poles results in an exponentially weighted sinusoidal signal. The distance r of the poles from the origin determines the envelope of the sinusoidal signal and their angle with the real positive axis, its relative frequency. Note that the amplitude of the signal is growing if $r > 1$, constant if $r = 1$ (sinusoidal signals), and decaying if $r < 1$.

Finally, Fig. 3.3.8 shows the behavior of a causal signal with a double pair of poles on the unit circle. This reinforces the corresponding results in Fig. 3.3.6 and illustrates that multiple poles on the unit circle should be treated with great care.

To summarize, causal real signals with simple real poles or simple complex-conjugate pairs of poles, which are inside or on the unit circle, are always bounded in amplitude. Furthermore, a signal with a pole (or a complex-conjugate pair of poles)

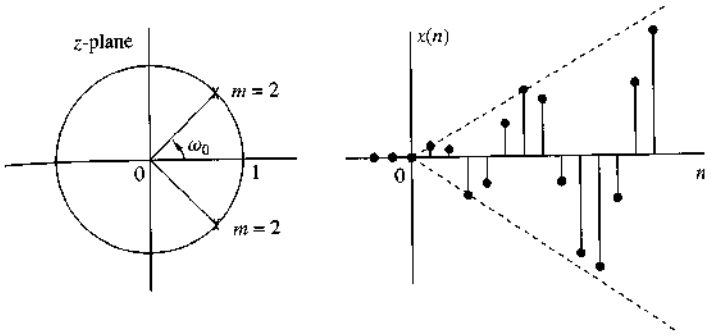


Figure 3.3.8 Causal signal corresponding to a double pair of complex-conjugate poles on the unit circle.

near the origin decays more rapidly than one associated with a pole near (but inside) the unit circle. Thus the time behavior of a signal depends strongly on the location of its poles relative to the unit circle. Zeros also affect the behavior of a signal but not as strongly as poles. For example, in the case of sinusoidal signals, the presence and location of zeros affects only their phase.

At this point, it should be stressed that everything we have said about causal signals applies as well to causal LTI systems, since their impulse response is a causal signal. Hence if a pole of a system is outside the unit circle, the impulse response of the system becomes unbounded and, consequently, the system is unstable.

3.3.3 The System Function of a Linear Time-Invariant System

In Chapter 2 we demonstrated that the output of a (relaxed) linear time-invariant system to an input sequence $x(n)$ can be obtained by computing the convolution of $x(n)$ with the unit sample response of the system. The convolution property, derived in Section 3.2, allows us to express this relationship in the z-domain as

$$Y(z) = H(z)X(z) \tag{3.3.4}$$

where $Y(z)$ is the z-transform of the output sequence $y(n)$, $X(z)$ is the z-transform of the input sequence $x(n)$ and $H(z)$ is the z-transform of the unit sample response $h(n)$.

If we know $h(n)$ and $x(n)$, we can determine their corresponding z-transforms $H(z)$ and $X(z)$, multiply them to obtain $Y(z)$, and therefore determine $y(n)$ by evaluating the inverse z-transform of $Y(z)$. Alternatively, if we know $x(n)$ and we observe the output $y(n)$ of the system, we can determine the unit sample response by first solving for $H(z)$ from the relation

$$H(z) = \frac{Y(z)}{X(z)} \tag{3.3.5}$$

and then evaluating the inverse z-transform of $H(z)$.

Since

$$H(z) = \sum_{n=-\infty}^{\infty} h(n)z^{-n} \tag{3.3.6}$$

it is clear that $H(z)$ represents the z-domain characterization of a system, whereas $h(n)$ is the corresponding time-domain characterization of the system. In other words, $H(z)$ and $h(n)$ are equivalent descriptions of a system in the two domains. The transform $H(z)$ is called the *system function*.

The relation in (3.3.5) is particularly useful in obtaining $H(z)$ when the system is described by a linear constant-coefficient difference equation of the form

$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k) \tag{3.3.7}$$

In this case the system function can be determined directly from (3.3.7) by computing the z-transform of both sides of (3.3.7). Thus, by applying the time-shifting property, we obtain

$$\begin{aligned} Y(z) &= -\sum_{k=1}^{N} a_k Y(z)z^{-k} + \sum_{k=0}^{M} b_k X(z)z^{-k} \\ Y(z)\left(1 + \sum_{k=1}^{N} a_k z^{-k}\right) &= X(z)\left(\sum_{k=0}^{M} b_k z^{-k}\right) \\ \frac{Y(z)}{X(z)} = H(z) &= \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}} \end{aligned} \tag{3.3.8}$$

Therefore, a linear time-invariant system described by a constant-coefficient difference equation has a rational system function.

This is the general form for the system function of a system described by a linear constant-coefficient difference equation. From this general form we obtain two important special forms. First, if $a_k = 0$ for $1 \le k \le N$, (3.3.8) reduces to

$$H(z) = \sum_{k=0}^{M} b_k z^{-k} = \frac{1}{z^M} \sum_{k=0}^{M} b_k z^{M-k} \tag{3.3.9}$$

In this case, $H(z)$ contains M zeros, whose values are determined by the system parameters $\{b_k\}$, and an Mth-order pole at the origin $z = 0$. Since the system contains only trivial poles (at $z = 0$) and M nontrivial zeros, it is called an *all-zero system*. Clearly, such a system has a finite-duration impulse response (FIR), and it is called an FIR system or a moving average (MA) system.

On the other hand, if $b_k = 0$ for $1 \le k \le M$, the system function reduces to

$$H(z) = \frac{b_0}{1 + \sum_{k=1}^{N} a_k z^{-k}} = \frac{b_0 z^N}{\sum_{k=0}^{N} a_k z^{N-k}}, \qquad a_0 \equiv 1 \tag{3.3.10}$$

In this case $H(z)$ consists of N poles, whose values are determined by the system parameters $\{a_k\}$ and an Nth-order zero at the origin $z = 0$. We usually do not make reference to these trivial zeros. Consequently, the system function in (3.3.10) contains only nontrivial poles and the corresponding system is called an *all-pole system*. Due to the presence of poles, the impulse response of such a system is infinite in duration, and hence it is an IIR system.

The general form of the system function given by (3.3.8) contains both poles and zeros, and hence the corresponding system is called a *pole–zero system*, with N poles and M zeros. Poles and/or zeros at $z = 0$ and $z = \infty$ are implied but are not counted explicitly. Due to the presence of poles, a pole–zero system is an IIR system.

The following example illustrates the procedure for determining the system function and the unit sample response from the difference equation.

EXAMPLE 3.3.4

Determine the system function and the unit sample response of the system described by the difference equation

$$y(n) = \frac{1}{2}y(n-1) + 2x(n)$$

Solution. By computing the z-transform of the difference equation, we obtain

$$Y(z) = \frac{1}{2}z^{-1}Y(z) + 2X(z)$$

Hence the system function is

$$H(z) = \frac{Y(z)}{X(z)} = \frac{2}{1 - \frac{1}{2}z^{-1}}$$

This system has a pole at $z = \frac{1}{2}$ and a zero at the origin. Using Table 3.3 we obtain the inverse transform

$$h(n) = 2\left(\frac{1}{2}\right)^n u(n)$$

This is the unit sample response of the system.

We have now demonstrated that rational z-transforms are encountered in commonly used systems and in the characterization of linear time-invariant systems. In Section 3.4 we describe several methods for determining the inverse z-transform of rational functions.

3.4 Inversion of the z-Transform

As we saw in Section 3.1.2, the inverse z-transform is formally given by

$$x(n) = \frac{1}{2\pi j} \oint_C X(z) z^{n-1} dz \tag{3.4.1}$$

where the integral is a contour integral over a closed path C that encloses the origin and lies within the region of convergence of $X(z)$. For simplicity, C can be taken as a circle in the ROC of $X(z)$ in the z-plane.

There are three methods that are often used for the evaluation of the inverse z-transform in practice:

1. Direct evaluation of (3.4.1), by contour integration.
2. Expansion into a series of terms, in the variables z, and z^{-1}.
3. Partial-fraction expansion and table lookup.

3.4.1 The Inverse z-Transform by Contour Integration

In this section we demonstrate the use of the Cauchy's integral theorem to determine the inverse z-transform directly from the contour integral.

Cauchy's integral theorem. Let $f(z)$ be a function of the complex variable z and C be a closed path in the z-plane. If the derivative $df(z)/dz$ exists on and inside the contour C and if $f(z)$ has no poles at $z = z_0$, then

$$\frac{1}{2\pi j} \oint_C \frac{f(z)}{z - z_0} dz = \begin{cases} f(z_0), & \text{if } z_0 \text{ is inside } C \\ 0, & \text{if } z_0 \text{ is outside } C \end{cases} \tag{3.4.2}$$

More generally, if the $(k+1)$-order derivative of $f(z)$ exists and $f(z)$ has no poles at $z = z_0$, then

$$\frac{1}{2\pi j} \oint_C \frac{f(z)}{(z - z_0)^k} dz = \begin{cases} \dfrac{1}{(k-1)!} \dfrac{d^{k-1} f(z)}{dz^{k-1}}\Big|_{z=z_0}, & \text{if } z_0 \text{ is inside } C \\ 0, & \text{if } z_0 \text{ is outside } C \end{cases} \tag{3.4.3}$$

The values on the right-hand side of (3.4.2) and (3.4.3) are called the residues of the pole at $z = z_0$. The results in (3.4.2) and (3.4.3) are two forms of the *Cauchy's integral theorem*.

We can apply (3.4.2) and (3.4.3) to obtain the values of more general contour integrals. To be specific, suppose that the integrand of the contour integral is a

proper fraction $f(z)/g(z)$, where $f(z)$ has no poles inside the contour C and $g(z)$ is a polynomial with distinct (simple) roots $z_1, z_2, \ldots, z_n$ inside C. Then

$$\begin{aligned}\frac{1}{2\pi j}\oint_C \frac{f(z)}{g(z)}\,dz &= \frac{1}{2\pi j}\oint_C \left[\sum_{i=1}^{n}\frac{A_i}{z-z_i}\right] dz \\ &= \sum_{i=1}^{n}\frac{1}{2\pi j}\oint_C \frac{A_i}{z-z_i}\,dz \qquad (3.4.4)\\ &= \sum_{i=1}^{n} A_i\end{aligned}$$

where

$$A_i = (z-z_i)\left.\frac{f(z)}{g(z)}\right|_{z=z_i} \qquad (3.4.5)$$

The values $\{A_i\}$ are residues of the corresponding poles at $z = z_i$, $i = 1, 2, \ldots, n$. Hence the value of the contour integral is equal to the sum of the residues of all the poles inside the contour C.

We observe that (3.4.4) was obtained by performing a partial-fraction expansion of the integrand and applying (3.4.2). When $g(z)$ has multiple-order roots as well as simple roots inside the contour, the partial-fraction expansion, with appropriate modifications, and (3.4.3) can be used to evaluate the residues at the corresponding poles.

In the case of the inverse z-transform, we have

$$\begin{aligned}x(n) &= \frac{1}{2\pi j}\oint_C X(z)z^{n-1}\,dz \\ &= \sum_{\text{all poles } \{z_i\} \text{ inside } C} [\text{residue of } X(z)z^{n-1} \text{ at } z = z_i] \qquad (3.4.6)\\ &= \sum_i (z-z_i)X(z)z^{n-1}|_{z=z_i}\end{aligned}$$

provided that the poles $\{z_i\}$ are simple. If $X(z)z^{n-1}$ has no poles inside the contour C for one or more values of n, then $x(n) = 0$ for these values.

The following example illustrates the evaluation of the inverse z-transform by use of the Cauchy's integral theorem.

EXAMPLE 3.4.1

Evaluate the inverse z-transform of

$$X(z) = \frac{1}{1-az^{-1}}, \qquad |z| > |a|$$

using the complex inversion integral.

Solution. We have

$$x(n) = \frac{1}{2\pi j} \oint_C \frac{z^{n-1}}{1 - az^{-1}}\, dz = \frac{1}{2\pi j} \oint_C \frac{z^n\, dz}{z - a}$$

where C is a circle at radius greater than $|a|$. We shall evaluate this integral using (3.4.2) with $f(z) = z^n$. We distinguish two cases.

1. If $n \geq 0$, $f(z)$ has only zeros and hence no poles inside C. The only pole inside C is $z = a$. Hence
$$x(n) = f(z_0) = a^n, \qquad n \geq 0$$
2. If $n < 0$, $f(z) = z^n$ has an nth-order pole at $z = 0$, which is also inside C. Thus there are contributions from both poles. For $n = -1$ we have
$$x(-1) = \frac{1}{2\pi j} \oint_C \frac{1}{z(z - a)}\, dz = \frac{1}{z - a}\bigg|_{z=0} + \frac{1}{z}\bigg|_{z=a} = 0$$
If $n = -2$, we have
$$x(-2) = \frac{1}{2\pi j} \oint_C \frac{1}{z^2(z - a)}\, dz = \frac{d}{dz}\left(\frac{1}{z - a}\right)\bigg|_{z=0} + \frac{1}{z^2}\bigg|_{z=a} = 0$$
By continuing in the same way we can show that $x(n) = 0$ for $n < 0$. Thus
$$x(n) = a^n u(n)$$

3.4.2 The Inverse z-Transform by Power Series Expansion

The basic idea in this method is the following: Given a z-transform $X(z)$ with its corresponding ROC, we can expand $X(z)$ into a power series of the form

$$X(z) = \sum_{n=-\infty}^{\infty} c_n z^{-n} \tag{3.4.7}$$

which converges in the given ROC. Then, by the uniqueness of the z-transform, $x(n) = c_n$ for all n. When $X(z)$ is rational, the expansion can be performed by long division.

To illustrate this technique, we will invert some z-transforms involving the same expression for $X(z)$, but different ROC. This will also serve to emphasize again the importance of the ROC in dealing with z-transforms.

EXAMPLE 3.4.2

Determine the inverse z-transform of

$$X(z) = \frac{1}{1 - 1.5z^{-1} + 0.5z^{-2}}$$

when

(a) ROC: $|z| > 1$

(b) ROC: $|z| < 0.5$

Solution.

(a) Since the ROC is the exterior of a circle, we expect $x(n)$ to be a causal signal. Thus we seek a power series expansion in negative powers of z. By dividing the numerator of $X(z)$ by its denominator, we obtain the power series

$$X(z) = \frac{1}{1 - \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2}} = 1 + \frac{3}{2}z^{-1} + \frac{7}{4}z^{-2} + \frac{15}{8}z^{-3} + \frac{31}{16}z^{-4} + \cdots$$

By comparing this relation with (3.1.1), we conclude that

$$x(n) = \{1, \underset{\uparrow}{\frac{3}{2}}, \frac{7}{4}, \frac{15}{8}, \frac{31}{16}, \ldots\}$$

Note that in each step of the long-division process, we eliminate the lowest-power term of z^{-1}.

(b) In this case the ROC is the interior of a circle. Consequently, the signal $x(n)$ is anticausal. To obtain a power series expansion in positive powers of z, we perform the long division in the following way:

$$\begin{array}{r} 2z^2 + 6z^3 + 14z^4 + 30z^5 + 62z^6 + \cdots \\ \hline \tfrac{1}{2}z^{-2} - \tfrac{3}{2}z^{-1} + 1 \;\big)\, 1 \\ \underline{1 - 3z + 2z^2} \\ 3z - 2z^2 \\ \underline{3z - 9z^2 + 6z^3} \\ 7z^2 - 6z^3 \\ \underline{7z^2 - 21z^3 + 14z^4} \\ 15z^3 - 14z^4 \\ \underline{15z^3 - 45z^4 + 30z^5} \\ 31z^4 - 30z^5 \end{array}$$

Thus

$$X(z) = \frac{1}{1 - \frac{3}{2}z^{-1} + \frac{1}{2}z^{-2}} = 2z^2 + 6z^3 + 14z^4 + 30z^5 + 62z^6 + \cdots$$

In this case $x(n) = 0$ for $n \geq 0$. By comparing this result to (3.1.1), we conclude that

$$x(n) = \{\quad 62, 30, 14, 6, 2, 0, \underset{\uparrow}{0}\}$$

We observe that in each step of the long-division process, the lowest-power term of z is eliminated. We emphasize that in the case of anticausal signals we simply carry out the long division by writing down the two polynomials in "reverse" order (i.e., starting with the most negative term on the left).

From this example we note that, in general, the method of long division will not provide answers for $x(n)$ when n is large because the long division becomes tedious. Although the method provides a direct evaluation of $x(n)$, a closed-form solution is not possible, except if the resulting pattern is simple enough to infer the general term $x(n)$. Hence this method is used only if one wishes to determine the values of the first few samples of the signal.

EXAMPLE 3.4.3

Determine the inverse z-transform of

$$X(z) = \log(1 + az^{-1}), \qquad |z| > |a|$$

Solution. Using the power series expansion for $\log(1 + x)$, with $|x| < 1$, we have

$$X(z) = \sum_{n=1}^{\infty} \frac{(-1)^{n+1} a^n z^{-n}}{n}$$

Thus

$$x(n) = \begin{cases} (-1)^{n+1} \dfrac{a^n}{n}, & n \geq 1 \\ 0, & n \leq 0 \end{cases}$$

Expansion of irrational functions into power series can be obtained from tables.

3.4.3 The Inverse z-Transform by Partial-Fraction Expansion

In the table lookup method, we attempt to express the function $X(z)$ as a linear combination

$$X(z) = \alpha_1 X_1(z) + \alpha_2 X_2(z) + \cdots + \alpha_K X_K(z) \tag{3.4.8}$$

where $X_1(z), \ldots, X_K(z)$ are expressions with inverse transforms $x_1(n), \ldots, x_K(n)$ available in a table of z-transform pairs. If such a decomposition is possible, then $x(n)$, the inverse z-transform of $X(z)$, can easily be found using the linearity property as

$$x(n) = \alpha_1 x_1(n) + \alpha_2 x_2(n) + \cdots + \alpha_K x_K(n) \tag{3.4.9}$$

This approach is particularly useful if $X(z)$ is a rational function, as in (3.3.1). Without loss of generality, we assume that $a_0 = 1$, so that (3.3.1) can be expressed as

$$X(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z^{-1} + \cdots + b_M z^{-M}}{1 + a_1 z^{-1} + \cdots + a_N z^{-N}} \tag{3.4.10}$$

Note that if $a_0 \neq 1$, we can obtain (3.4.10) from (3.3.1) by dividing both numerator and denominator by a_0.

A rational function of the form (3.4.10) is called *proper* if $a_N \neq 0$ and $M < N$. From (3.3.2) it follows that this is equivalent to saying that the number of finite zeros is less than the number of finite poles.

An improper rational function ($M \geq N$) can always be written as the sum of a polynomial and a proper rational function. This procedure is illustrated by the following example.

EXAMPLE 3.4.4

Express the improper rational transform

$$X(z) = \frac{1 + 3z^{-1} + \frac{11}{6} z^{-2} + \frac{1}{3} z^{-3}}{1 + \frac{5}{6} z^{-1} + \frac{1}{6} z^{-2}}$$

in terms of a polynomial and a proper function.

Solution. First, we note that we should reduce the numerator so that the terms z^{-2} and z^{-3} are eliminated. Thus we should carry out the long division with these two polynomials written in *reverse* order. We stop the division when the order of the remainder becomes z^{-1}. Then we obtain

$$X(z) = 1 + 2z^{-1} + \frac{\frac{1}{6}z^{-1}}{1 + \frac{5}{6}z^{-1} + \frac{1}{6}z^{-2}}$$

In general, any improper rational function ($M \geq N$) can be expressed as

$$X(z) = \frac{B(z)}{A(z)} = c_0 + c_1 z^{-1} + \cdots + c_{M-N} z^{-(M-N)} + \frac{B_1(z)}{A(z)} \tag{3.4.11}$$

The inverse z-transform of the polynomial can easily be found by inspection. We focus our attention on the inversion of proper rational transforms, since any improper function can be transformed into a proper function by using (3.4.11). We carry out the development in two steps. First, we perform a partial fraction expansion of the proper rational function and then we invert each of the terms.

Let $X(z)$ be a proper rational function, that is,

$$X(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z^{-1} + \cdots + b_M z^{-M}}{1 + a_1 z^{-1} + \cdots + a_N z^{-N}} \tag{3.4.12}$$

where

$$a_N \neq 0 \qquad \text{and} \qquad M < N$$

To simplify our discussion we eliminate negative powers of z by multiplying both the numerator and denominator of (3.4.12) by z^N. This results in

$$X(z) = \frac{b_0 z^N + b_1 z^{N-1} + \cdots + b_M z^{N-M}}{z^N + a_1 z^{N-1} + \cdots + a_N} \tag{3.4.13}$$

which contains only positive powers of z. Since $N > M$, the function

$$\frac{X(z)}{z} = \frac{b_0 z^{N-1} + b_1 z^{N-2} + \cdots + b_M z^{N-M-1}}{z^N + a_1 z^{N-1} + \cdots + a_N} \tag{3.4.14}$$

is also always proper.

Our task in performing a partial-fraction expansion is to express (3.4.14) or, equivalently, (3.4.12) as a sum of simple fractions. For this purpose we first factor the denominator polynomial in (3.4.14) into factors that contain the poles $p_1, p_2, \ldots, p_N$ of $X(z)$. We distinguish two cases.

Distinct poles. Suppose that the poles $p_1, p_2, \ldots, p_N$ are all different (distinct). Then we seek an expansion of the form

$$\frac{X(z)}{z} = \frac{A_1}{z - p_1} + \frac{A_2}{z - p_2} + \cdots + \frac{A_N}{z - p_N} \tag{3.4.15}$$

The problem is to determine the coefficients $A_1, A_2, \ldots, A_N$. There are two ways to solve this problem, as illustrated in the following example.

EXAMPLE 3.4.5

Determine the partial-fraction expansion of the proper function

$$X(z) = \frac{1}{1 - 1.5z^{-1} + 0.5z^{-2}} \tag{3.4.16}$$

Solution. First we eliminate the negative powers, by multiplying both numerator and denominator by z^2. Thus

$$X(z) = \frac{z^2}{z^2 - 1.5z + 0.5}$$

The poles of $X(z)$ are $p_1 = 1$ and $p_2 = 0.5$. Consequently, the expansion of the form (3.4.15) is

$$\frac{X(z)}{z} = \frac{z}{(z-1)(z-0.5)} = \frac{A_1}{z-1} + \frac{A_2}{z-0.5} \tag{3.4.17}$$

A very simple method to determine A_1 and A_2 is to multiply the equation by the denominator term $(z-1)(z-0.5)$. Thus we obtain

$$z = (z-0.5)A_1 + (z-1)A_2 \tag{3.4.18}$$

Now if we set $z = p_1 = 1$ in (3.4.18), we eliminate the term involving A_2. Hence

$$1 = (1-0.5)A_1$$

Thus we obtain the result $A_1 = 2$. Next we return to (3.4.18) and set $z = p_2 = 0.5$, thus eliminating the term involving A_1, so we have

$$0.5 = (0.5-1)A_2$$

and hence $A_2 = -1$. Therefore, the result of the partial-fraction expansion is

$$\frac{X(z)}{z} = \frac{2}{z-1} - \frac{1}{z-0.5} \tag{3.4.19}$$

The example given above suggests that we can determine the coefficients $A_1, A_2, \ldots, A_N$, by multiplying both sides of (3.4.15) by each of the terms $(z - p_k)$, $k = 1, 2, \ldots, N$, and evaluating the resulting expressions at the corresponding pole positions, $p_1, p_2, \ldots, p_N$. Thus we have, in general,

$$\frac{(z-p_k)X(z)}{z} = \frac{(z-p_k)A_1}{z-p_1} + \cdots + A_k + \cdots + \frac{(z-p_k)A_N}{z-p_N} \tag{3.4.20}$$

Consequently, with $z = p_k$, (3.4.20) yields the kth coefficient as

$$A_k = \left.\frac{(z-p_k)X(z)}{z}\right|_{z=p_k}, \qquad k = 1, 2, \ldots, N \tag{3.4.21}$$

EXAMPLE 3.4.6

Determine the partial-fraction expansion of

$$X(z) = \frac{1 + z^{-1}}{1 - z^{-1} + 0.5z^{-2}} \tag{3.4.22}$$

Solution. To eliminate negative powers of z in (3.4.22), we multiply both numerator and denominator by z^2. Thus

$$\frac{X(z)}{z} = \frac{z + 1}{z^2 - z + 0.5}$$

The poles of $X(z)$ are complex conjugates

$$p_1 = \frac{1}{2} + j\frac{1}{2}$$

and

$$p_2 = \frac{1}{2} - j\frac{1}{2}$$

Since $p_1 \neq p_2$, we seek an expansion of the form (3.4.15). Thus

$$\frac{X(z)}{z} = \frac{z + 1}{(z - p_1)(z - p_2)} = \frac{A_1}{z - p_1} + \frac{A_2}{z - p_2}$$

To obtain A_1 and A_2, we use the formula (3.4.21). Thus we obtain

$$A_1 = \left.\frac{(z - p_1)X(z)}{z}\right|_{z=p_1} = \left.\frac{z + 1}{z - p_2}\right|_{z=p_1} = \frac{\frac{1}{2} + j\frac{1}{2} + 1}{\frac{1}{2} + j\frac{1}{2} - \frac{1}{2} + j\frac{1}{2}} = \frac{1}{2} - j\frac{3}{2}$$

$$A_2 = \left.\frac{(z - p_2)X(z)}{z}\right|_{z=p_2} = \left.\frac{z + 1}{z - p_1}\right|_{z=p_2} = \frac{\frac{1}{2} - j\frac{1}{2} + 1}{\frac{1}{2} - j\frac{1}{2} - \frac{1}{2} - j\frac{1}{2}} = \frac{1}{2} + j\frac{3}{2}$$

The expansion (3.4.15) and the formula (3.4.21) hold for both real and complex poles. The only constraint is that all poles be distinct. We also note that $A_2 = A_1^*$. It can be easily seen that this is a consequence of the fact that $p_2 = p_1^*$. In other words, *complex-conjugate poles result in complex-conjugate coefficients in the partial-fraction expansion.* This simple result will prove very useful later in our discussion.

Multiple-order poles. If $X(z)$ has a pole of multiplicity l, that is, it contains in its denominator the factor $(z - p_k)^l$, then the expansion (3.4.15) is no longer true. In this case a different expansion is needed. First, we investigate the case of a double pole (i.e., $l = 2$).

EXAMPLE 3.4.7

Determine the partial-fraction expansion of

$$X(z) = \frac{1}{(1+z^{-1})(1-z^{-1})^2} \tag{3.4.23}$$

Solution. First, we express (3.4.23) in terms of positive powers of z, in the form

$$\frac{X(z)}{z} = \frac{z^2}{(z+1)(z-1)^2}$$

$X(z)$ has a simple pole at $p_1 = -1$ and a double pole $p_2 = p_3 = 1$. In such a case the appropriate partial-fraction expansion is

$$\frac{X(z)}{z} = \frac{z^2}{(z+1)(z-1)^2} = \frac{A_1}{z+1} + \frac{A_2}{z-1} + \frac{A_3}{(z-1)^2} \tag{3.4.24}$$

The problem is to determine the coefficients A_1, A_2, and A_3.

We proceed as in the case of distinct poles. To determine A_1, we multiply both sides of (3.4.24) by $(z+1)$ and evaluate the result at $z = -1$. Thus (3.4.24) becomes

$$\frac{(z+1)X(z)}{z} = A_1 + \frac{z+1}{z-1}A_2 + \frac{z+1}{(z-1)^2}A_3$$

which, when evaluated at $z = -1$, yields

$$A_1 = \left.\frac{(z+1)X(z)}{z}\right|_{z=-1} = \frac{1}{4}$$

Next, if we multiply both sides of (3.4.24) by $(z-1)^2$, we obtain

$$\frac{(z-1)^2X(z)}{z} = \frac{(z-1)^2}{z+1}A_1 + (z-1)A_2 + A_3 \tag{3.4.25}$$

Now, if we evaluate (3.4.25) at $z = 1$, we obtain A_3. Thus

$$A_3 = \left.\frac{(z-1)2X(z)}{z}\right|_{z=1} = \frac{1}{2}$$

The remaining coefficient A_2 can be obtained by differentiating both sides of (3.4.25) with respect to z and evaluating the result at $z = 1$. Note that it is not necessary formally to carry out the differentiation of the right-hand side of (3.4.25), since all terms except A_2 vanish when we set $z = 1$. Thus

$$A_2 = \frac{d}{dz}\left[\frac{(z-1)^2X(z)}{z}\right]_{z=1} = \frac{3}{4} \tag{3.4.26}$$

The generalization of the procedure in the example above to the case of an mth-order pole $(z-p_k)^m$ is straightforward. The partial-fraction expansion must contain the terms

$$\frac{A_{1k}}{z-p_k} + \frac{A_{2k}}{(z-p_k)^2} + \cdots + \frac{A_{mk}}{(z-p_k)^m}$$

The coefficients $\{A_{ik}\}$ can be evaluated through differentiation as illustrated in Example 3.4.7 for $m = 2$.

Now that we have performed the partial-fraction expansion, we are ready to take the final step in the inversion of $X(z)$. First, let us consider the case in which $X(z)$ contains distinct poles. From the partial-fraction expansion (3.4.15), it easily follows that

$$X(z) = A_1 \frac{1}{1 - p_1 z^{-1}} + A_2 \frac{1}{1 - p_2 z^{-1}} + \cdots + A_N \frac{1}{1 - p_N z^{-1}} \tag{3.4.27}$$

The inverse z-transform, $x(n) = Z^{-1}\{X(z)\}$, can be obtained by inverting each term in (3.4.27) and taking the corresponding linear combination. From Table 3.3 it follows that these terms can be inverted using the formula

$$Z^{-1}\left\{\frac{1}{1 - p_k z^{-1}}\right\} = \begin{cases} (p_k)^n u(n), & \text{if ROC: } |z| > |p_k| \\ & \text{(causal signals)} \\ -(p_k)^n u(-n-1), & \text{if ROC: } |z| < |p_k| \\ & \text{(anticausal signals)} \end{cases} \tag{3.4.28}$$

If the signal $x(n)$ is causal, the ROC is $|z| > p_{\max}$, where $p_{\max} = \max\{|p_1|, |p_2|, \ldots, |p_N|\}$. In this case all terms in (3.4.27) result in causal signal components and the signal $x(n)$ is given by

$$x(n) = (A_1 p_1^n + A_2 p_2^n + \cdots + A_N p_N^n) u(n) \tag{3.4.29}$$

If all poles are real, (3.4.29) is the desired expression for the signal $x(n)$. Thus a causal signal, having a z-transform that contains real and distinct poles, is a linear combination of real exponential signals.

Suppose now that all poles are distinct but some of them are complex. In this case some of the terms in (3.4.27) result in complex exponential components. However, if the signal $x(n)$ is real, we should be able to reduce these terms into real components. If $x(n)$ is real, the polynomials appearing in $X(z)$ have real coefficients. In this case, as we have seen in Section 3.3, if p_j is a pole, its complex conjugate p_j^* is also a pole. As was demonstrated in Example 3.4.6, the corresponding coefficients in the partial-fraction expansion are also complex conjugates. Thus the contribution of two complex-conjugate poles is of the form

$$x_k(n) = [A_k (p_k)^n + A_k^* (p_k^*)^n] u(n) \tag{3.4.30}$$

These two terms can be combined to form a real signal component. First, we express A_j and p_j in polar form (i.e., amplitude and phase) as

$$A_k = |A_k| e^{j\alpha_k} \tag{3.4.31}$$

$$p_k = r_k e^{j\beta_k} \tag{3.4.32}$$

where α_k and β_k are the phase components of A_k and p_k. Substitution of these relations into (3.4.30) gives

$$x_k(n) = |A_k| r_k^n [e^{j(\beta_k n + \alpha_k)} + e^{-j(\beta_k n + \alpha_k)}] u(n)$$

or, equivalently,

$$x_k(n) = 2|A_k|r_k^n \cos(\beta_k n + \alpha_k)u(n) \tag{3.4.33}$$

Thus we conclude that

$$Z^{-1}\left(\frac{A_k}{1 - p_k z^{-1}} + \frac{A_k^*}{1 - p_k^* z^{-1}}\right) = 2|A_k|r_k^n \cos(\beta_k n + \alpha_k)u(n) \tag{3.4.34}$$

if the ROC is $|z| > |p_k| = r_k$.

From (3.4.34) we observe that each pair of complex-conjugate poles in the z-domain results in a causal sinusoidal signal component with an exponential envelope. The distance r_k of the pole from the origin determines the exponential weighting (growing if $r_k > 1$, decaying if $r_k < 1$, constant if $r_k = 1$). The angle of the poles with respect to the positive real axis provides the frequency of the sinusoidal signal. The zeros, or equivalently the numerator of the rational transform, affect only indirectly the amplitude and the phase of $x_k(n)$ through A_k.

In the case of *multiple* poles, either real or complex, the inverse transform of terms of the form $A/(z - p_k)^n$ is required. In the case of a double pole the following transform pair (see Table 3.3) is quite useful:

$$Z^{-1}\left\{\frac{pz^{-1}}{(1 - pz^{-1})^2}\right\} = np^n u(n) \tag{3.4.35}$$

provided that the ROC is $|z| > |p|$. The generalization to the case of poles with higher multiplicity is obtained by using multiple differentiation.

EXAMPLE 3.4.8

Determine the inverse z-transform of

$$X(z) = \frac{1}{1 - 1.5z^{-1} + 0.5z^{-2}}$$

if

(a) ROC: $|z| > 1$

(b) ROC: $|z| < 0.5$

(c) ROC: $0.5 < |z| < 1$

Solution. This is the same problem that we treated in Example 3.4.2. The partial-fraction expansion for $X(z)$ was determined in Example 3.4.5. The partial-fraction expansion of $X(z)$ yields

$$X(z) = \frac{2}{1 - z^{-1}} - \frac{1}{1 - 0.5z^{-1}} \tag{3.4.36}$$

To invert $X(z)$ we should apply (3.4.28) for $p_1 = 1$ and $p_2 = 0.5$. However, this requires the specification of the corresponding ROC.

(a) In the case when the ROC is $|z| > 1$, the signal $x(n)$ is causal and both terms in (3.4.36) are causal terms. According to (3.4.28), we obtain

$$x(n) = 2(1)^n u(n) - (0.5)^n u(n) = (2 - 0.5^n)u(n) \tag{3.4.37}$$

which agrees with the result in Example 3.4.2(a).

(b) When the ROC is $|z| < 0.5$, the signal $x(n)$ is anticausal. Thus both terms in (3.4.36) result in anticausal components. From (3.4.28) we obtain

$$x(n) = [-2 + (0.5)^n]u(-n-1) \tag{3.4.38}$$

(c) In this case the ROC $0.5 < |z| < 1$ is a ring, which implies that the signal $x(n)$ is two-sided. Thus one of the terms corresponds to a causal signal and the other to an anticausal signal. Obviously, the given ROC is the overlapping of the regions $|z| > 0.5$ and $|z| < 1$. Hence the pole $p_2 = 0.5$ provides the causal part and the pole $p_1 = 1$ the anticausal. Thus

$$x(n) = -2(1)^n u(-n-1) - (0.5)^n u(n) \tag{3.4.39}$$

EXAMPLE 3.4.9

Determine the causal signal $x(n)$ whose z-transform is given by

$$X(z) = \frac{1 + z^{-1}}{1 - z^{-1} + 0.5z^{-2}}$$

Solution. In Example 3.4.6 we have obtained the partial-fraction expansion as

$$X(z) = \frac{A_1}{1 - p_1 z^{-1}} + \frac{A_2}{1 - p_2 z^{-1}}$$

where

$$A_1 = A_2^* = \frac{1}{2} - j\frac{3}{2}$$

and

$$p_1 = p_2^* = \frac{1}{2} + j\frac{1}{2}$$

Since we have a pair of complex-conjugate poles, we should use (3.4.34). The polar forms of A_1 and p_1 are

$$A_1 = \frac{\sqrt{10}}{2} e^{-j71.565}$$

$$p_1 = \frac{1}{\sqrt{2}} e^{j\pi/4}$$

Hence

$$x(n) = \sqrt{10}\left(\frac{1}{\sqrt{2}}\right)^n \cos\left(\frac{\pi n}{4} - 71.565^\circ\right) u(n)$$

EXAMPLE 3.4.10

Determine the causal signal $x(n)$ having the z-transform

$$X(z) = \frac{1}{(1 + z^{-1})(1 - z^{-1})^2}$$

Solution. From Example 3.4.7 we have

$$X(z) = \frac{1}{4}\frac{1}{1 + z^{-1}} + \frac{3}{4}\frac{1}{1 - z^{-1}} + \frac{1}{2}\frac{z^{-1}}{(1 - z^{-1})2}$$

By applying the inverse transform relations in (3.4.28) and (3.4.35), we obtain

$$x(n) = \frac{1}{4}(-1)^n u(n) + \frac{3}{4}u(n) + \frac{1}{2}nu(n) = \left[\frac{1}{4}(-1)^n + \frac{3}{4} + \frac{n}{2}\right]u(n)$$

3.4.4 Decomposition of Rational z-Transforms

At this point it is appropriate to discuss some additional issues concerning the decomposition of rational z-transforms, which will prove very useful in the implementation of discrete-time systems.

Suppose that we have a rational z-transform $X(z)$ expressed as

$$X(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}} = b_0 \frac{\prod_{k=1}^{M}(1 - z_k z^{-1})}{\prod_{k=1}^{N}(1 - p_k z^{-1})} \tag{3.4.40}$$

where, for simplicity, we have assumed that $a_0 \equiv 1$. If $M \geq N$ [i.e., $X(z)$ is improper], we convert $X(z)$ to a sum of a polynomial and a proper function

$$X(z) = \sum_{k=0}^{M-N} c_k z^{-k} + X_{\text{pr}}(z) \tag{3.4.41}$$

If the poles of $X_{\text{pr}}(z)$ are distinct, it can be expanded in partial fractions as

$$X_{\text{pr}}(z) = A_1 \frac{1}{1 - p_1 z^{-1}} + A_2 \frac{1}{1 - p_2 z^{-1}} + \cdots + A_N \frac{1}{1 - p_N z^{-1}} \tag{3.4.42}$$

As we have already observed, there may be some complex-conjugate pairs of poles in (3.4.42). Since we usually deal with real signals, we should avoid complex coefficients in our decomposition. This can be achieved by grouping and combining terms containing complex-conjugate poles, in the following way:

$$\begin{aligned} \frac{A}{1 - pz^{-1}} + \frac{A^*}{1 - p^* z^{-1}} &= \frac{A - Ap^* z^{-1} + A^* - A^* p z^{-1}}{1 - pz^{-1} - p^* z^{-1} + pp^* z^{-2}} \\ &= \frac{b_0 + b_1 z^{-1}}{1 + a_1 z^{-1} + a_2 z^{-2}} \end{aligned} \tag{3.4.43}$$

where

$$\begin{aligned} b_0 &= 2\,\text{Re}(A), & a_1 &= -2\,\text{Re}(p) \\ b_1 &= 2\,\text{Re}(Ap*), & a_2 &= |p|^2 \end{aligned} \tag{3.4.44}$$

are the desired coefficients. Obviously, any rational transform of the form (3.4.43) with coefficients given by (3.4.44), which is the case when $a_1^2 - 4a_2 < 0$, can be inverted using (3.4.34). By combining (3.4.41), (3.4.42), and (3.4.43) we obtain a

partial-fraction expansion for the z-transform with *distinct* poles that contains real coefficients. The general result is

$$X(z) = \sum_{k=0}^{M-N} c_k z^{-k} + \sum_{k=1}^{K_1} \frac{b_k}{1 + a_k z^{-1}} + \sum_{k=1}^{K_2} \frac{b_{0k} + b_{1k} z^{-1}}{1 + a_{1k} z^{-1} + a_{2k} z^{-2}} \tag{3.4.45}$$

where $K_1 + 2K_2 = N$. Obviously, if $M = N$, the first term is just a constant, and when $M < N$, this term vanishes. When there are also multiple poles, some additional higher-order terms should be included in (3.4.45).

An alternative form is obtained by expressing $X(z)$ as a product of simple terms as in (3.4.40). However, the complex-conjugate poles and zeros should be combined to avoid complex coefficients in the decomposition. Such combinations result in second-order rational terms of the following form:

$$\frac{(1 - z_k z^{-1})(1 - z_k^* z^{-1})}{(1 - p_k z^{-1})(1 - p_k^* z^{-1})} = \frac{1 + b_{1k} z^{-1} + b_{2k} z^{-2}}{1 + a_{1k} z^{-1} + a_{2k} z^{-2}} \tag{3.4.46}$$

where

$$\begin{aligned} b_{1k} &= -2\,\mathrm{Re}(z_k), & a_{1k} &= -2\,\mathrm{Re}(p_k) \\ b_{2k} &= |z_k|^2, & a_{2k} &= |p_k|^2 \end{aligned} \tag{3.4.47}$$

Assuming for simplicity that $M = N$, we see that $X(z)$ can be decomposed in the following way:

$$X(z) = b_0 \prod_{k=1}^{K_1} \frac{1 + b_k z^{-1}}{1 + a_k z^{-1}} \prod_{k=1}^{K_2} \frac{1 + b_{1k} z^{-1} + b_{2k} z^{-2}}{1 + a_{1k} z^{-1} + a_{2k} z^{-2}} \tag{3.4.48}$$

where $N = K_1 + 2K_2$. We will return to these important forms in Chapters 9 and 10.

3.5 Analysis of Linear Time-Invariant Systems in the z-Domain

In Section 3.3.3 we introduced the system function of a linear time-invariant system and related it to the unit sample response and to the difference equation description of systems. In this section we describe the use of the system function in the determination of the response of the system to some excitation signal. In Section 3.6.3, we extend this method of analysis to nonrelaxed systems. Our attention is focused on the important class of pole–zero systems represented by linear constant-coefficient difference equations with arbitrary initial conditions.

We also consider the topic of stability of linear time-invariant systems and describe a test for determining the stability of a system based on the coefficients of the denominator polynomial in the system function. Finally, we provide a detailed analysis of second-order systems, which form the basic building blocks in the realization of higher-order systems.

3.5.1 Response of Systems with Rational System Functions

Let us consider a pole–zero system described by the general linear constant-coefficient difference equation in (3.3.7) and the corresponding system function in (3.3.8). We represent $H(z)$ as a ratio of two polynomials $B(z)/A(z)$, where $B(z)$ is the numerator polynomial that contains the zeros of $H(z)$, and $A(z)$ is the denominator polynomial that determines the poles of $H(z)$. Furthermore, let us assume that the input signal $x(n)$ has a rational z-transform $X(z)$ of the form

$$X(z) = \frac{N(z)}{Q(z)} \tag{3.5.1}$$

This assumption is not overly restrictive, since, as indicated previously, most signals of practical interest have rational z-transforms.

If the system is initially relaxed, that is, the initial conditions for the difference equation are zero, $y(-1) = y(-2) = \cdots = y(-N) = 0$, the z-transform of the output of the system has the form

$$Y(z) = H(z)X(z) = \frac{B(z)N(z)}{A(z)Q(z)} \tag{3.5.2}$$

Now suppose that the system contains simple poles $p_1, p_2, \ldots, p_N$ and the z-transform of the input signal contains poles $q_1, q_2, \ldots, q_L$, where $p_k \neq q_m$ for all $k = 1, 2, \ldots, N$ and $m = 1, 2, \ldots, L$. In addition, we assume that the zeros of the numerator polynomials $B(z)$ and $N(z)$ do not coincide with the poles $\{p_k\}$ and $\{q_k\}$, so that there is no pole–zero cancellation. Then a partial-fraction expansion of $Y(z)$ yields

$$Y(z) = \sum_{k=1}^{N} \frac{A_k}{1 - p_k z^{-1}} + \sum_{k=1}^{L} \frac{Q_k}{1 - q_k z^{-1}} \tag{3.5.3}$$

The inverse transform of $Y(z)$ yields the output signal from the system in the form

$$y(n) = \sum_{k=1}^{N} A_k (p_k)^n u(n) + \sum_{k=1}^{L} Q_k (q_k)^n u(n) \tag{3.5.4}$$

We observe that the output sequence $y(n)$ can be subdivided into two parts. The first part is a function of the poles $\{p_k\}$ of the system and is called the *natural response* of the system. The influence of the input signal on this part of the response is through the scale factors $\{A_k\}$. The second part of the response is a function of the poles $\{q_k\}$ of the input signal and is called the *forced response* of the system. The influence of the system on this response is exerted through the scale factors $\{Q_k\}$.

We should emphasize that the scale factors $\{A_k\}$ and $\{Q_k\}$ are functions of both sets of poles $\{p_k\}$ and $\{q_k\}$. For example, if $X(z) = 0$ so that the input is zero, then $Y(z) = 0$, and consequently, the output is zero. Clearly, then, the natural response of the system is zero. This implies that the natural response of the system is different from the zero-input response.

When $X(z)$ and $H(z)$ have one or more poles in common or when $X(z)$ and/or $H(z)$ contain multiple-order poles, then $Y(z)$ will have multiple-order poles. Consequently, the partial-fraction expansion of $Y(z)$ will contain factors of the form $1/(1 - p_l z^{-1})^k$, $k = 1, 2, \ldots, m$, where m is the pole order. The inversion of these factors will produce terms of the form $n^{k-1}p_l^n$ in the output $y(n)$ of the system, as indicated in Section 3.4.3.

3.5.2 Transient and Steady-State Responses

As we have seen from our previous discussion, the zero-state response of a system to a given input can be separated into two components, the natural response and the forced response. The natural response of a causal system has the form

$$y_{\text{nr}}(n) = \sum_{k=1}^{N} A_k(p_k)^n u(n) \tag{3.5.5}$$

where $\{p_k\}$, $k = 1, 2, \ldots, N$ are the poles of the system and $\{A_k\}$ are scale factors that depend on the initial conditions and on the characteristics of the input sequence.

If $|p_k| < 1$ for all k, then, $y_{\text{nr}}(n)$ decays to zero as n approaches infinity. In such a case we refer to the natural response of the system as the *transient response*. The rate at which $y_{\text{nr}}(n)$ decays toward zero depends on the magnitude of the pole positions. If all the poles have small magnitudes, the decay is very rapid. On the other hand, if one or more poles are located near the unit circle, the corresponding terms in $y_{\text{nr}}(n)$ will decay slowly toward zero and the transient will persist for a relatively long time.

The forced response of the system has the form

$$y_{\text{fr}}(n) = \sum_{k=1}^{L} Q_k(q_k)^n u(n) \tag{3.5.6}$$

where $\{q_k\}$, $k = 1, 2, \ldots, L$ are the poles in the forcing function and $\{Q_k\}$ are scale factors that depend on the input sequence and on the characteristics of the system. If all the poles of the input signal fall inside the unit circle, $y_{\text{fr}}(n)$ will decay toward zero as n approaches infinity, just as in the case of the natural response. This should not be surprising since the input signal is also a transient signal. On the other hand, when the causal input signal is a sinusoid, the poles fall on the unit circle and consequently, the forced response is also a sinusoid that persists for all $n \geq 0$. In this case, the forced response is called the *steady-state response* of the system. Thus, for the system to sustain a steady-state output for $n \geq 0$, the input signal must persist for all $n \geq 0$.

The following example illustrates the presence of the steady-state response.

EXAMPLE 3.5.1

Determine the transient and steady-state responses of the system characterized by the difference equation

$$y(n) = 0.5y(n-1) + x(n)$$

when the input signal is $x(n) = 10\cos(\pi n/4)u(n)$. The system is initially at rest (i.e., it is relaxed).

Solution. The system function for this system is

$$H(z) = \frac{1}{1 - 0.5z^{-1}}$$

and therefore the system has a pole at $z = 0.5$. The z-transform of the input signal is (from Table 3.3)

$$X(z) = \frac{10(1 - (1/\sqrt{2})z^{-1})}{1 - \sqrt{2}z^{-1} + z^{-2}}$$

Consequently,

$$\begin{aligned} Y(z) &= H(z)X(z) \\ &= \frac{10(1 - (1/\sqrt{2})z^{-1})}{(1 - 0.5z^{-1})(1 - e^{j\pi/4}z^{-1})(1 - e^{-j\pi/4}z^{-1})} \\ &= \frac{6.3}{1 - 0.5z^{-1}} + \frac{6.78e^{-j28.7^\circ}}{1 - e^{j\pi/4}z^{-1}} + \frac{6.78e^{j28.7^\circ}}{1 - e^{-j\pi/4}z^{-1}} \end{aligned}$$

The natural or transient response is

$$y_{tr}(n) = 6.3(0.5)^n u(n)$$

and the forced or steady-state response is

$$\begin{aligned} y_{ss}(n) &= [6.78e^{-j28.7^\circ}(e^{j\pi n/4}) + 6.78e^{j28.7^\circ}e^{-j\pi n/4}]u(n) \\ &= 13.56\cos\left(\frac{\pi}{4}n - 28.7^\circ\right)u(n) \end{aligned}$$

Thus we see that the steady-state response persists for all $n \geq 0$, just as the input signal persists for all $n \geq 0$.

3.5.3 Causality and Stability

As defined previously, a causal linear time-invariant system is one whose unit sample response $h(n)$ satisfies the condition

$$h(n) = 0, \qquad n < 0$$

We have also shown that the ROC of the z-transform of a causal sequence is the exterior of a circle. Consequently, *a linear time-invariant system is causal if and only if the ROC of the system function is the exterior of a circle of radius $r < \infty$, including the point $z = \infty$.*

The stability of a linear time-invariant system can also be expressed in terms of the characteristics of the system function. As we recall from our previous discussion, a necessary and sufficient condition for a linear time-invariant system to be BIBO stable is

$$\sum_{n=-\infty}^{\infty} |h(n)| < \infty$$

In turn, this condition implies that $H(z)$ must contain the unit circle within its ROC.

Indeed, since

$$H(z) = \sum_{n=-\infty}^{\infty} h(n)z^{-n}$$

it follows that

$$|H(z)| \leq \sum_{n=-\infty}^{\infty} |h(n)z^{-n}| = \sum_{n=-\infty}^{\infty} |h(n)||z^{-n}|$$

When evaluated on the unit circle (i.e., $|z| = 1$),

$$|H(z)| \leq \sum_{n=-\infty}^{\infty} |h(n)|$$

Hence, if the system is BIBO stable, the unit circle is contained in the ROC of $H(z)$. The converse is also true. Therefore, a *linear time-invariant system* is BIBO stable if and only if the ROC of the system function includes the unit circle.

We should stress, however, that the conditions for causality and stability are different and that one does not imply the other. For example, a causal system may be stable or unstable, just as a noncausal system may be stable or unstable. Similarly, an unstable system may be either causal or noncausal, just as a stable system may be causal or noncausal.

For a causal system, however, the condition on stability can be narrowed to some extent. Indeed, a causal system is characterized by a system function $H(z)$ having as a ROC the exterior of some circle of radius r. For a stable system, the ROC must include the unit circle. Consequently, a causal and stable system must have a system function that converges for $|z| > r < 1$. Since the ROC cannot contain any poles of $H(z)$, it follows that a *causal linear time-invariant system* is BIBO stable if and only if all the poles of $H(z)$ are inside the unit circle.

EXAMPLE 3.5.2

A linear time-invariant system is characterized by the system function

$$H(z) = \frac{3 - 4z^{-1}}{1 - 3.5z^{-1} + 1.5z^{-2}}$$
$$= \frac{1}{1 - \frac{1}{2}z^{-1}} + \frac{2}{1 - 3z^{-1}}$$

Specify the ROC of $H(z)$ and determine $h(n)$ for the following conditions:

(a) The system is stable.

(b) The system is causal.

(c) The system is anticausal.