ElCoM

Electrical | Computer | Mechatronics

www.ELCOM-HU.com

إرادة .. ثقة .. تغيير

ElCoM
Electrical | Computer | Mechatronics
ELCOM-HU.COM

# 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** xvii

**Introduction** 1

**1.1 Signals, Systems, and Signal Processing** 2
- 1.1.1 Basic Elements of a Digital Signal Processing System 4
- 1.1.2 Advantages of Digital over Analog Signal Processing 5

**1.2 Classification of Signals** 6
- 1.2.1 Multichannel and Multidimensional Signals 6
- 1.2.2 Continuous-Time Versus Discrete-Time Signals 9
- 1.2.3 Continuous-Valued Versus Discrete-Valued Signals 10
- 1.2.4 Deterministic Versus Random Signals 11

**1.3 The Concept of Frequency in Continuous-Time and Discrete-Time Signals** 12
- 1.3.1 Continuous-Time Sinusoidal Signals 12
- 1.3.2 Discrete-Time Sinusoidal Signals 14
- 1.3.3 Harmonically Related Complex Exponentials 17

**1.4 Analog-to-Digital and Digital-to-Analog Conversion** 19
- 1.4.1 Sampling of Analog Signals 21
- 1.4.2 The Sampling Theorem 26
- 1.4.3 Quantization of Continuous-Amplitude Signals 31
- 1.4.4 Quantization of Sinusoidal Signals 34
- 1.4.5 Coding of Quantized Samples 35
- 1.4.6 Digital-to-Analog Conversion 36
- 1.4.7 Analysis of Digital Signals and Systems Versus Discrete-Time Signals and Systems 36

**1.5 Summary and References** 37

**Problems** 37

**2 Discrete-Time Signals and Systems** 41

**2.1 Discrete-Time Signals** 42
2.1.1 Some Elementary Discrete-Time Signals 43
2.1.2 Classification of Discrete-Time Signals 45
2.1.3 Simple Manipulations of Discrete-Time Signals 50

**2.2 Discrete-Time Systems** 53
2.2.1 Input–Output Description of Systems 54
2.2.2 Block Diagram Representation of Discrete-Time Systems 57
2.2.3 Classification of Discrete-Time Systems 59
2.2.4 Interconnection of Discrete-Time Systems 67

**2.3 Analysis of Discrete-Time Linear Time-Invariant Systems** 69
2.3.1 Techniques for the Analysis of Linear Systems 69
2.3.2 Resolution of a Discrete-Time Signal into Impulses 71
2.3.3 Response of LTI Systems to Arbitrary Inputs: The Convolution Sum 73
2.3.4 Properties of Convolution and the Interconnection of LTI Systems 80
2.3.5 Causal Linear Time-Invariant Systems 83
2.3.6 Stability of Linear Time-Invariant Systems 85
2.3.7 Systems with Finite-Duration and Infinite-Duration Impulse Response 88

**2.4 Discrete-Time Systems Described by Difference Equations** 89
2.4.1 Recursive and Nonrecursive Discrete-Time Systems 90
2.4.2 Linear Time-Invariant Systems Characterized by Constant-Coefficient Difference Equations 93
2.4.3 Solution of Linear Constant-Coefficient Difference Equations 98
2.4.4 The Impulse Response of a Linear Time-Invariant Recursive System 106

**2.5 Implementation of Discrete-Time Systems** 109
2.5.1 Structures for the Realization of Linear Time-Invariant Systems 109
2.5.2 Recursive and Nonrecursive Realizations of FIR Systems 113

**2.6 Correlation of Discrete-Time Signals** 116
2.6.1 Crosscorrelation and Autocorrelation Sequences 118
2.6.2 Properties of the Autocorrelation and Crosscorrelation Sequences 120
2.6.3 Correlation of Periodic Sequences 123
2.6.4 Input–Output Correlation Sequences 125

**2.7 Summary and References** 128

**Problems** 129

**3 The $z$-Transform and Its Application to the Analysis of LTI Systems** **147**

**3.1 The $z$-Transform** **147**
3.1.1 The Direct $z$-Transform 147
3.1.2 The Inverse $z$-Transform 156

**3.2 Properties of the $z$-Transform** **157**

**3.3 Rational $z$-Transforms** **170**
3.3.1 Poles and Zeros 170
3.3.2 Pole Location and Time-Domain Behavior for Causal Signals 174
3.3.3 The System Function of a Linear Time-Invariant System 177

**3.4 Inversion of the $z$-Transform** **180**
3.4.1 The Inverse $z$-Transform by Contour Integration 180
3.4.2 The Inverse $z$-Transform by Power Series Expansion 182
3.4.3 The Inverse $z$-Transform by Partial-Fraction Expansion 184
3.4.4 Decomposition of Rational $z$-Transforms 192

**3.5 Analysis of Linear Time-Invariant Systems in the $z$-Domain** **193**
3.5.1 Response of Systems with Rational System Functions 194
3.5.2 Transient and Steady-State Responses 195
3.5.3 Causality and Stability 196
3.5.4 Pole–Zero Cancellations 198
3.5.5 Multiple-Order Poles and Stability 200
3.5.6 Stability of Second-Order Systems 201

**3.6 The One-sided $z$-Transform** **205**
3.6.1 Definition and Properties 206
3.6.2 Solution of Difference Equations 210
3.6.3 Response of Pole–Zero Systems with Nonzero Initial Conditions 211

**3.7 Summary and References** **214**

**Problems** **214**

**4 Frequency Analysis of Signals** **224**

**4.1 Frequency Analysis of Continuous-Time Signals** **225**
4.1.1 The Fourier Series for Continuous-Time Periodic Signals 226
4.1.2 Power Density Spectrum of Periodic Signals 230
4.1.3 The Fourier Transform for Continuous-Time Aperiodic Signals 234
4.1.4 Energy Density Spectrum of Aperiodic Signals 238

**4.2 Frequency Analysis of Discrete-Time Signals** **241**
4.2.1 The Fourier Series for Discrete-Time Periodic Signals 241
4.2.2 Power Density Spectrum of Periodic Signals 245
4.2.3 The Fourier Transform of Discrete-Time Aperiodic Signals 248
4.2.4 Convergence of the Fourier Transform 251
4.2.5 Energy Density Spectrum of Aperiodic Signals 254
4.2.6 Relationship of the Fourier Transform to the $z$-Transform 259
4.2.7 The Cepstrum 261
4.2.8 The Fourier Transform of Signals with Poles on the Unit Circle 262
4.2.9 Frequency-Domain Classification of Signals: The Concept of Bandwidth 265
4.2.10 The Frequency Ranges of Some Natural Signals 267

**4.3 Frequency-Domain and Time-Domain Signal Properties** **268**

**4.4 Properties of the Fourier Transform for Discrete-Time Signals** **271**
4.4.1 Symmetry Properties of the Fourier Transform 272
4.4.2 Fourier Transform Theorems and Properties 279

**4.5 Summary and References** **291**

**Problems** **292**

**5 Frequency-Domain Analysis of LTI Systems** **300**

**5.1 Frequency-Domain Characteristics of Linear Time-Invariant Systems** **300**
5.1.1 Response to Complex Exponential and Sinusoidal Signals: The Frequency Response Function 301
5.1.2 Steady-State and Transient Response to Sinusoidal Input Signals 310
5.1.3 Steady-State Response to Periodic Input Signals 311
5.1.4 Response to Aperiodic Input Signals 312

**5.2 Frequency Response of LTI Systems** **314**
5.2.1 Frequency Response of a System with a Rational System Function 314
5.2.2 Computation of the Frequency Response Function 317

**5.3 Correlation Functions and Spectra at the Output of LTI Systems** **321**
5.3.1 Input–Output Correlation Functions and Spectra 322
5.3.2 Correlation Functions and Power Spectra for Random Input Signals 323

**5.4 Linear Time-Invariant Systems as Frequency-Selective Filters** **326**
5.4.1 Ideal Filter Characteristics 327
5.4.2 Lowpass, Highpass, and Bandpass Filters 329
5.4.3 Digital Resonators 335
5.4.4 Notch Filters 339
5.4.5 Comb Filters 341

5.4.6 All-Pass Filters 345
5.4.7 Digital Sinusoidal Oscillators 347

**5.5 Inverse Systems and Deconvolution 349**

5.5.1 Invertibility of Linear Time-Invariant Systems 350
5.5.2 Minimum-Phase, Maximum-Phase, and Mixed-Phase Systems 354
5.5.3 System Identification and Deconvolution 358
5.5.4 Homomorphic Deconvolution 360

**5.6 Summary and References 362**

**Problems 363**

## 6 Sampling and Reconstruction of Signals 384

**6.1 Ideal Sampling and Reconstruction of Continuous-Time Signals 384**

**6.2 Discrete-Time Processing of Continuous-Time Signals 395**

**6.3 Analog-to-Digital and Digital-to-Analog Converters 401**

6.3.1 Analog-to-Digital Converters 401
6.3.2 Quantization and Coding 403
6.3.3 Analysis of Quantization Errors 406
6.3.4 Digital-to-Analog Converters 408

**6.4 Sampling and Reconstruction of Continuous-Time Bandpass Signals 410**

6.4.1 Uniform or First-Order Sampling 411
6.4.2 Interleaved or Nonuniform Second-Order Sampling 416
6.4.3 Bandpass Signal Representations 422
6.4.4 Sampling Using Bandpass Signal Representations 426

**6.5 Sampling of Discrete-Time Signals 427**

6.5.1 Sampling and Interpolation of Discrete-Time Signals 427
6.5.2 Representation and Sampling of Bandpass Discrete-Time Signals 430

**6.6 Oversampling A/D and D/A Converters 433**

6.6.1 Oversampling A/D Converters 433
6.6.2 Oversampling D/A Converters 439

**6.7 Summary and References 440**

**Problems 440**

**7 The Discrete Fourier Transform: Its Properties and Applications 449**

**7.1 Frequency-Domain Sampling: The Discrete Fourier Transform 449**

7.1.1 Frequency-Domain Sampling and Reconstruction of Discrete-Time Signals 449
7.1.2 The Discrete Fourier Transform (DFT) 454
7.1.3 The DFT as a Linear Transformation 459
7.1.4 Relationship of the DFT to Other Transforms 461

**7.2 Properties of the DFT 464**

7.2.1 Periodicity, Linearity, and Symmetry Properties 465
7.2.2 Multiplication of Two DFTs and Circular Convolution 471
7.2.3 Additional DFT Properties 476

**7.3 Linear Filtering Methods Based on the DFT 480**

7.3.1 Use of the DFT in Linear Filtering 481
7.3.2 Filtering of Long Data Sequences 485

**7.4 Frequency Analysis of Signals Using the DFT 488**

**7.5 The Discrete Cosine Transform 495**

7.5.1 Forward DCT 495
7.5.2 Inverse DCT 497
7.5.3 DCT as an Orthogonal Transform 498

**7.6 Summary and References 501**

**Problems 502**

**8 Efficient Computation of the DFT: Fast Fourier Transform Algorithms 511**

**8.1 Efficient Computation of the DFT: FFT Algorithms 511**

8.1.1 Direct Computation of the DFT 512
8.1.2 Divide-and-Conquer Approach to Computation of the DFT 513
8.1.3 Radix-2 FFT Algorithms 519
8.1.4 Radix-4 FFT Algorithms 527
8.1.5 Split-Radix FFT Algorithms 532
8.1.6 Implementation of FFT Algorithms 536

**8.2 Applications of FFT Algorithms 538**

8.2.1 Efficient Computation of the DFT of Two Real Sequences 538
8.2.2 Efficient Computation of the DFT of a $2N$-Point Real Sequence 539
8.2.3 Use of the FFT Algorithm in Linear Filtering and Correlation 540

**8.3 A Linear Filtering Approach to Computation of the DFT** **542**
8.3.1 The Goertzel Algorithm 542
8.3.2 The Chirp-$z$ Transform Algorithm 544
**8.4 Quantization Effects in the Computation of the DFT** **549**
8.4.1 Quantization Errors in the Direct Computation of the DFT 549
8.4.2 Quantization Errors in FFT Algorithms 552
**8.5 Summary and References** **555**
**Problems** **556**

**9 Implementation of Discrete-Time Systems** **563**

**9.1 Structures for the Realization of Discrete-Time Systems** **563**
**9.2 Structures for FIR Systems** **565**
9.2.1 Direct-Form Structure 566
9.2.2 Cascade-Form Structures 567
9.2.3 Frequency-Sampling Structures 569
9.2.4 Lattice Structure 574
**9.3 Structures for IIR Systems** **582**
9.3.1 Direct-Form Structures 582
9.3.2 Signal Flow Graphs and Transposed Structures 585
9.3.3 Cascade-Form Structures 589
9.3.4 Parallel-Form Structures 591
9.3.5 Lattice and Lattice-Ladder Structures for IIR Systems 594
**9.4 Representation of Numbers** **601**
9.4.1 Fixed-Point Representation of Numbers 601
9.4.2 Binary Floating-Point Representation of Numbers 605
9.4.3 Errors Resulting from Rounding and Truncation 608
**9.5 Quantization of Filter Coefficients** **613**
9.5.1 Analysis of Sensitivity to Quantization of Filter Coefficients 613
9.5.2 Quantization of Coefficients in FIR Filters 620
**9.6 Round-Off Effects in Digital Filters** **624**
9.6.1 Limit-Cycle Oscillations in Recursive Systems 624
9.6.2 Scaling to Prevent Overflow 629
9.6.3 Statistical Characterization of Quantization Effects in Fixed-Point Realizations of Digital Filters 631
**9.7 Summary and References** **640**
**Problems** **641**

**10 Design of Digital Filters** 654

10.1 General Considerations 654
- 10.1.1 Causality and Its Implications 655
- 10.1.2 Characteristics of Practical Frequency-Selective Filters 659

10.2 Design of FIR Filters 660
- 10.2.1 Symmetric and Antisymmetric FIR Filters 660
- 10.2.2 Design of Linear-Phase FIR Filters Using Windows 664
- 10.2.3 Design of Linear-Phase FIR Filters by the Frequency-Sampling Method 671
- 10.2.4 Design of Optimum Equiripple Linear-Phase FIR Filters 678
- 10.2.5 Design of FIR Differentiators 691
- 10.2.6 Design of Hilbert Transformers 693
- 10.2.7 Comparison of Design Methods for Linear-Phase FIR Filters 700

10.3 Design of IIR Filters From Analog Filters 701
- 10.3.1 IIR Filter Design by Approximation of Derivatives 703
- 10.3.2 IIR Filter Design by Impulse Invariance 707
- 10.3.3 IIR Filter Design by the Bilinear Transformation 712
- 10.3.4 Characteristics of Commonly Used Analog Filters 717
- 10.3.5 Some Examples of Digital Filter Designs Based on the Bilinear Transformation 727

10.4 Frequency Transformations 730
- 10.4.1 Frequency Transformations in the Analog Domain 730
- 10.4.2 Frequency Transformations in the Digital Domain 732

10.5 Summary and References 734

Problems 735

**11 Multirate Digital Signal Processing** 750

11.1 Introduction 751

11.2 Decimation by a Factor $D$ 755

11.3 Interpolation by a Factor $I$ 760

11.4 Sampling Rate Conversion by a Rational Factor $I/D$ 762

11.5 Implementation of Sampling Rate Conversion 766
- 11.5.1 Polyphase Filter Structures 766
- 11.5.2 Interchange of Filters and Downsamplers/Upsamplers 767
- 11.5.3 Sampling Rate Conversion with Cascaded Integrator Comb Filters 769
- 11.5.4 Polyphase Structures for Decimation and Interpolation Filters 771
- 11.5.5 Structures for Rational Sampling Rate Conversion 774

**11.6 Multistage Implementation of Sampling Rate Conversion** **775**

**11.7 Sampling Rate Conversion of Bandpass Signals** **779**

**11.8 Sampling Rate Conversion by an Arbitrary Factor** **781**

11.8.1 Arbitrary Resampling with Polyphase Interpolators 782

11.8.2 Arbitrary Resampling with Farrow Filter Structures 782

**11.9 Applications of Multirate Signal Processing** **784**

11.9.1 Design of Phase Shifters 784

11.9.2 Interfacing of Digital Systems with Different Sampling Rates 785

11.9.3 Implementation of Narrowband Lowpass Filters 786

11.9.4 Subband Coding of Speech Signals 787

**11.10 Digital Filter Banks** **790**

11.10.1 Polyphase Structures of Uniform Filter Banks 794

11.10.2 Transmultiplexers 796

**11.11 Two-Channel Quadrature Mirror Filter Bank** **798**

11.11.1 Elimination of Aliasing 799

11.11.2 Condition for Perfect Reconstruction 801

11.11.3 Polyphase Form of the QMF Bank 801

11.11.4 Linear Phase FIR QMF Bank 802

11.11.5 IIR QMF Bank 803

11.11.6 Perfect Reconstruction Two-Channel FIR QMF Bank 803

11.11.7 Two-Channel QMF Banks in Subband Coding 806

**11.12 $M$-Channel QMF Bank** **807**

11.12.1 Alias-Free and Perfect Reconstruction Condition 808

11.12.2 Polyphase Form of the $M$-Channel QMF Bank 808

**11.13 Summary and References** **813**

**Problems** **813**

**12 Linear Prediction and Optimum Linear Filters** **823**

**12.1 Random Signals, Correlation Functions, and Power Spectra** **823**

12.1.1 Random Processes 824

12.1.2 Stationary Random Processes 825

12.1.3 Statistical (Ensemble) Averages 825

12.1.4 Statistical Averages for Joint Random Processes 826

12.1.5 Power Density Spectrum 828

12.1.6 Discrete-Time Random Signals 829

12.1.7 Time Averages for a Discrete-Time Random Process 830

12.1.8 Mean-Ergodic Process 831

12.1.9 Correlation-Ergodic Processes 832

**12.2 Innovations Representation of a Stationary Random Process** 834
12.2.1 Rational Power Spectra 836
12.2.2 Relationships Between the Filter Parameters and the Autocorrelation Sequence 837
**12.3 Forward and Backward Linear Prediction** 838
12.3.1 Forward Linear Prediction 839
12.3.2 Backward Linear Prediction 841
12.3.3 The Optimum Reflection Coefficients for the Lattice Forward and Backward Predictors 845
12.3.4 Relationship of an AR Process to Linear Prediction 846
**12.4 Solution of the Normal Equations** 846
12.4.1 The Levinson–Durbin Algorithm 847
12.4.2 The Schur Algorithm 850
**12.5 Properties of the Linear Prediction-Error Filters** 855
**12.6 AR Lattice and ARMA Lattice-Ladder Filters** 858
12.6.1 AR Lattice Structure 858
12.6.2 ARMA Processes and Lattice-Ladder Filters 860
**12.7 Wiener Filters for Filtering and Prediction** 863
12.7.1 FIR Wiener Filter 864
12.7.2 Orthogonality Principle in Linear Mean-Square Estimation 866
12.7.3 IIR Wiener Filter 867
12.7.4 Noncausal Wiener Filter 872
**12.8 Summary and References** 873
**Problems** 874

**13 Adaptive Filters** 880

**13.1 Applications of Adaptive Filters** 880
13.1.1 System Identification or System Modeling 882
13.1.2 Adaptive Channel Equalization 883
13.1.3 Echo Cancellation in Data Transmission over Telephone Channels 887
13.1.4 Suppression of Narrowband Interference in a Wideband Signal 891
13.1.5 Adaptive Line Enhancer 895
13.1.6 Adaptive Noise Cancelling 896
13.1.7 Linear Predictive Coding of Speech Signals 897
13.1.8 Adaptive Arrays 900
**13.2 Adaptive Direct-Form FIR Filters—The LMS Algorithm** 902
13.2.1 Minimum Mean-Square-Error Criterion 903
13.2.2 The LMS Algorithm 905

13.2.3 Related Stochastic Gradient Algorithms 907
13.2.4 Properties of the LMS Algorithm 909

**13.3 Adaptive Direct-Form Filters—RLS Algorithms 916**
13.3.1 RLS Algorithm 916
13.3.2 The LDU Factorization and Square-Root Algorithms 921
13.3.3 Fast RLS Algorithms 923
13.3.4 Properties of the Direct-Form RLS Algorithms 925

**13.4 Adaptive Lattice-Ladder Filters 927**
13.4.1 Recursive Least-Squares Lattice-Ladder Algorithms 928
13.4.2 Other Lattice Algorithms 949
13.4.3 Properties of Lattice-Ladder Algorithms 950

**13.5 Summary and References 954**

**Problems 955**

## 14 Power Spectrum Estimation 960

**14.1 Estimation of Spectra from Finite-Duration Observations of Signals 961**
14.1.1 Computation of the Energy Density Spectrum 961
14.1.2 Estimation of the Autocorrelation and Power Spectrum of Random Signals: The Periodogram 966
14.1.3 The Use of the DFT in Power Spectrum Estimation 971

**14.2 Nonparametric Methods for Power Spectrum Estimation 974**
14.2.1 The Bartlett Method: Averaging Periodograms 974
14.2.2 The Welch Method: Averaging Modified Periodograms 975
14.2.3 The Blackman and Tukey Method: Smoothing the Periodogram 978
14.2.4 Performance Characteristics of Nonparametric Power Spectrum Estimators 981
14.2.5 Computational Requirements of Nonparametric Power Spectrum Estimates 984

**14.3 Parametric Methods for Power Spectrum Estimation 986**
14.3.1 Relationships Between the Autocorrelation and the Model Parameters 988
14.3.2 The Yule–Walker Method for the AR Model Parameters 990
14.3.3 The Burg Method for the AR Model Parameters 991
14.3.4 Unconstrained Least-Squares Method for the AR Model Parameters 994
14.3.5 Sequential Estimation Methods for the AR Model Parameters 995
14.3.6 Selection of AR Model Order 996
14.3.7 MA Model for Power Spectrum Estimation 997
14.3.8 ARMA Model for Power Spectrum Estimation 999
14.3.9 Some Experimental Results 1001

**14.4 Filter Bank Methods** **1009**
14.4.1 Filter Bank Realization of the Periodogram 1010
14.4.2 Minimum Variance Spectral Estimates 1012
**14.5 Eigenanalysis Algorithms for Spectrum Estimation** **1015**
14.5.1 Pisarenko Harmonic Decomposition Method 1017
14.5.2 Eigen-decomposition of the Autocorrelation Matrix for Sinusoids in White Noise 1019
14.5.3 MUSIC Algorithm 1021
14.5.4 ESPRIT Algorithm 1022
14.5.5 Order Selection Criteria 1025
14.5.6 Experimental Results 1026
**14.6 Summary and References** **1029**
**Problems** **1030**
**A Random Number Generators** **1041**
**B Tables of Transition Coefficients for the Design of Linear-Phase FIR Filters** **1047**
**References and Bibliography** **1053**
**Answers to Selected Problems** **1067**
**Index** **1077**

# Preface

ElCoM
Electrical | Computer | Mechatronics
ELCOM-HU.COM

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

ElCoM
Electrical | Computer | Mechatronics
ELCOM-HU.COM

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

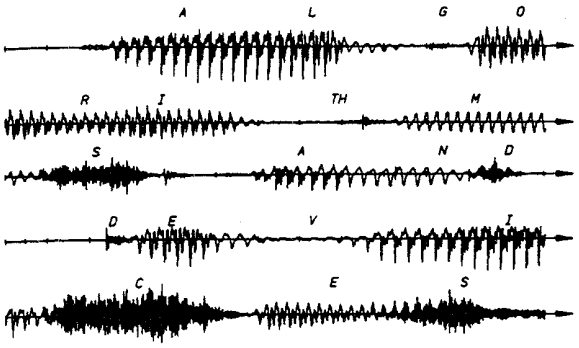



**Figure 1.1.1**
Example of a speech signal.

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.

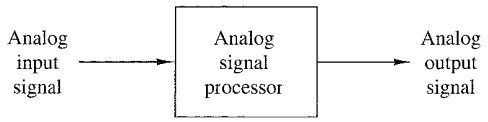



**Figure 1.1.2**
Analog signal processing.

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-

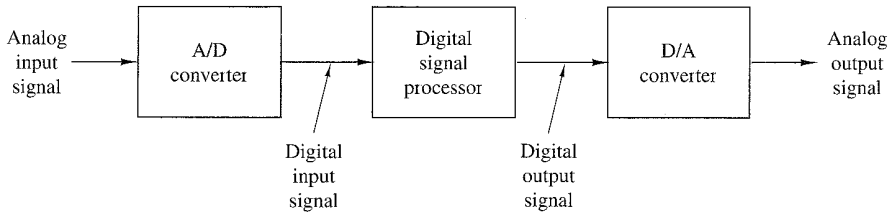



**Figure 1.1.3** Block diagram of a digital signal processing system.

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

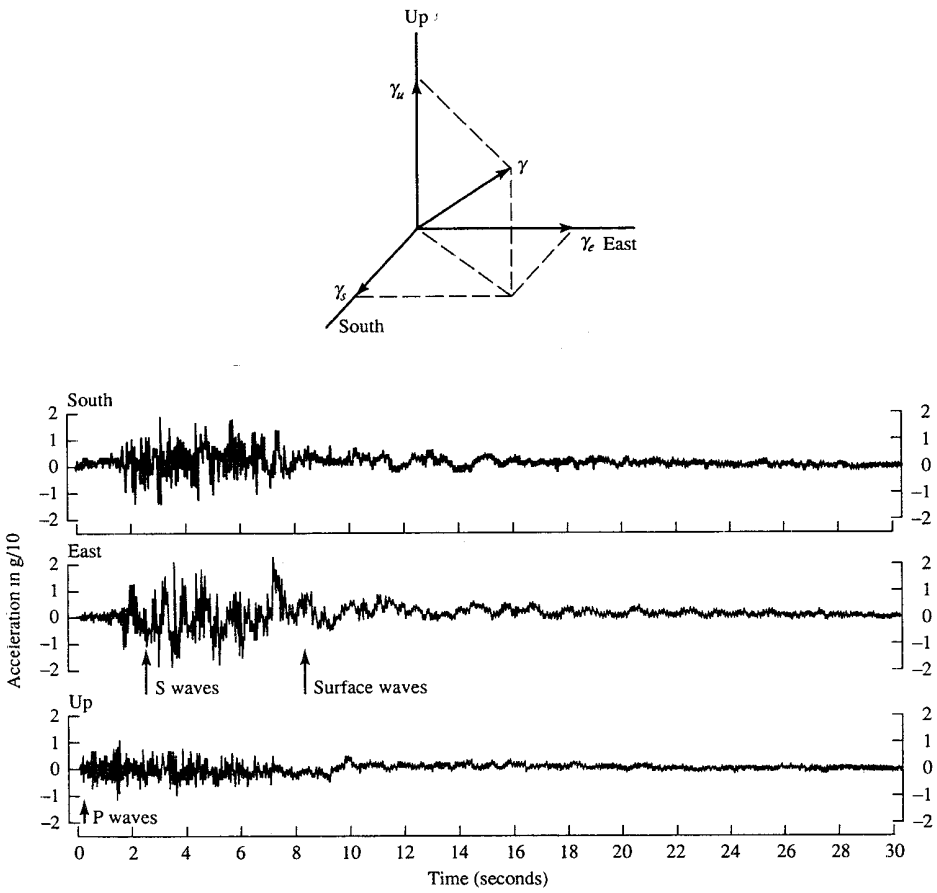



**Figure 1.2.1** Three components of ground acceleration measured a few kilometers from the epicenter of an earthquake. (From *Earthquakes*, by B. A. Bold, ©1988 by W. H. Freeman and Company. Reprinted with permission of the publisher.)

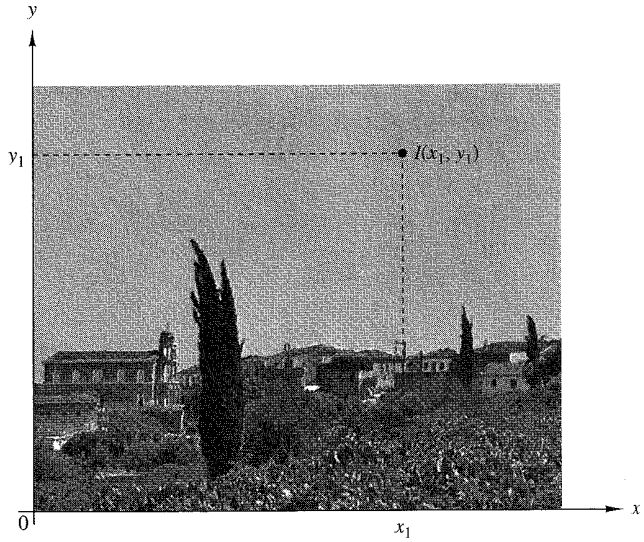



**Figure 1.2.2**
Example of a two-dimensional signal.

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 $k$th 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 $\infty$. 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

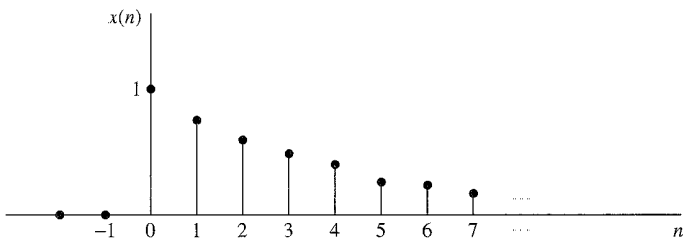



**Figure 1.2.3** Graphical representation of the discrete time signal $x(n) = 0.8^n$ for $n > 0$ and $x(n) = 0$ for $n < 0$.

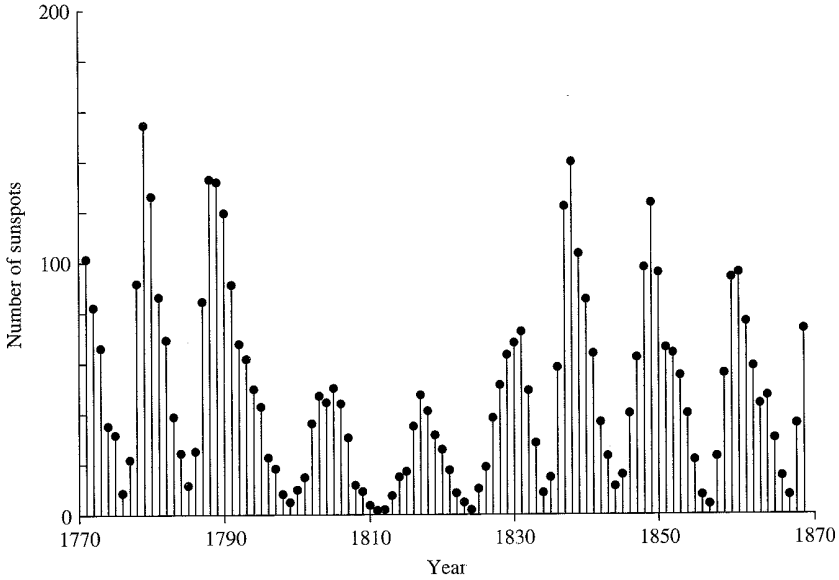



**Figure 1.2.4** Wölfer annual sunspot numbers (1770–1869).

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

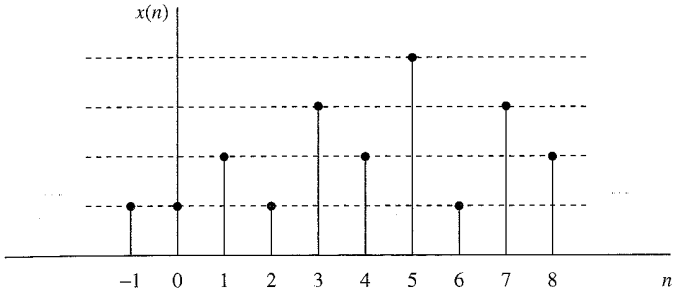



**Figure 1.2.5** Digital signal with four different amplitude values.

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, $\Omega$ is the *frequency* in radians per second (rad/s), and $\theta$ is the *phase* in radians. Instead of $\Omega$, 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.

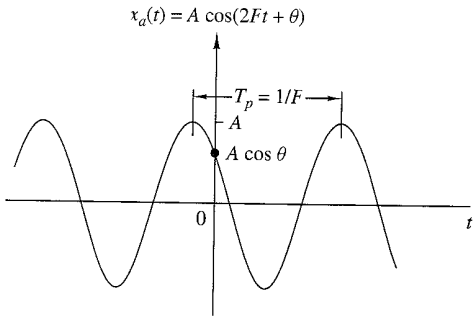



**Figure 1.3.1**
Example of an analog sinusoidal signal.

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 $-\infty < F < \infty$.

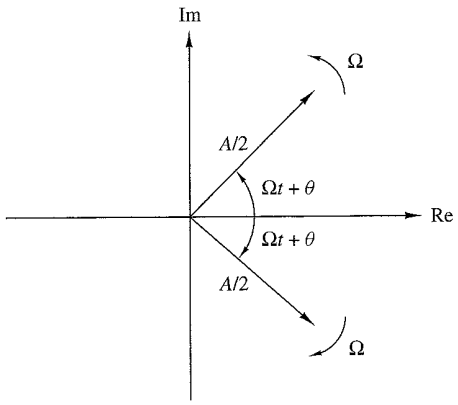



**Figure 1.3.2**
Representation of a cosine function by a pair of complex-conjugate exponentials (phasors).

### 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, $\omega$ is the *frequency* in radians per sample, and $\theta$ is the *phase* in radians.

If instead of $\omega$ 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 $\theta = \pi/3$.

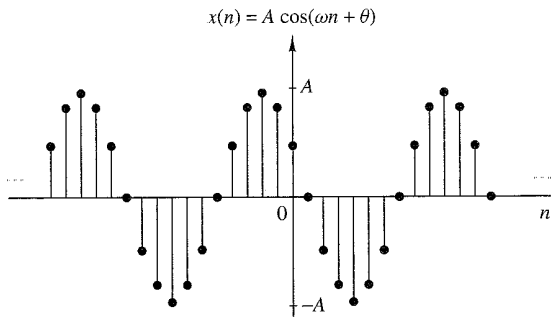



**Figure 1.3.3**
Example of a discrete-time sinusoidal signal ($\omega = \pi/6$ and $\theta = \pi/3$).

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 $2\pi$ 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 $\Omega$ 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 $\pi$. 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 $\omega_1$ varies from $\pi$ to $2\pi$, $\omega_2$ varies from $\pi$ to 0. It can be easily seen that

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

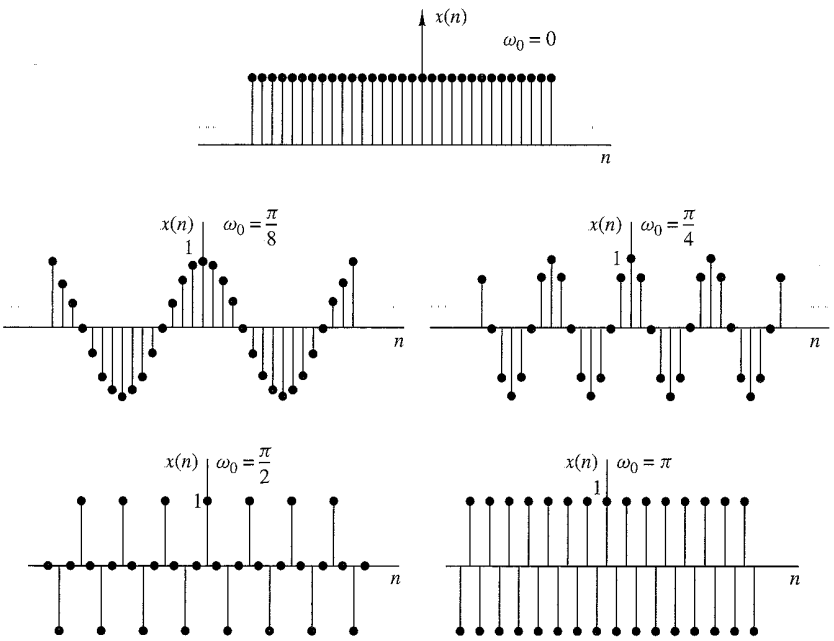



**Figure 1.3.4** Signal $x(n) = \cos \omega_0 n$ for various values of the frequency $\omega_0$.

Hence $\omega_2$ is an alias of $\omega_1$. 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 $\omega_0$ of a discrete-time sinusoid from $\pi$ to $2\pi$, 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 $2\pi$ 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 $2\pi$. 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

$$s_k(t) = e^{jk\Omega_0 t} = e^{j2\pi k F_0 t} \qquad k = 0, \pm 1, \pm 2, \ldots \tag{1.3.16}$$

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 $k_1 \ne k_2$, then $s_{k1}(t) \ne s_{k2}(t)$.

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 $k$th 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 $k$th 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

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

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 $k$th 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 $\theta$ 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 $\theta$ 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*.

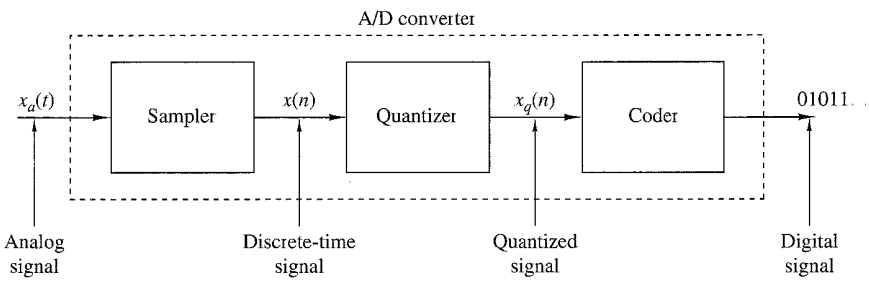



**Figure 1.4.1** Basic parts of an analog-to-digital (A/D) converter.

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

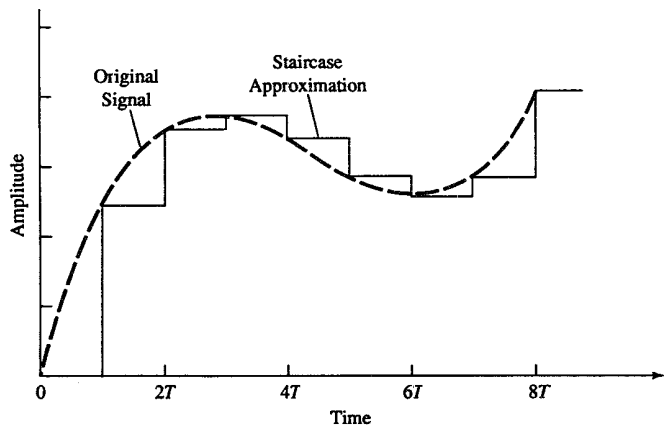



**Figure 1.4.2** Zero-order hold digital-to-analog (D/A) conversion.

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 $\Omega$) for analog signals and the frequency variable $f$ (or $\omega$) for discrete-time signals. To establish this relationship, consider an analog sinusoidal signal of the form

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

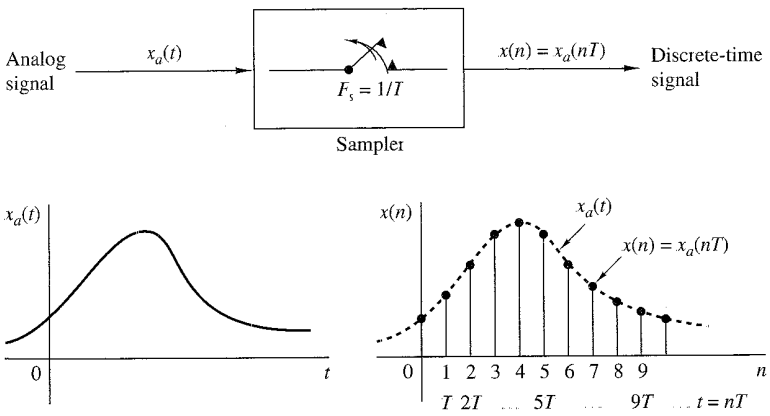



**Figure 1.4.3** Periodic sampling of an analog signal.

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 $\Omega$ 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 $\Omega$ and $\omega$. Periodic sampling of a continuous-time signal implies a mapping of the infinite frequency range for the variable $F$ (or $\Omega$) into a finite frequency range for the variable $f$ (or $\omega$). 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 | | $\frac{\text{radians}}{\text{sample}}$ $\frac{\text{cycles}}{\text{sample}}$ |
| | $\xrightarrow{\omega = \Omega T,\ f = F/F_s}$ | $-\pi \le \omega \le \pi$ |
| | | $-\frac{1}{2} \le f \le \frac{1}{2}$ |
| | $\xleftarrow{\Omega = \omega/T,\ F = f \cdot F_s}$ | |
| $-\infty < \Omega < \infty$ | | $-\pi/T \le \Omega \le \pi/T$ |
| $-\infty < F < \infty$ | | $-F_2/2 \le F \le F_s/2$ |

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 $\Omega$ 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.

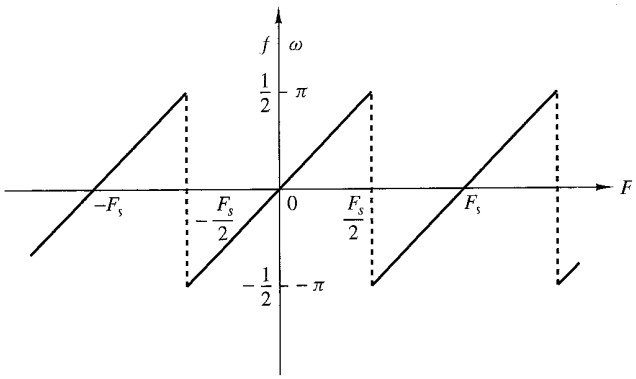



**Figure 1.4.4** Relationship between the continuous-time and discrete-time frequency variables in the case of periodic sampling.

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*.

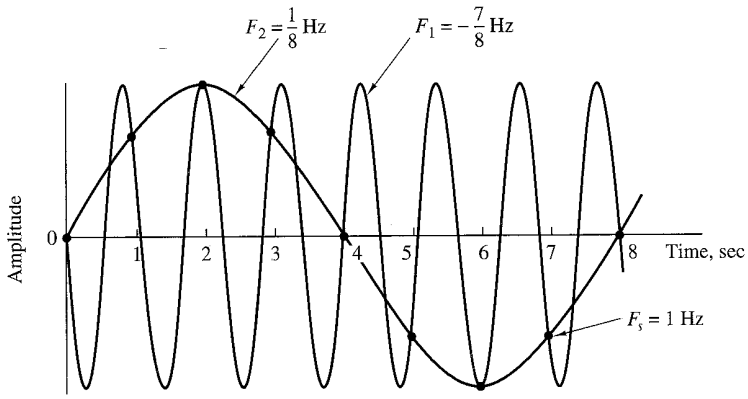



**Figure 1.4.5** Illustration of aliasing.

## 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.

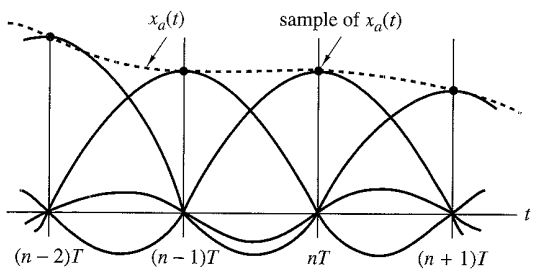



**Figure 1.4.6**
Ideal D/A conversion (interpolation).

### 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 $\theta$. 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 $\pi$, 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 $\theta$ 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

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

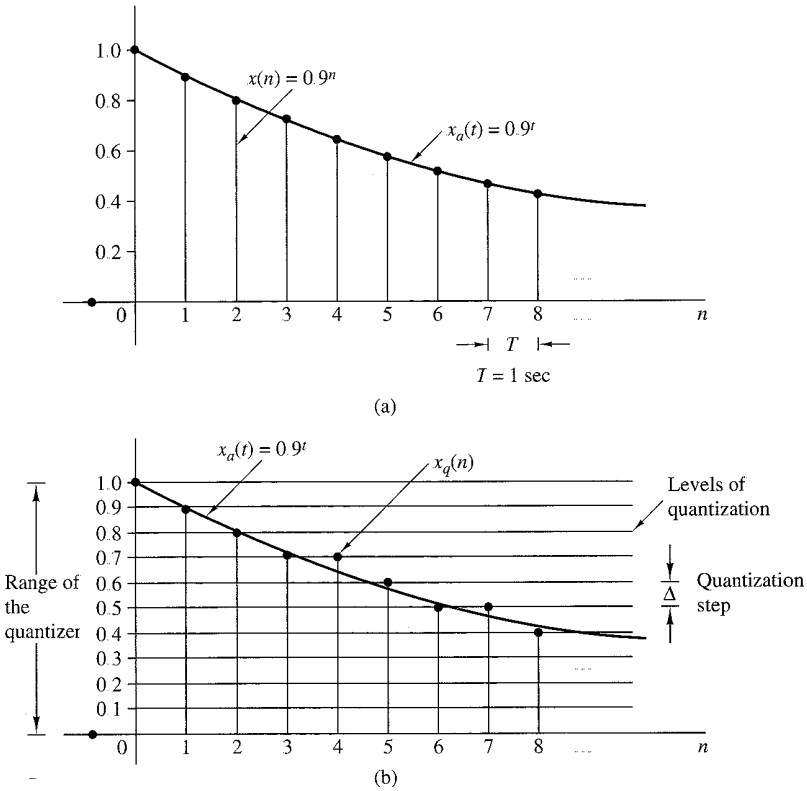



**Figure 1.4.7** Illustration of quantization.

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 $\Delta$ 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,

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

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.

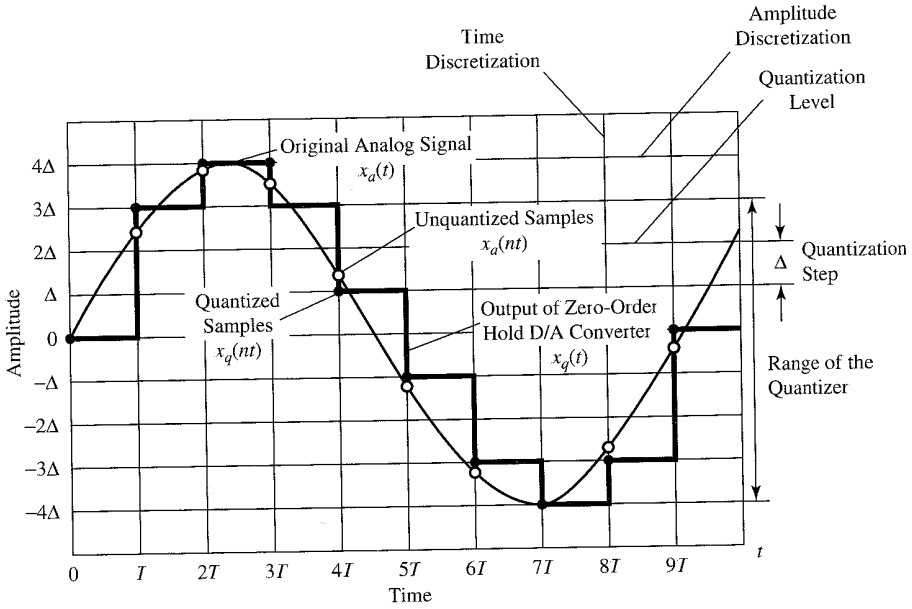



**Figure 1.4.8** Sampling and quantization of a sinusoidal signal.

### 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

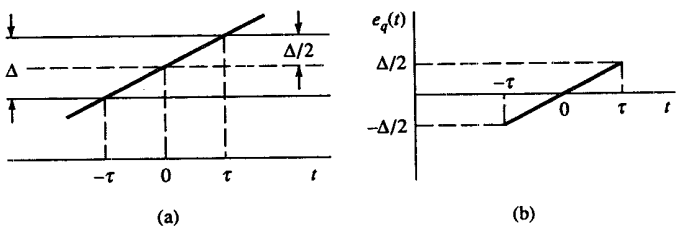



**Figure 1.4.9** The quantization error $e_q(t) = x_a(t) - x_q(t)$.

corresponding quantization error $e_q(t) = x_a(t) - x_q(t)$ is shown in Fig. 1.4.9. In Fig. 1.4.9, $\tau$ 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.

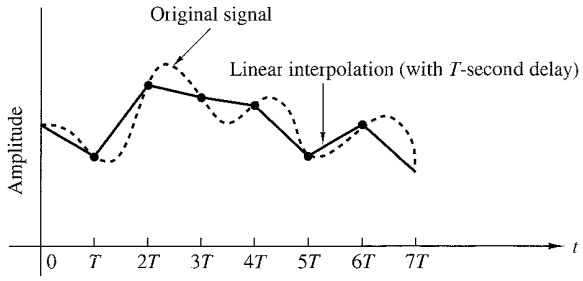



**Figure 1.4.10**
Linear point connector (with $T$-second delay).

### 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 $N_p = N/\mathrm{GCD}(k, N)$, 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 $\Delta$?

**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 $F_s/2$.

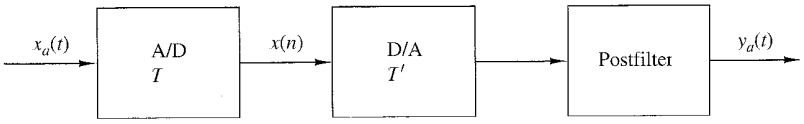



**Figure P1.11**

**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.

ElCoM
Electrical | Computer | Mechatronics
ELCOM-HU.COM

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 "$n$th 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

| $n$ | ... | $-2$ | $-1$ | 0 | 1 | 2 | 3 | 4 | 5 | ... |
|---|---|---|---|---|---|---|---|---|---|---|
| $x(n)$ | ... | 0 | 0 | 0 | 1 | 4 | 1 | 0 | 0 | ... |

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

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

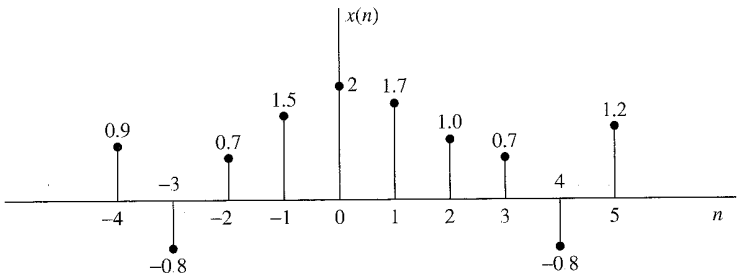



**Figure 2.1.1** Graphical representation of a discrete-time signal.

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.

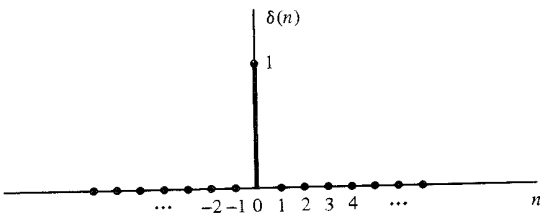



**Figure 2.1.2**
Graphical representation of the unit sample signal.

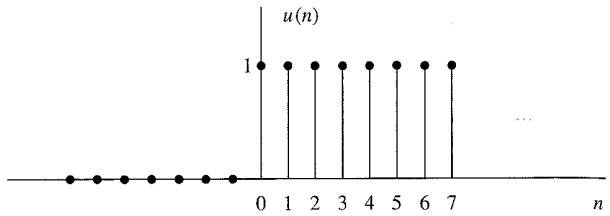



**Figure 2.1.3**
Graphical representation of the unit step signal.

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 $\theta$ 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

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

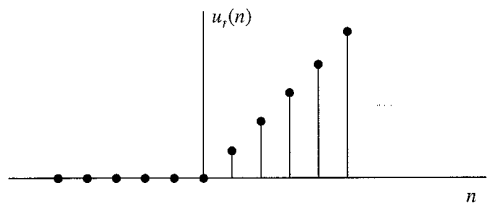



**Figure 2.1.4**
Graphical representation of the unit ramp signal.

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 $\theta$ is simply the frequency of the sinusoid, previously denoted by the (normalized) frequency variable $\omega$. 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 $2\pi$ from $\phi(n)$ before plotting. The subtraction of multiples of $2\pi$ from $\phi(n)$ is equivalent to interpreting the function $\phi(n)$ as $\phi(n)$, modulo $2\pi$.

### 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

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

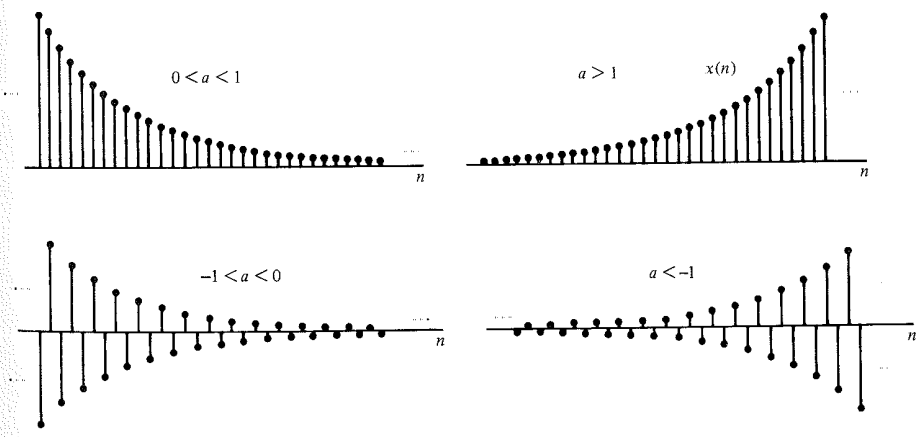



**Figure 2.1.5** Graphical representation of exponential signals.

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

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

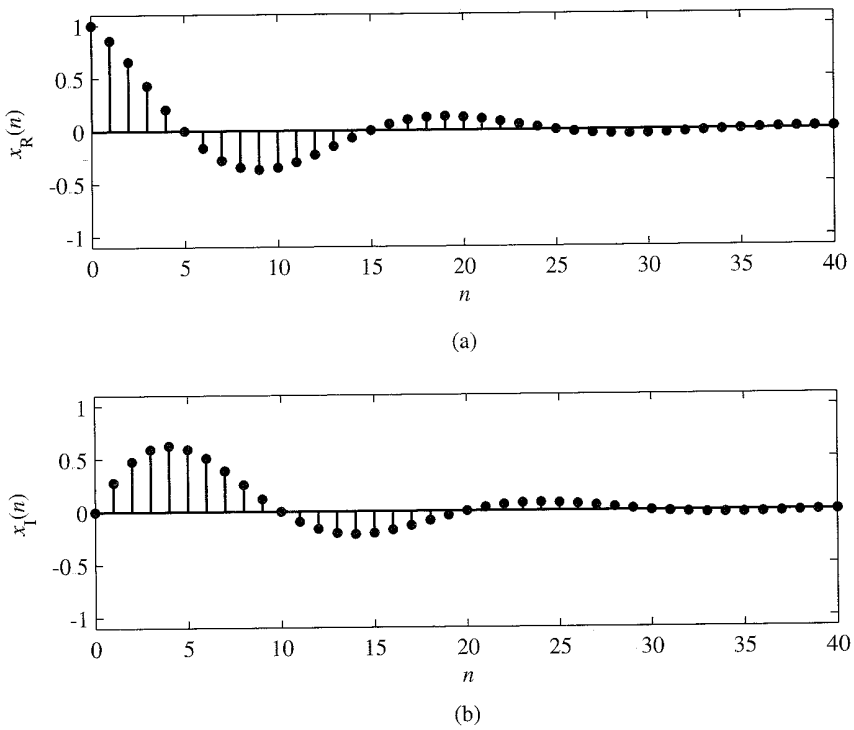



**Figure 2.1.6** Graph of the real and imaginary components of a complex-valued exponential signal.

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

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

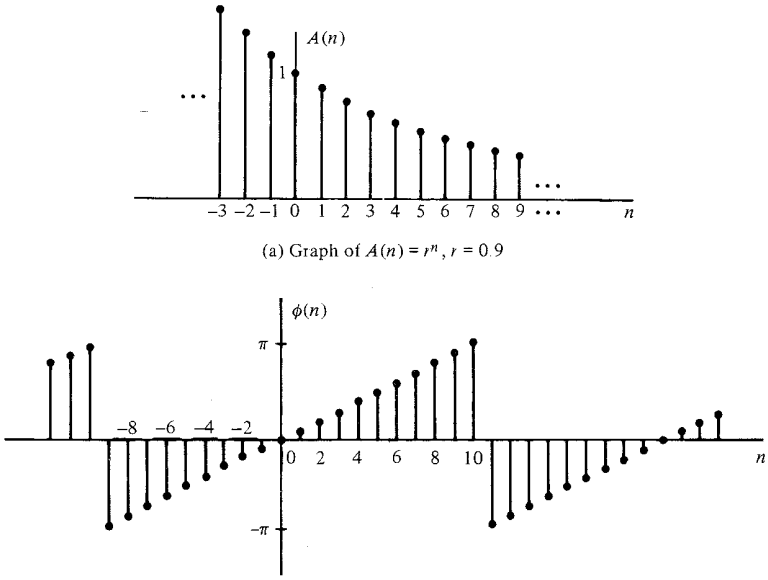
Clearly, if $E$ is finite, $P = 0$. On the other hand, if $E$ is infinite, the average power $P$ may be either finite or infinite. If $P$ is finite (and nonzero), the signal is called a *power signal*. The following example illustrates such a signal.

EXAMPLE 2.1.1

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

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

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 $2\pi$ 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 $2\pi$ 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.

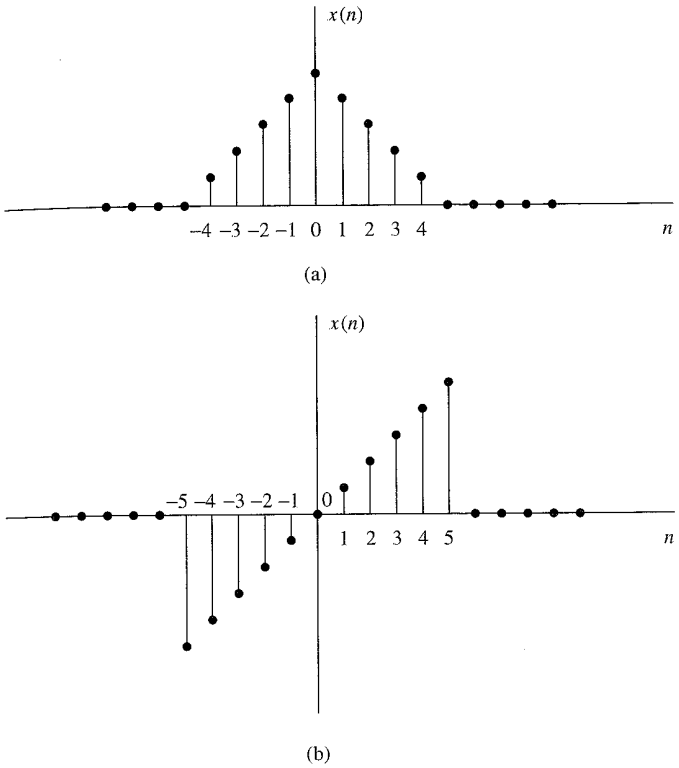



**Figure 2.1.8** Example of even (a) and odd (b) signals.

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.

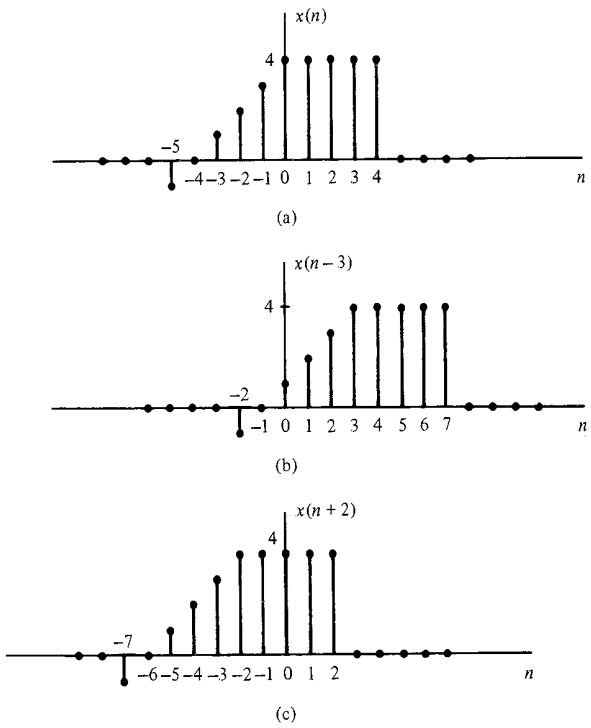



**Figure 2.1.9** Graphical representation of a signal, and its delayed and advanced versions.

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).

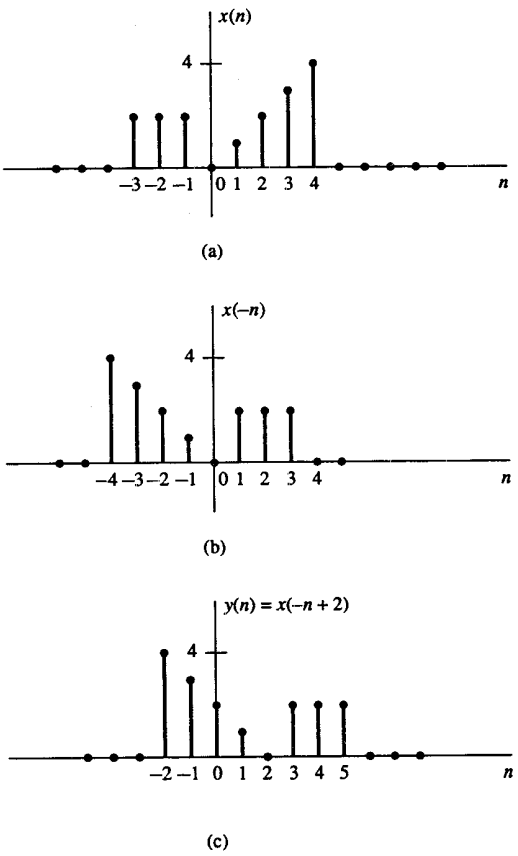



**Figure 2.1.10**
Graphical illustration of the folding and shifting operations.

**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 $\mu n$, where $\mu$ 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.

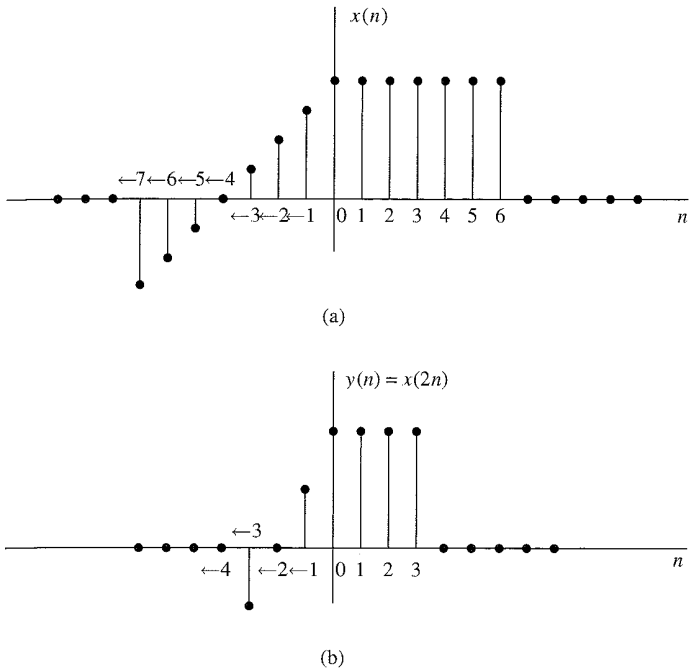



**Figure 2.1.11** Graphical illustration of down-sampling operation.

*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

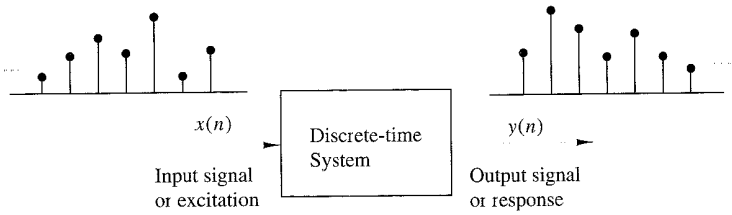



**Figure 2.2.1** Block diagram representation of a discrete-time system.

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

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

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

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

**(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*.

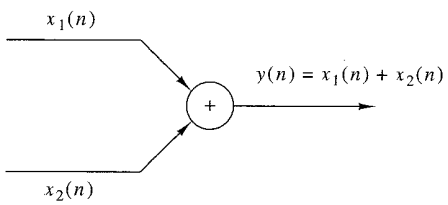



**Figure 2.2.2**
Graphical representation of an adder.

**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.

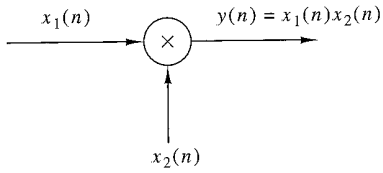



**Figure 2.2.4**
Graphical representation of a signal multiplier.

**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.

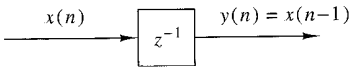



**Figure 2.2.5**
Graphical representation of the unit delay element.

**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.

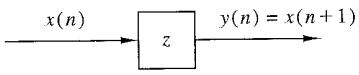



**Figure 2.2.6**
Graphical representation of the unit advance element.

## 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.

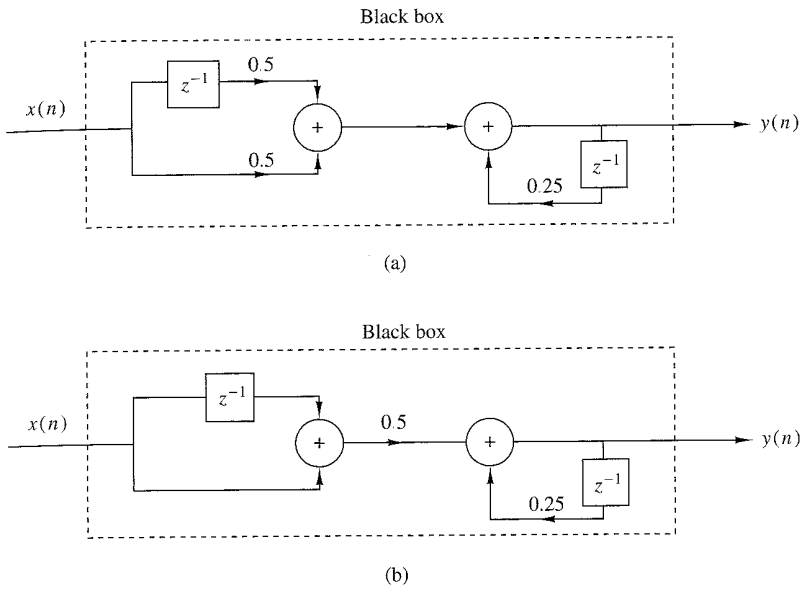



**Figure 2.2.7** Block diagram realizations of the system $y(n) = 0.25y(n-1) + 0.5x(n) + 0.5x(n-1)$.

**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.

ELCoM
Electrical | Computer | Mechatronics
ELCOM-HU.COM

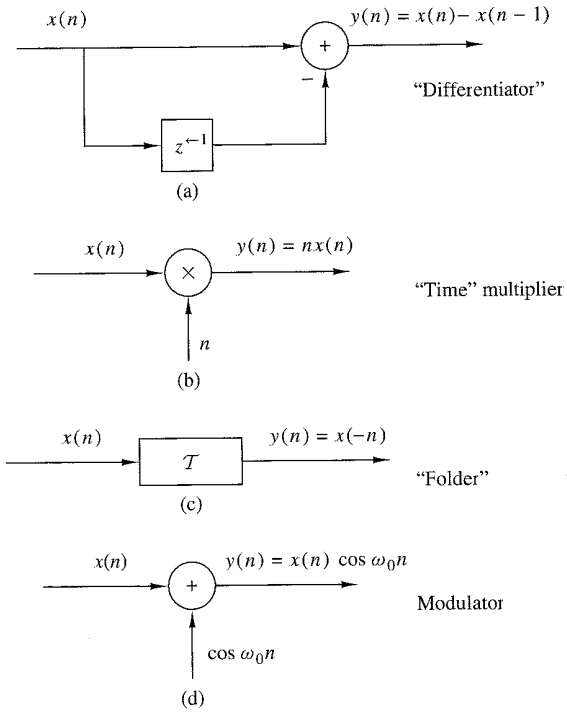



**Figure 2.2.8** Examples of a time-invariant (a) and some time-variant systems (b)–(d).

**(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

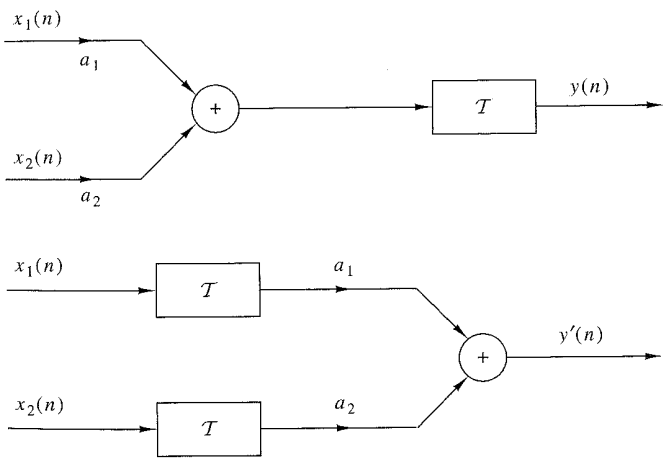



**Figure 2.2.9** Graphical representation of the superposition principle. $\mathcal{T}$ is linear if and only if $y(n) = y'(n)$.

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

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 $F[\cdot]$ 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

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

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

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

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.

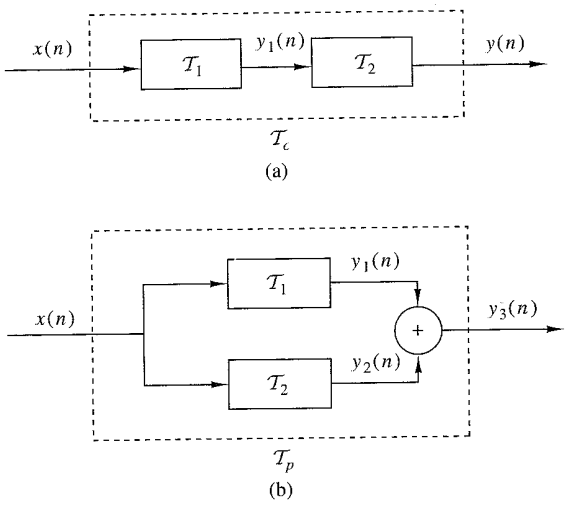



**Figure 2.2.10**
Cascade (a) and parallel (b) interconnections of systems.

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

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

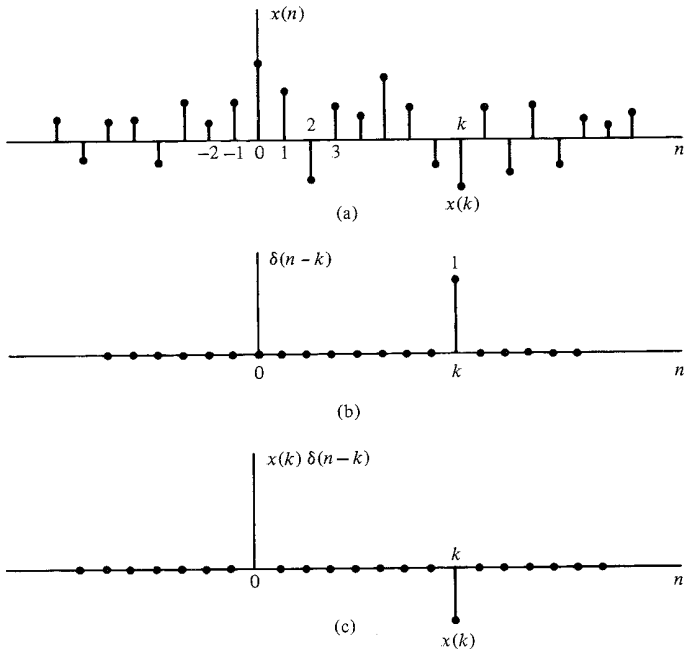



**Figure 2.3.1** Multiplication of a signal $x(n)$ with a shifted unit sample sequence.

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),

$$y(1) = \sum_{h=-\infty}^{\infty} x(k)h(1-k) \tag{2.3.23}$$

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

$$y(1) = \sum_{k=-\infty}^{\infty} v_1(k) = 8$$

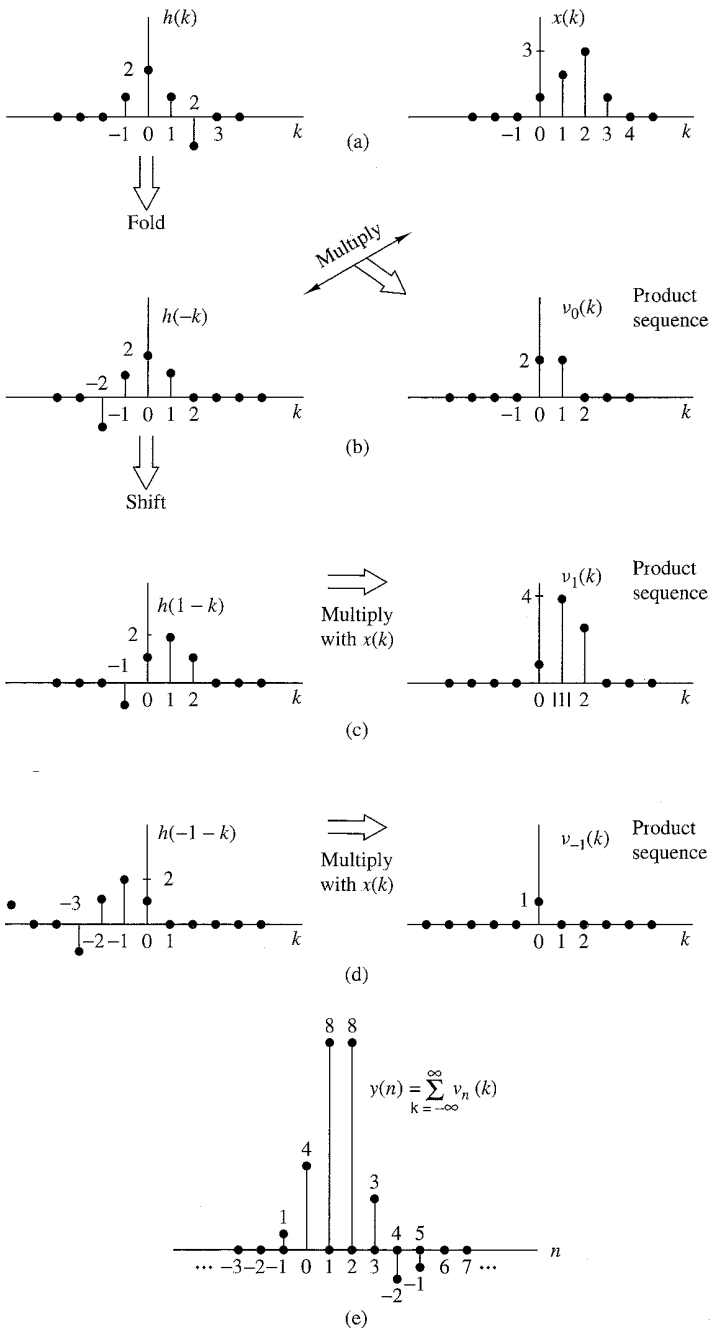



**Figure 2.3.2** Graphical computation of convolution.

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

$$h(n) = a^{nu}(n), \ |a|1$$

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

$$\begin{aligned} y(n) &= 1 + a + a^2 + \cdots + a^n \\ &= \frac{1 - a^{n+1}}{1 - a} \end{aligned} \tag{2.3.29}$$

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

$$y(\infty) = \lim_{n\to\infty} y(n) = \frac{1}{1-a} \tag{2.3.30}$$

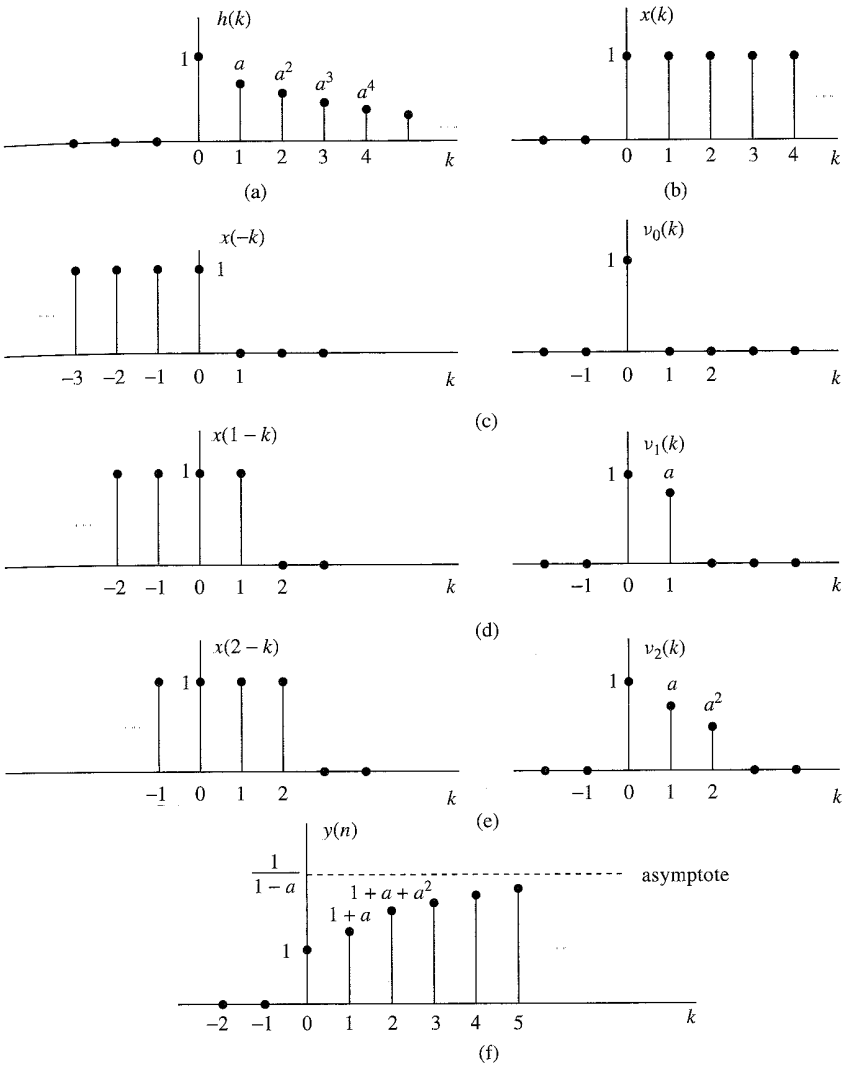



**Figure 2.3.3** Graphical computation of convolution in Example 2.3.3.

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.

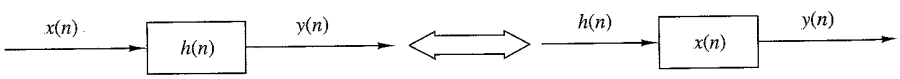



**Figure 2.3.4** Interpretation of the commutative property of convolution.

**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.

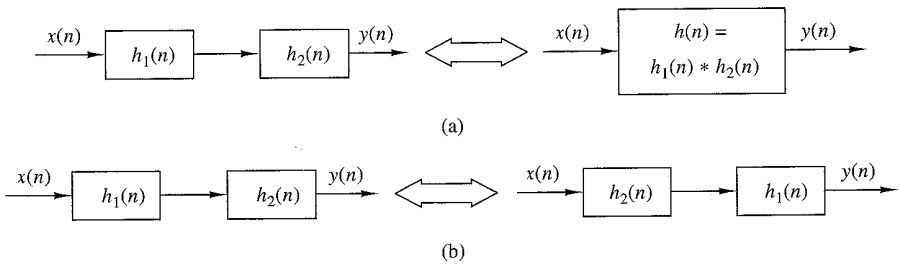



**Figure 2.3.5** Implications of the associative (a) and the associative and commutative (b) properties of convolution.

### 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

$$\begin{aligned} v_n(k) &= h_1(k)h_2(n-k) \\ &= (\frac{1}{2})^k(\frac{1}{4})^{n-k} \end{aligned}$$

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

$$\begin{aligned} h(n) &= \sum_{k=0}^{n}(\frac{1}{2})^k(\frac{1}{4})^{n-k} \\ &= (\frac{1}{4})^n \sum_{k=0}^{n} 2^k \\ &= (\frac{1}{4})^n(2^{n+1} - 1) \\ &= (\frac{1}{2})^n[2 - (\frac{1}{2})^n], \quad n \geq 0 \end{aligned}$$

---

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,

$$h(n) = h_1(n) * h_2(n) * \cdots * h_L(n) \tag{2.3.35}$$

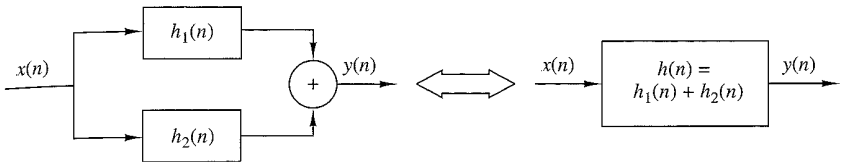



**Figure 2.3.6** Interpretation of the distributive property of convolution: two LTI systems connected in parallel can be replaced by a single system with $h(n) = h_1(n) + h_2(n)$.

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 $x(n_0)$, $x(n_0 - 1)$, ..., 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 $x(n_0+1)$, $x(n_0+2)$, .... 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., $x(n-M)$, $x(n-M-1)$, ...]. 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 $y(n-1)$, $y(n-2), \ldots$ is called a recursive system.

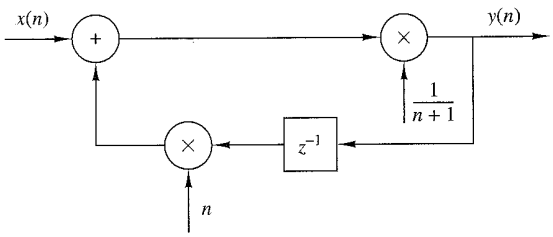



**Figure 2.4.1**
Realization of a recursive cumulative averaging system.

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.

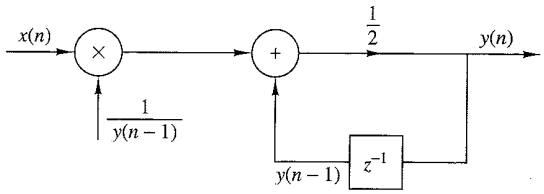



**Figure 2.4.2** Realization of the square-root system.

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.

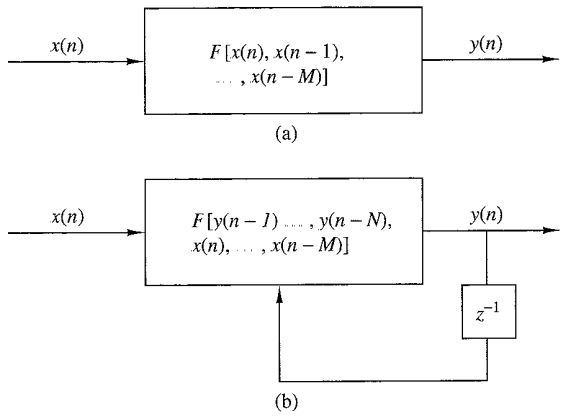



**Figure 2.4.3**
Basic form for a causal and realizable (a) nonrecursive and (b) recursive system.

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.

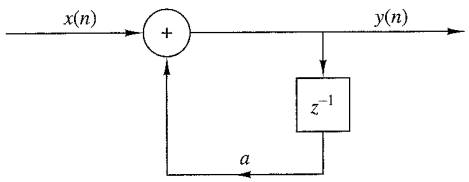



**Figure 2.4.4**
Block diagram realization of a simple recursive system.

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

$$\begin{aligned}
y(0) &= ay(-1) + x(0) \\
y(1) &= ay(0) + x(1) = a^2 y(-1) + ax(0) + x(1) \\
y(2) &= ay(1) + x(2) = a^3 y(-1) + a^2 x(0) + ax(1) + x(2) \\
&\vdots \qquad \vdots \\
y(n) &= ay(n-1) + x(n) \\
&= a^{n+1} y(-1) + a^n x(0) + a^{n-1} x(1) + \cdots + ax(n-1) + x(n)
\end{aligned}$$

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

$$\begin{aligned} y_{zs}(n) &= \sum_{k=0}^{n} a^k[c_1x_1(n-k) + c_2x_2(n-k)] \\ &= c_1\sum_{k=0}^{n} a^k x_1(n-k) + c_2\sum_{k=0}^{n} a^k x_2(n-k) \\ &= c_1y_{zs}^{(1)}(n) + c_2y_{zs}^{(2)}(n) \end{aligned}$$

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

$$y(n) + a_1 y(n-1) = x(n) \tag{2.4.18}$$

**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

$$y(0) = -a_1 y(-1)$$

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 $\lambda_1$ 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

$$\begin{aligned} y(0) &= 3y(-1) + 4y(-2) + 1 \\ y(1) &= 3y(0) + 4y(-1) + 6 \\ &= 13y(-1) + 12y(-2) + 9 \end{aligned}$$

On the other hand, (2.4.35) evaluated at $n = 0$ and $n = 1$ yields

$$\begin{aligned} y(0) &= C_1 + C_2 \\ y(1) &= -C_1 + 4C_2 + \frac{24}{5} \end{aligned}$$

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

$$\begin{aligned} y_{zs}(n) &= \sum_{k=0}^{n} a^k \delta(n-k) \\ &= a^n, \qquad n \geq 0 \end{aligned}$$

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 $N$th-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 $N$th-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

$$w(n) = -a_1 w(n-1) + x(n) \tag{2.5.4}$$

$$y(n) = b_0 w(n) + b_1 w(n-1) \tag{2.5.5}$$

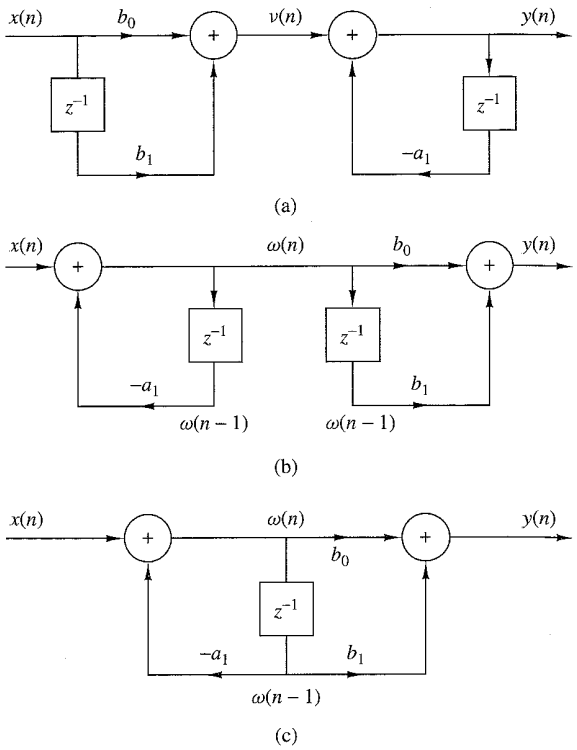



**Figure 2.5.1**
Steps in converting from the direct form I realization in (a) to the direct form II realization in (c).

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

$$y(n) = \sum_{k=0}^{M} b_k w(n-k) \tag{2.5.10}$$

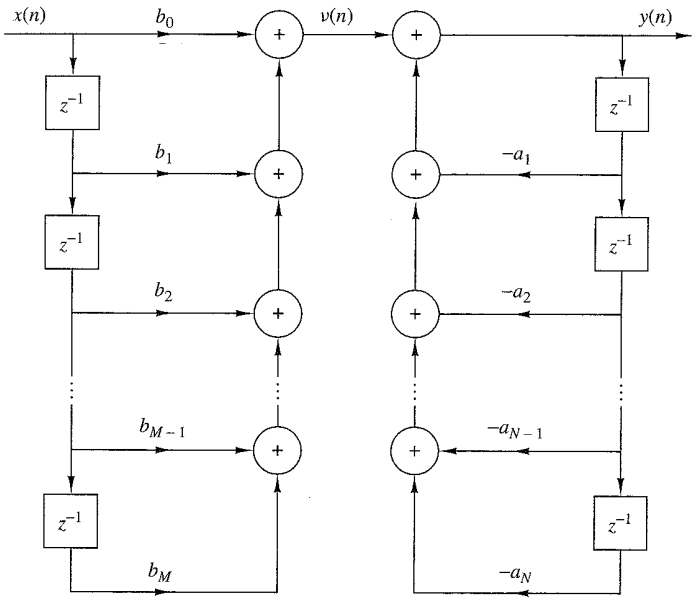



**Figure 2.5.2** Direct form I structure of the system described by (2.5.6).

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

$$y(n) = \sum_{k=0}^{M} b_k x(n-k) \tag{2.5.1}$$

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

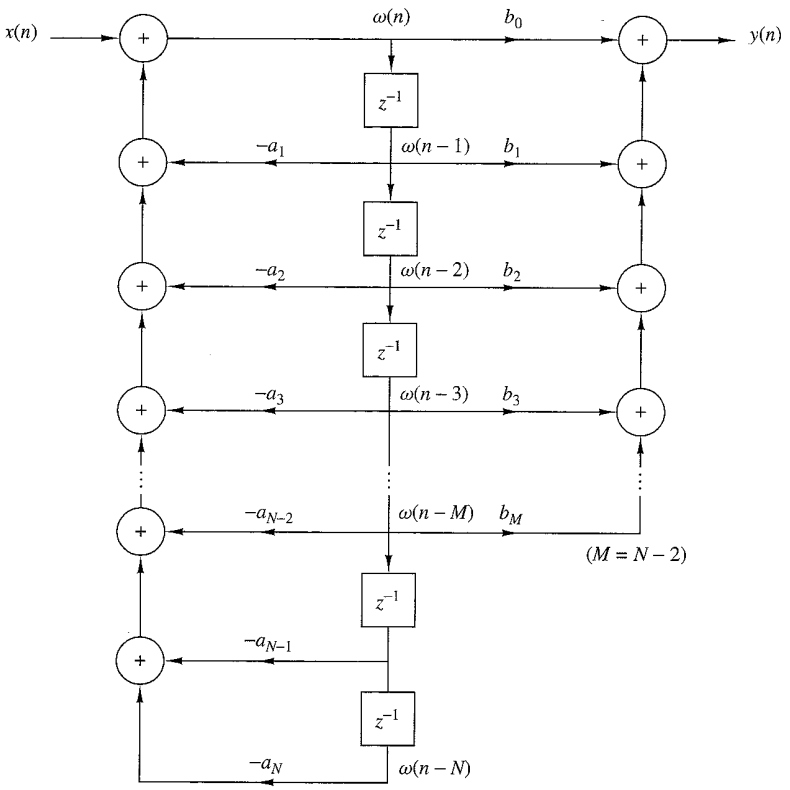



**Figure 2.5.3** Direct form II structure for the system described by (2.5.6).

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

$$y(n) = F[y(n-1), \ldots, y(n-N), x(n), \ldots, x(n-M)] \tag{2.5.17}$$

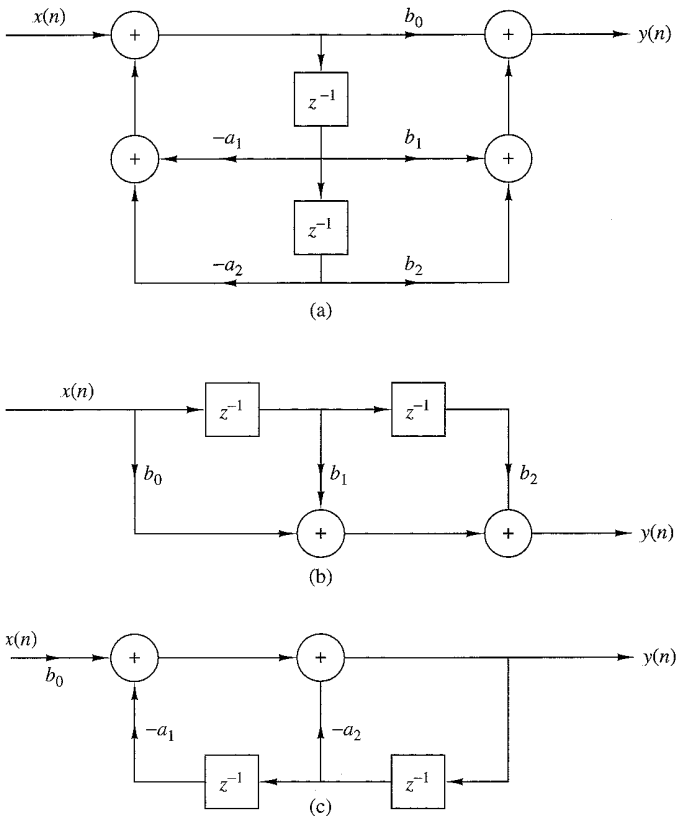



**Figure 2.5.4** Structures for the realization of second-order systems: (a) general second-order system; (b) FIR system; (c) "purely recursive system."

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)

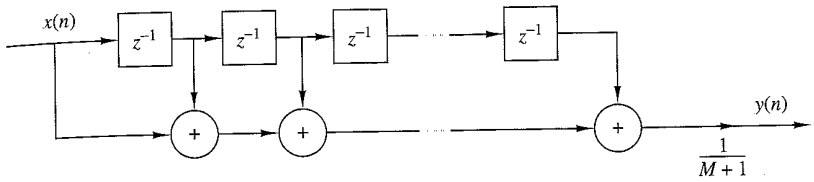



**Figure 2.5.5** Nonrecursive realization of an FIR moving average system.

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.

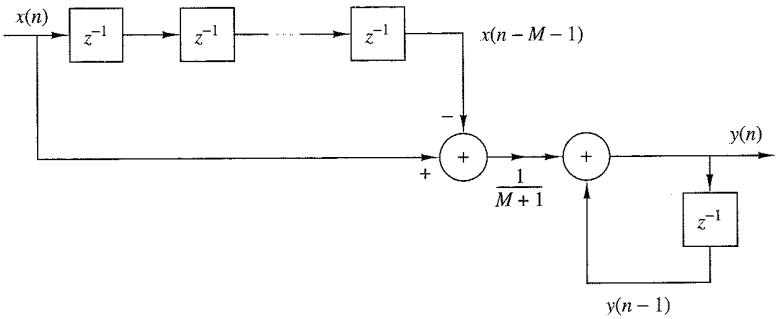



**Figure 2.5.6** Recursive realization of an FIR moving average system.

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 $\alpha$ 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.

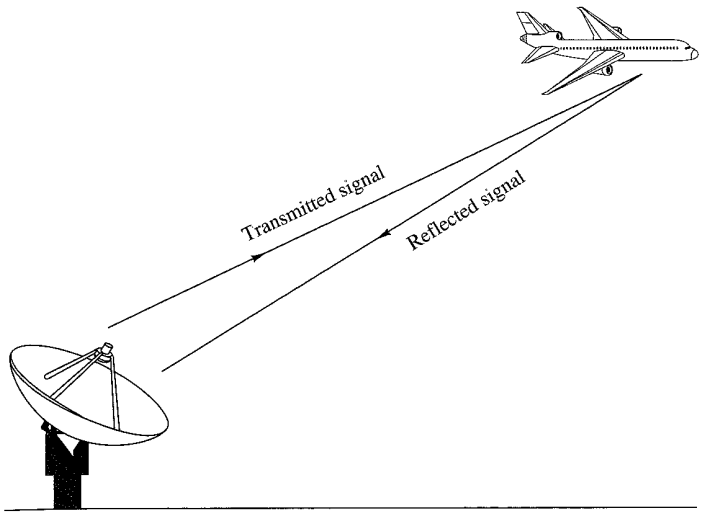



**Figure 2.6.1** Radar target detection.

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 $0 \le n \le L - 1$, and to transmit a 1 we can transmit the signal sequence $x_1(n)$ for $0 \le n \le L-1$, 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

$$y(n) = x_i(n) + w(n), \qquad i = 0, 1, \qquad 0 \le n \le L - 1 \tag{2.6.2}$$

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

$$\begin{aligned} &r_{xy}(1) = 13, \quad r_{xy}(2) = -18, \quad r_{xy}(3) = 16, \quad r_{xy}(4) = -7 \\ &r_{xy}(5) = 5, \quad r_{xy}(6) = -3, \quad r_{xy}(l) = 0, \quad l \geq 7 \end{aligned}$$

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

$$\begin{aligned} &r_{xy}(-1) = 0, \quad r_{xy}(-2) = 33, \quad r_{xy}(-3) = -14, \quad r_{xy}(-4) = 36 \\ &r_{xy}(-5) = 19, \quad r_{xy}(-6) = -9, \quad r_{xy}(-7) = 10, \quad r_{xy}(l) = 0, \ l \leq -8 \end{aligned}$$

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

$$\begin{aligned}\sum_{n=-\infty}^{\infty} [ax(n) + by(n-l)]^2 &= a^2 \sum_{n=-\infty}^{\infty} x^2(n) + b^2 \sum_{n=-\infty}^{\infty} y^2(n-l) \\ &\quad + 2ab \sum_{n=-\infty}^{\infty} x(n)y(n-l) \\ &= a^2 r_{xx}(0) + b^2 r_{yy}(0) + 2ab r_{xy}(l)\end{aligned} \tag{2.6.13}$$

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

$$a^2 r_{xx}(0) + b^2 r_{yy}(0) + 2ab r_{xy}(l) \geq 0 \tag{2.6.14}$$

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

$$r_{xx}(l) = \frac{1}{1-a^2} a^{|l|}, \qquad l \geq 0$$

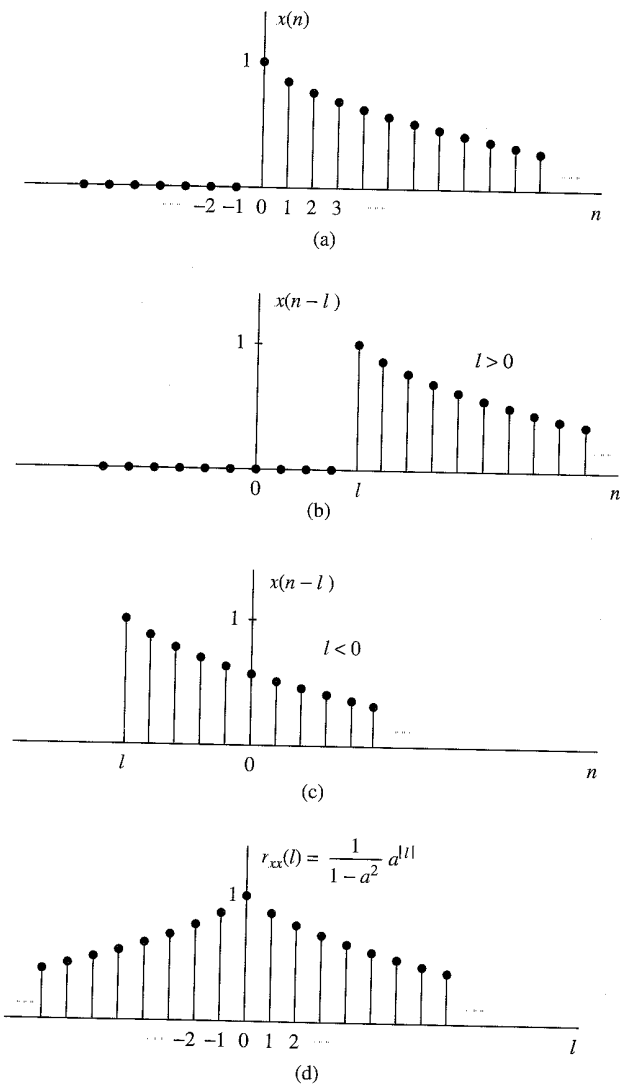



**Figure 2.6.2** Computation of the autocorrelation of the signal $x(n) = a^n$, $0 < a < 1$.

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 $\Delta$ 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 $\Delta$. 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 $r_{xx}(l)$

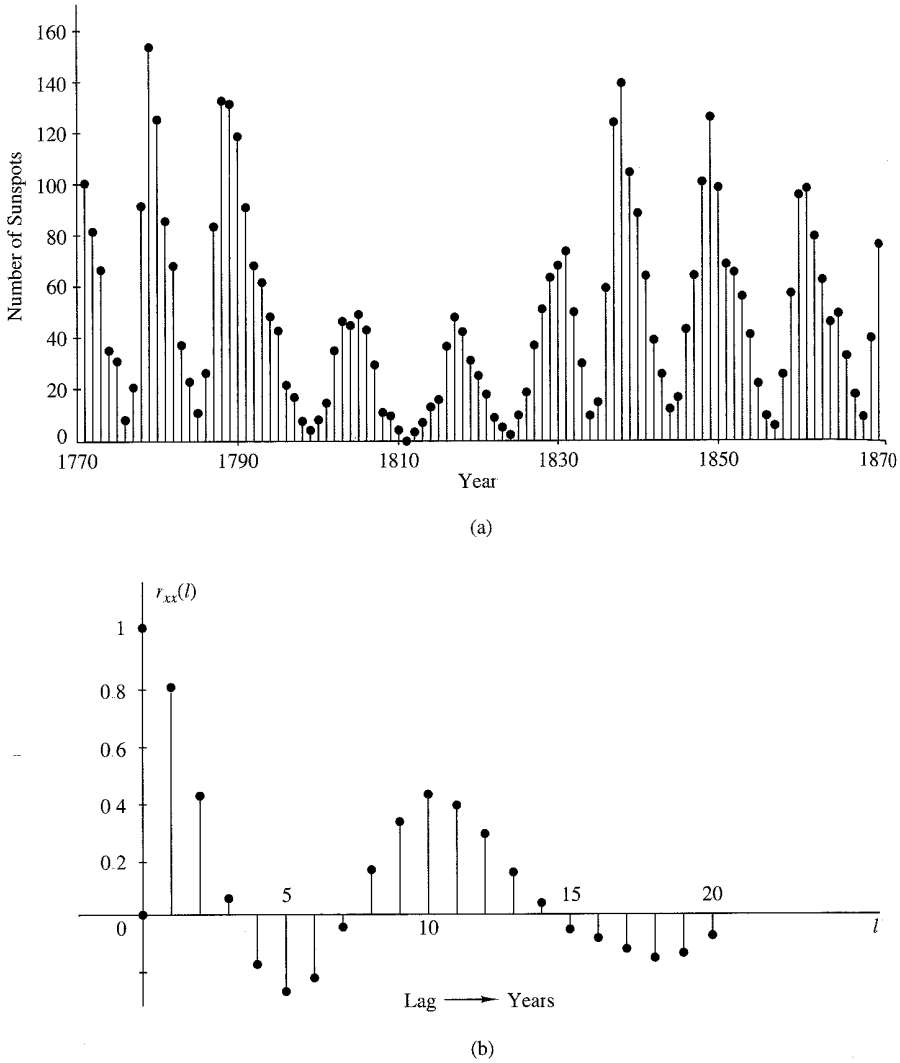



**Figure 2.6.3** Identification of periodicity in the Wölfer sunspot numbers: (a) annual Wölfer sunspot numbers; (b) normalized autocorrelation sequence.

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

$$r_{yx}(l) = y(l) * x(-l) = h(l) * [x(l) * x(-l)]$$

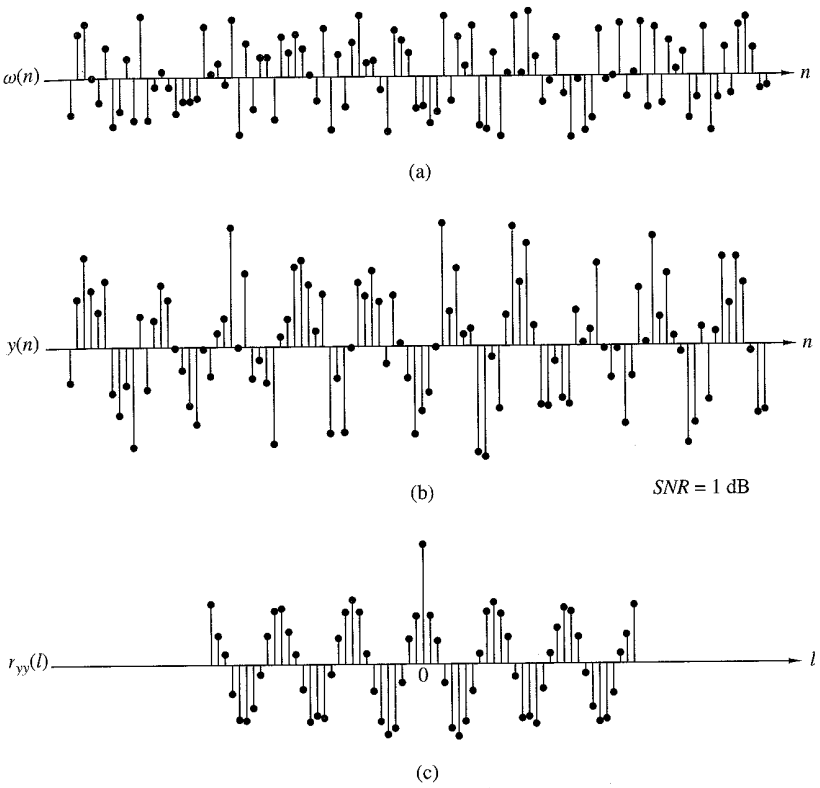



**Figure 2.6.4** Use of autocorrelation to detect the presence of a periodic signal corrupted by noise.

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.

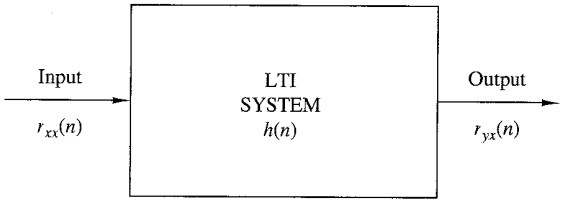



**Figure 2.6.5**
Input–output relation for crosscorrelation $r_{yx}(n)$.

## 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.

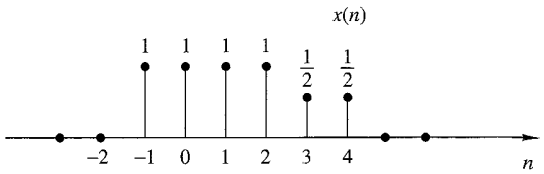



Figure P2.2

**(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.

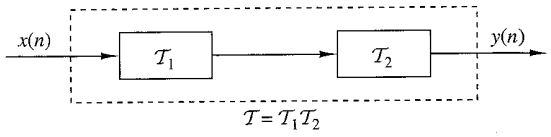



**Figure P2.8**

**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:

$$x_1(n) = \{\underset{\uparrow}{1}, 0, 2\} \overset{\mathcal{T}}{\longleftrightarrow} y_1(n) = \{\underset{\uparrow}{0}, 1, 2\}$$

$$x_2(n) = \{\underset{\uparrow}{0}, 0, 3\} \overset{\mathcal{T}}{\longleftrightarrow} y_2(n) = \{\underset{\uparrow}{0}, 1, 0, 2\}$$

$$x_3(n) = \{\underset{\uparrow}{0}, 0, 0, 1\} \overset{\mathcal{T}}{\longleftrightarrow} y_3(n) = \{1, \underset{\uparrow}{2}, 1\}$$

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:

$$x_1(n) = \{-1, \underset{\uparrow}{2}, 1\} \overset{\mathcal{T}}{\longleftrightarrow} y_1(n) = \{1, \underset{\uparrow}{2}, -1, 0, 1\}$$

$$x_2(n) = \{1, \underset{\uparrow}{-1}, -1\} \overset{\mathcal{T}}{\longleftrightarrow} y_2(n) = \{-1, \underset{\uparrow}{1}, 0, 2\}$$

$$x_3(n) = \{0, \underset{\uparrow}{1}, 1\} \overset{\mathcal{T}}{\longleftrightarrow} y_3(n) = \{\underset{\uparrow}{1}, 2, 1\}$$

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 $M_n$.

**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

$$\sum^{N} n = Ma^n = \begin{cases} \dfrac{a^M - a^{N+1}}{1-a}, & \text{if } a \neq 1 \\ N - M + 1, & \text{if } a = 1 \end{cases}$$

**(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.

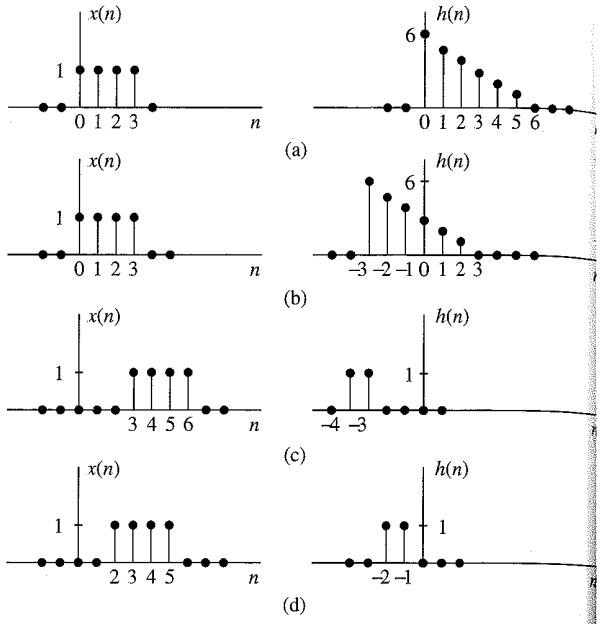



**Figure P2.17**

**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)** $x(n) = \begin{cases} 1, & n = -2, 0, 1 \\ 2, & n = -1 \\ 0, & \text{elsewhere} \end{cases}$ $\quad h(n) = \delta(n) - \delta(n-1) + \delta(n-4) + \delta(n-5)$

**(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 $\gamma(n) = a^n u(n)$, $0 < a < 1$.

**(a)** Show that any sequence $x(n)$ can be decomposed as

$$x(n) = \sum_{n=-\infty}^{\infty} c_k \gamma(n-k)$$

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 $g(n) = \mathcal{T}[\gamma(n)]$, 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.

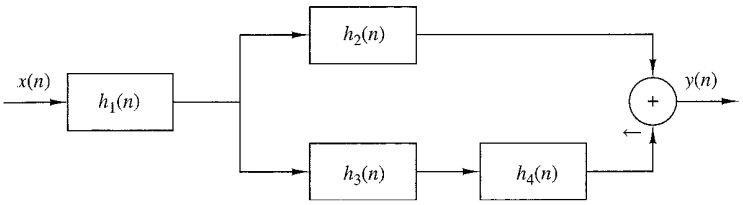



**Figure P2.35**

**(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

$$x(n) = u(n+5) - u(n-10)$$

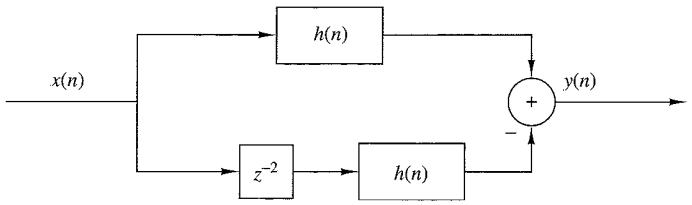



**Figure P2.36**

**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.

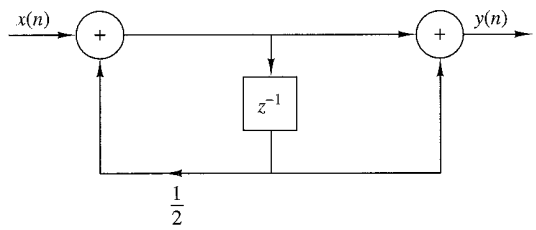



**Figure P2.47**

**(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.

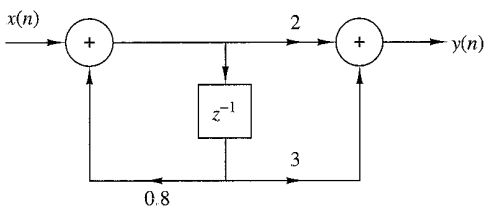



**Figure P2.49**

**2.50** Consider the discrete-time system shown in Fig. P2.50.

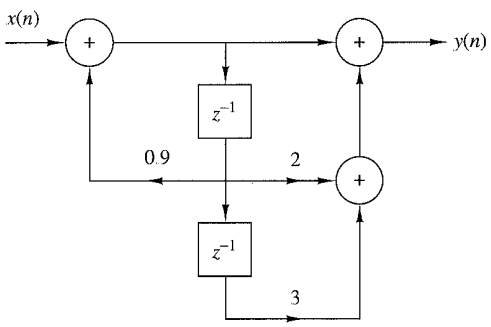



**Figure P2.50**

**(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 $n = 0, 1, \ldots, 9$.

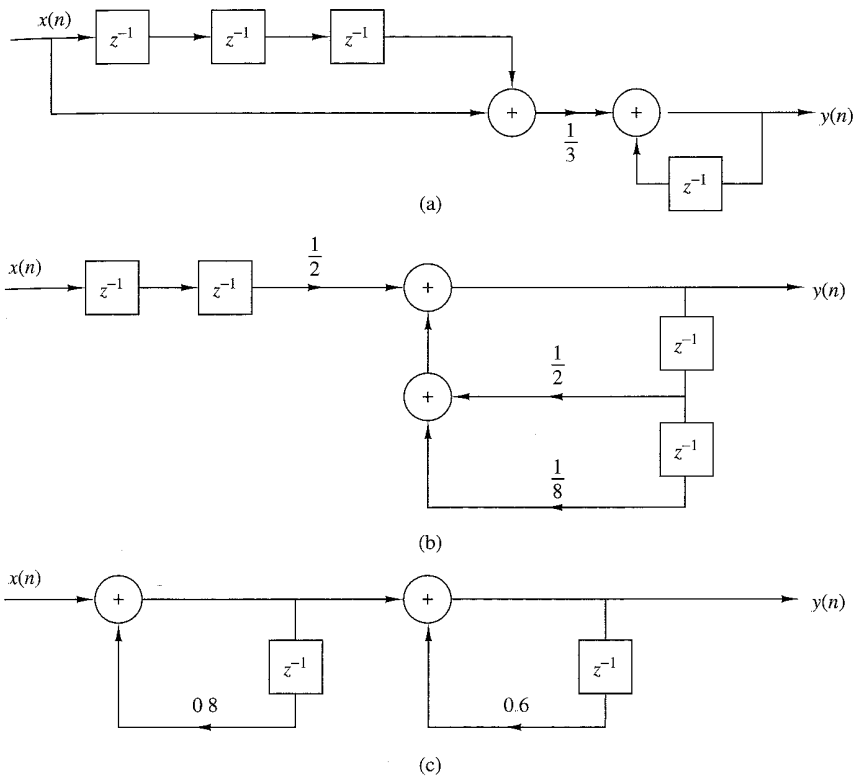



**Figure P2.51**

**(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.

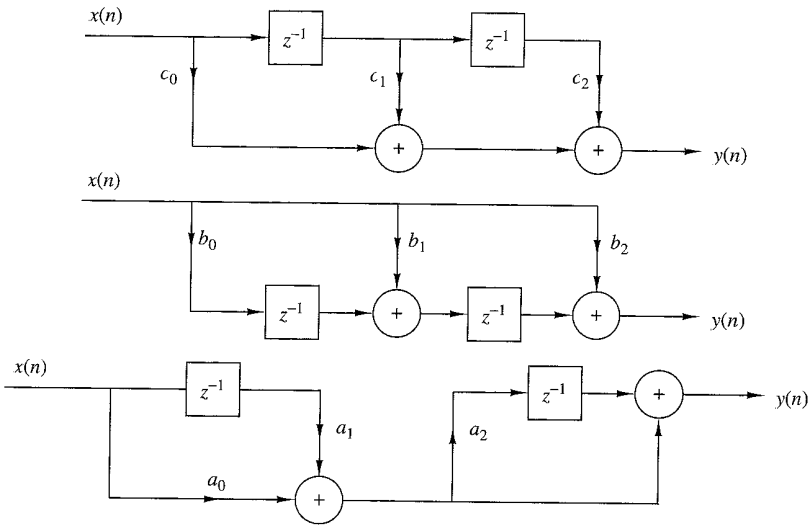



**Figure P2.52**

**(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.

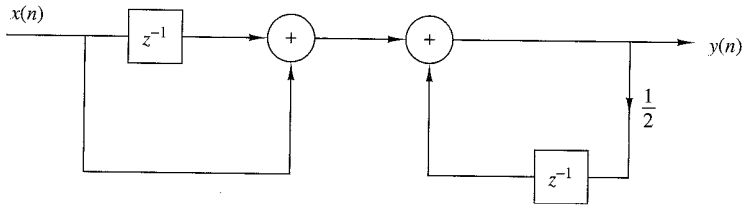



**Figure P2.53**

**(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 the 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

$$\begin{aligned} x(n) &= x_a(nT) \\ y(n) &= y_a(nT) = a x_a(nT - DT) + v_a(nT) \\ &\triangleq a x(n - D) + v(n) \end{aligned}$$

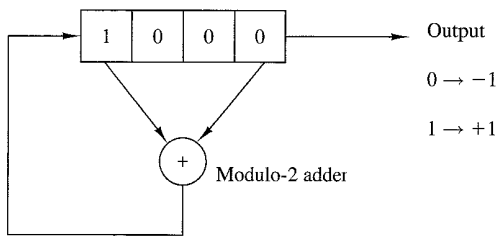



**Figure P2.65** Linear feedback shift register.

**(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

$$\mathcal{T}_1 : h_1(n) = \{\underset{\uparrow}{1}, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}\}$$

$$\mathcal{T}_2 : h_2(n) = \{\underset{\uparrow}{1}, 1, 1, 1, 1\}$$

$$\mathcal{T}_3 : y_3(n) = \frac{1}{4}x(n) + \frac{1}{2}x(n-1) + \frac{1}{4}x(n-2)$$

$$\mathcal{T}_4 : y(n) = 0.9y(n-1) - 0.81y(n-2) + v(n) + v(n-1)$$

Plot $h(n)$ for $0 \le n \le 99$.

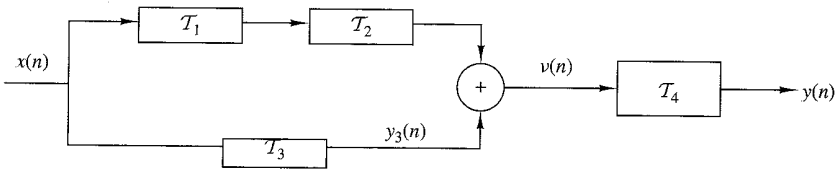



**Figure P2.67**

ElCoM
Electrical | Computer | Mechatronics
ELCOM-HU.COM

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

$$\begin{aligned} X(z) &= 1 + \frac{1}{2}z^{-1} + (\frac{1}{2})^2 z^{-2} + (\frac{1}{2})^n z^{-n} + \cdots \\ &= \sum_{n=0}^{\infty} (\frac{1}{2})^n z^{-n} = \sum_{n=0}^{\infty} (\frac{1}{2}z^{-1})^n \end{aligned}$$

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.

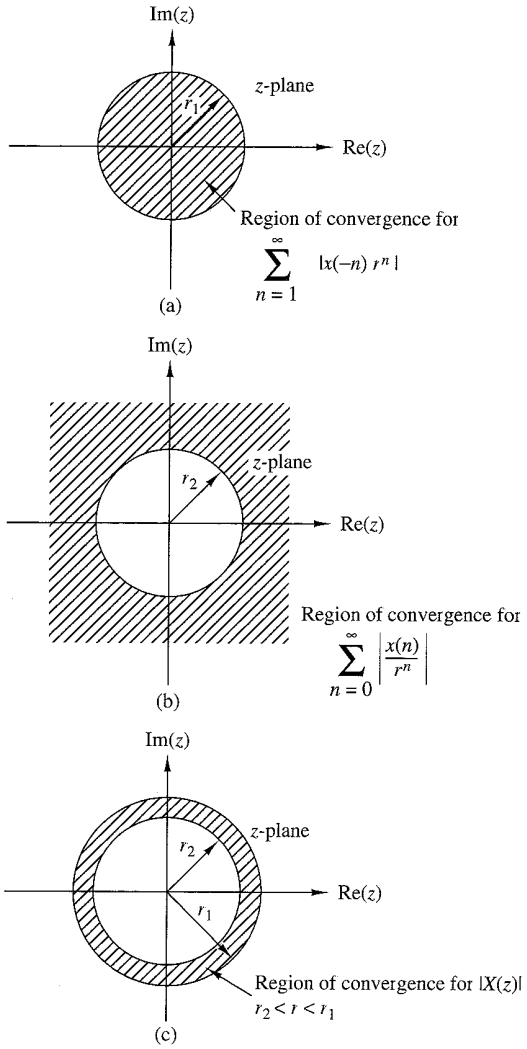



**Figure 3.1.1**
Region of convergence for $X(z)$ and its corresponding causal and anticausal components.

EXAMPLE 3.1.3

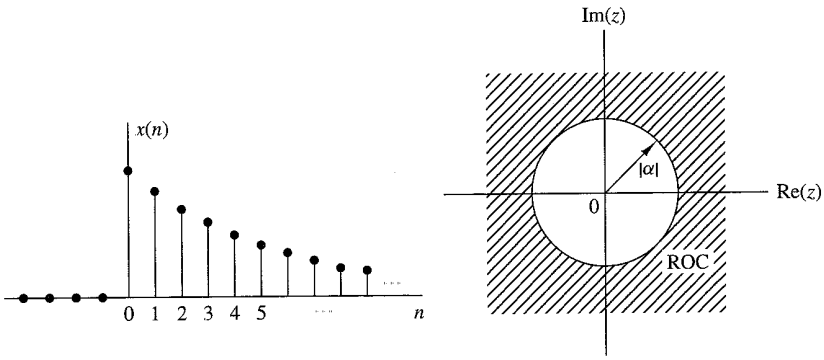
Determine the $z$-transform of the signal

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

**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




(a) (b)

**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, $\alpha$ 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.

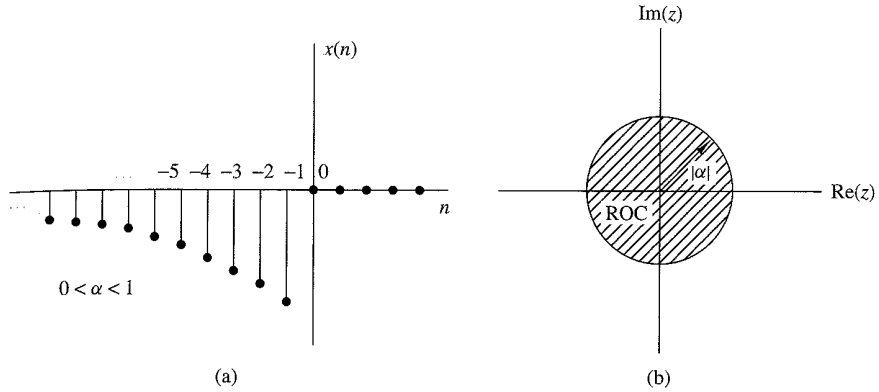



**Figure 3.1.3** Anticausal signal $x(n) = -\alpha^n u(-n-1)$ (a), and the ROC of its $z$-transform (b).

---

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.

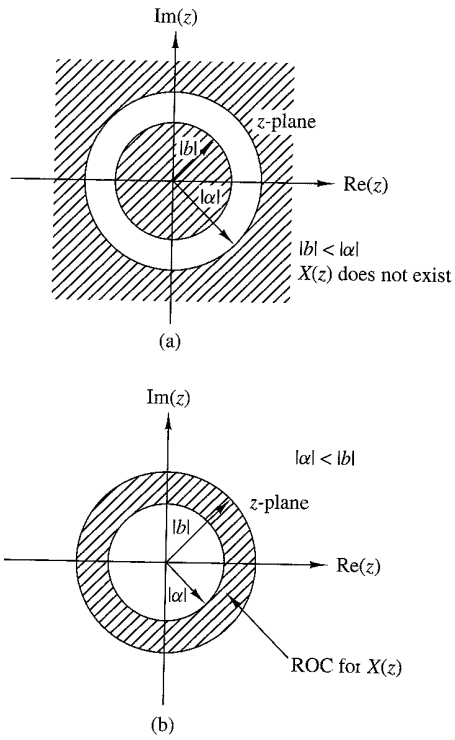



**Figure 3.1.4**
ROC for $z$-transform in Example 3.1.5.

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

$$\begin{aligned} X(z) &= \frac{1}{1-\alpha z^{-1}} - \frac{1}{1-bz^{-1}} \\ &= \frac{b-\alpha}{\alpha + b - z - \alpha b z^{-1}} \end{aligned} \tag{3.1.10}$$

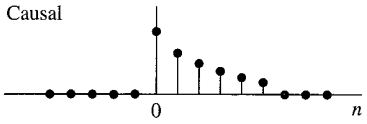
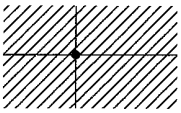
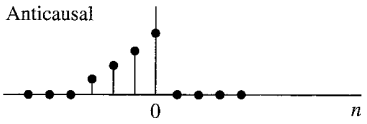
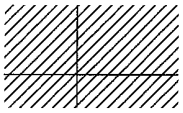
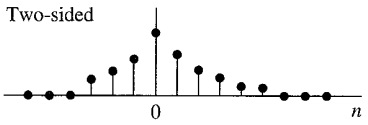
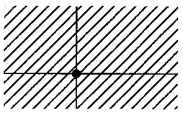
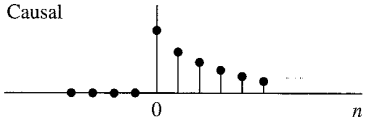
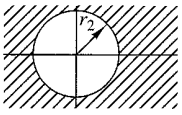
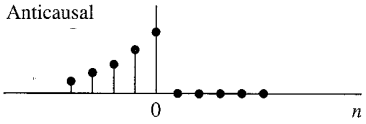
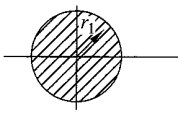
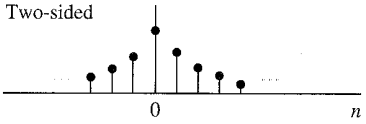
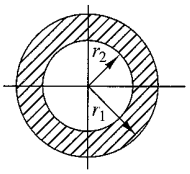
The ROC of $X(z)$ is $|\alpha| < |z| < |b|$.

---

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

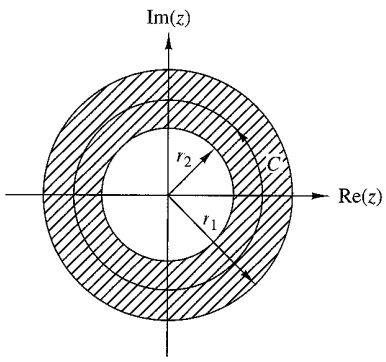



**Figure 3.1.5**
Contour $C$ for integral in (3.1.13).

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., $x(n) \to x(n-k)$] 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.

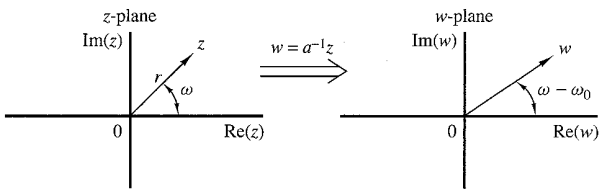



**Figure 3.2.1** Mapping of the $z$-plane to the $w$-plane via the transformation $w = a^{-1}z$, $a = r_0 e^{j\omega_0}$.

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

$$a^n(\cos\omega_0 n)u(n) \overset{z}{\longleftrightarrow} \frac{1 - az^{-1}\cos\omega_0}{1 - 2az^{-1}\cos\omega_0 + a^2z^{-2}}, \qquad |z| > |a| \tag{3.2.10}$$

**(b)** Similarly, (3.2.4) and (3.2.9) yield

$$a^n(\sin\omega_0 n)u(n) \overset{z}{\longleftrightarrow} \frac{az^{-1}\sin\omega_0}{1 - 2az^{-1}\cos\omega_0 + a^2z^{-2}}, \qquad |z| > |a| \tag{3.2.11}$$

---

**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

$$\begin{aligned}\frac{dX(z)}{dz} &= \sum_{n=-\infty}^{\infty} x(n)(-n)z^{-n-1} = -z^{-1}\sum_{n=-\infty}^{\infty}[nx(n)]z^{-n} \\ &= -z^{-1}Z\{nx(n)\}\end{aligned}$$

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.

$$X_1(z) = Z\{x_1(n)\}$$

$$\text{(time domain} \longrightarrow z\text{-domain)}$$

$$X_2(z) = Z\{x_2(n)\}$$

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: $r_2 < \|z\| < r_1$ |
| | $x_1(n)$ | $X_1(z)$ | $\text{ROC}_1$ |
| | $x_2(n)$ | $X_2(z)$ | $\text{ROC}_2$ |
| Linearity | $a_1x_1(n) + a_2x_2(n)$ | $a_1X_1(z) + a_2X_2(z)$ | At least the intersection of $\text{ROC}_1$ and $\text{ROC}_2$ |
| 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 | $nx(n)$ | $-z\dfrac{dX(z)}{dz}$ | $r_2 < \|z\| < r_1$ |
| Convolution | $x_1(n) * x_2(n)$ | $X_1(z)X_2(z)$ | At least, the intersection of $\text{ROC}_1$ and $\text{ROC}_2$ |
| 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 | $u(n)$ | $\frac{1}{1-z^{-1}}$ | $\lvert z\rvert > 1$ |
| 3 | $a^n u(n)$ | $\frac{1}{1-az^{-1}}$ | $\lvert z\rvert > \lvert a\rvert$ |
| 4 | $na^n u(n)$ | $\frac{az^{-1}}{(1-az^{-1})^2}$ | $\lvert z\rvert > \lvert a\rvert$ |
| 5 | $-a^n u(-n-1)$ | $\frac{1}{1-az^{-1}}$ | $\lvert z\rvert < \lvert a\rvert$ |
| 6 | $-na^n u(-n-1)$ | $\frac{az^{-1}}{(1-az^{-1})^2}$ | $\lvert z\rvert < \lvert a\rvert$ |
| 7 | $(\cos\omega_0 n)u(n)$ | $\frac{1-z^{-1}\cos\omega_0}{1-2z^{-1}\cos\omega_0+z^{-2}}$ | $\lvert z\rvert > 1$ |
| 8 | $(\sin\omega_0 n)u(n)$ | $\frac{z^{-1}\sin\omega_0}{1-2z^{-1}\cos\omega_0+z^{-2}}$ | $\lvert z\rvert > 1$ |
| 9 | $(a^n\cos\omega_0 n)u(n)$ | $\frac{1-az^{-1}\cos\omega_0}{1-2az^{-1}\cos\omega_0+a^2z^{-2}}$ | $\lvert z\rvert > \lvert a\rvert$ |
| 10 | $(a^n\sin\omega_0 n)u(n)$ | $\frac{az^{-1}\sin\omega_0}{1-2az^{-1}\cos\omega_0+a^2z^{-2}}$ | $\lvert z\rvert > \lvert a\rvert$ |

## 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.

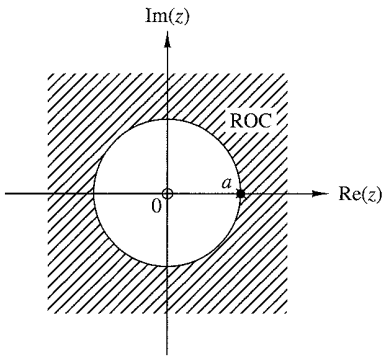



**Figure 3.3.1**
Pole–zero plot for the causal exponential signal $x(n) = a^n u(n)$.

### 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.

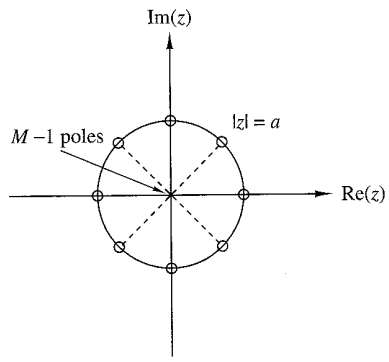



**Figure 3.3.2** Pole–zero pattern for the finite-duration signal $x(n) = a^n$, $0 \le n \le M-1 (a > 0)$, for $M = 8$.

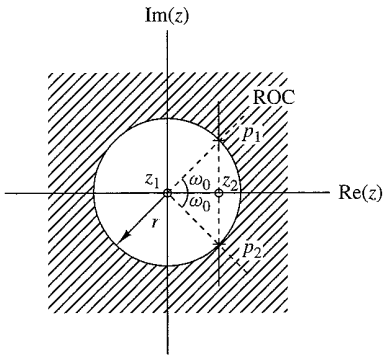



**Figure 3.3.3**
Pole-zero pattern for Example 3.3.3.

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.

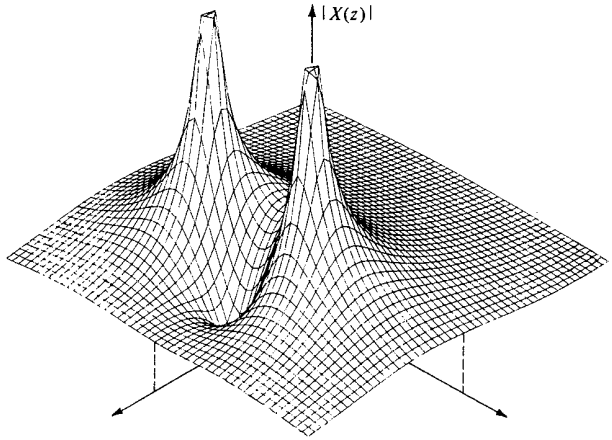



**Figure 3.3.4** Graph of $|X(z)|$ for the $z$-transform in (3.3.3).

### 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.

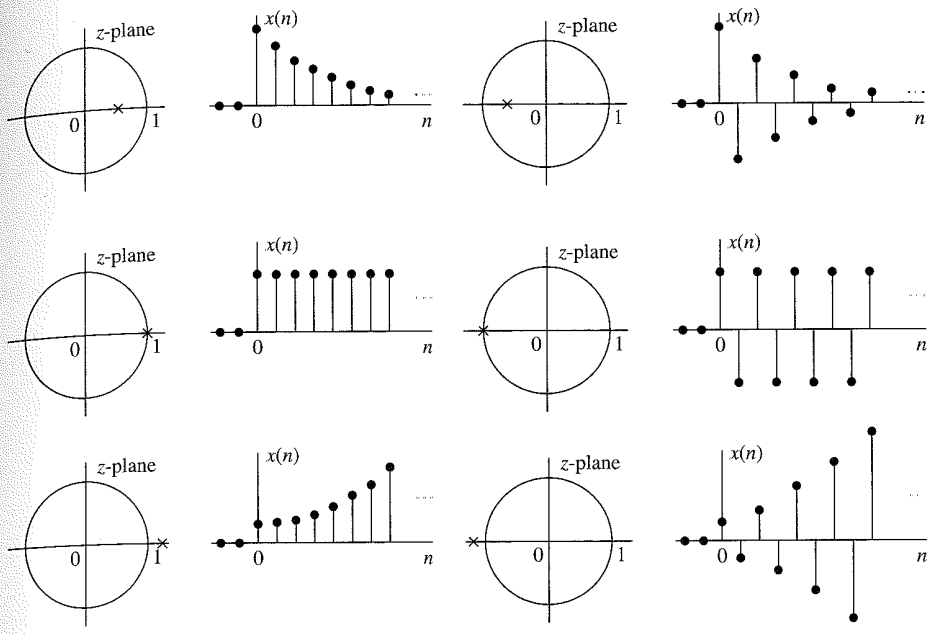



**Figure 3.3.5** Time-domain behavior of a single-real-pole causal signal as a function of the location of the pole with respect to the unit circle.

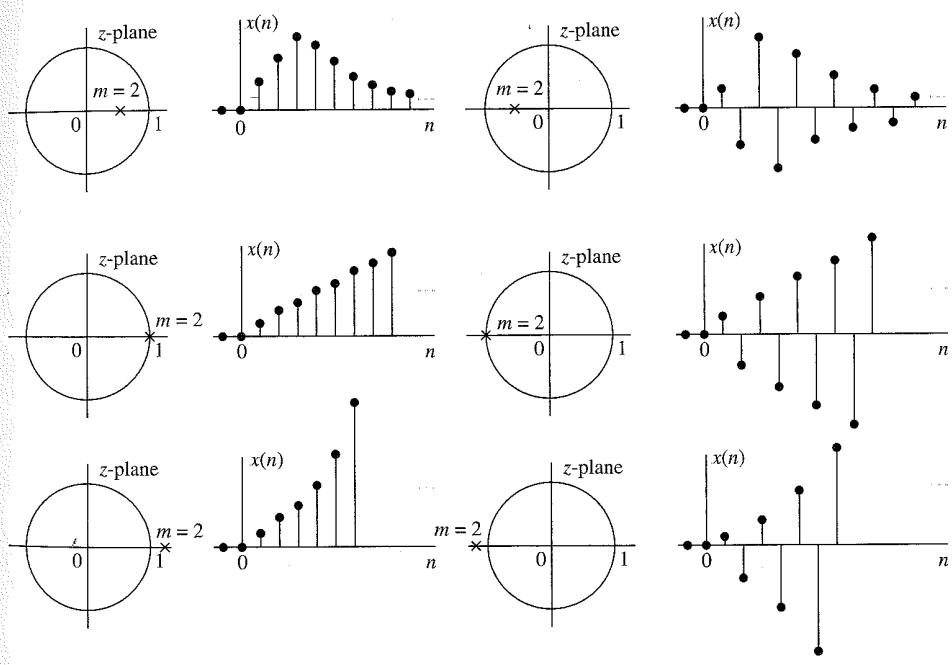



**Figure 3.3.6** Time-domain behavior of causal signals corresponding to a double ($m = 2$) real pole, as a function of the pole location.

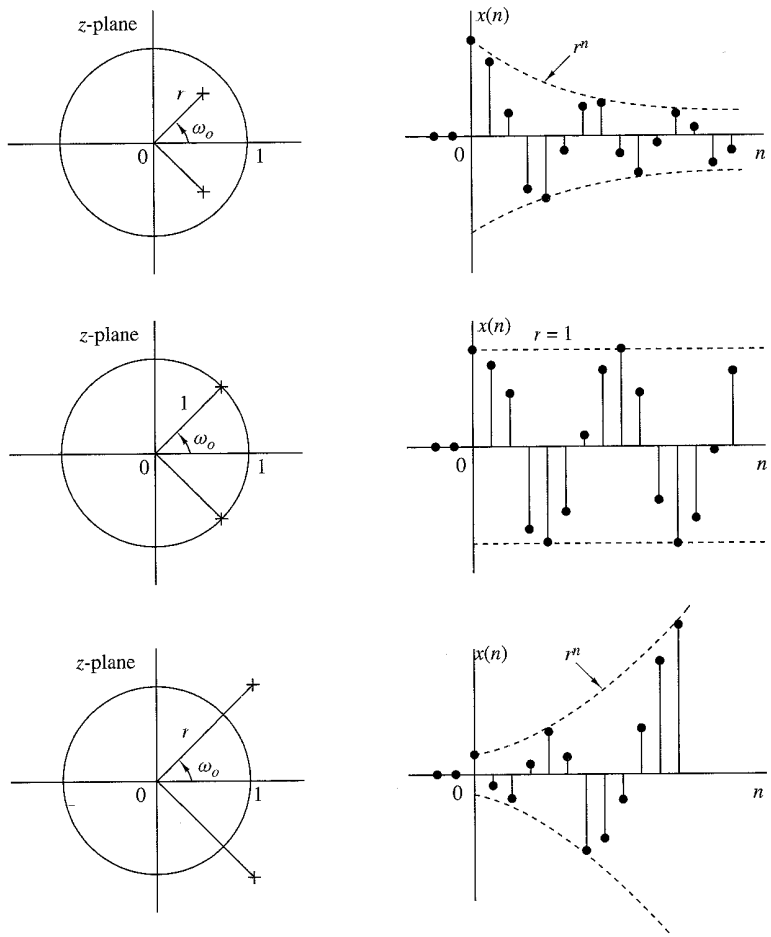



**Figure 3.3.7** A pair of complex-conjugate poles corresponds to causal signals with oscillatory behavior.

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)

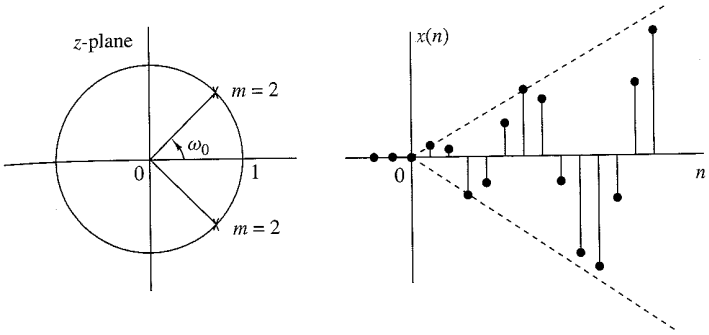



**Figure 3.3.8** Causal signal corresponding to a double pair of complex-conjugate poles on the unit circle.

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 $M$th-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 $N$th-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

$$\frac{1}{2\pi j} \oint_C \frac{f(z)}{(z - z_0)^k} dz = \begin{cases} \dfrac{1}{(k-1)!} \left. \dfrac{d^{k-1} f(z)}{dz^{k-1}} \right|_{z=z_0}, & \text{if } z_0 \text{ is inside } C \\ 0, & \text{if } z_0 \text{ is outside } C \end{cases} \tag{3.4.3}$$

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 $n$th-order pole at $z = 0$, which is also inside $C$. Thus there are contributions from both poles. For $n = -1$ we have

$$x(-1) = \frac{1}{2\pi j} \oint_C \frac{1}{z(z-a)}\, dz = \left.\frac{1}{z-a}\right|_{z=0} + \left.\frac{1}{z}\right|_{z=a} = 0$$

If $n = -2$, we have

$$x(-2) = \frac{1}{2\pi j} \oint_C \frac{1}{z^2(z-a)}\, dz = \left.\frac{d}{dz}\left(\frac{1}{z-a}\right)\right|_{z=0} + \left.\frac{1}{z^2}\right|_{z=a} = 0$$

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

$$x(n) = \{\underset{\uparrow}{1}, \frac{3}{2}, \frac{7}{4}, \frac{15}{8}, \frac{31}{16}, \cdots\}$$

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:

$$\begin{array}{r|l}
 & 2z^2 + 6z^3 + 14z^4 + 30z^5 + 62z^6 + \cdots \\
\hline
\frac{1}{2}z^{-2} - \frac{3}{2}z^{-1} + 1 & 1 \\
 & \underline{1 - 3z + 2z^2} \\
 & \quad 3z - 2z^2 \\
 & \quad \underline{3z - 9z^2 + 6z^3} \\
 & \qquad 7z^2 - 6z^3 \\
 & \qquad \underline{7z^2 - 21z^3 + 14z^4} \\
 & \qquad\quad 15z^3 - 14z^4 \\
 & \qquad\quad \underline{15z^3 - 45z^4 + 30z^5} \\
 & \qquad\qquad 31z^4 - 30z^5
\end{array}$$

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

$$x(n) = \{\cdots 62, 30, 14, 6, 2, 0, \underset{\uparrow}{0}\}$$

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

$$x(n) = \begin{cases} (-1)^{n+1} \frac{a^n}{n}, & n \geq 1 \\ 0, & n \leq 0 \end{cases}$$

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 $k$th 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 $m$th-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 $\alpha_k$ and $\beta_k$ 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

$$\begin{aligned} b_{1k} &= -2\,\mathrm{Re}(z_k), \qquad & a_{1k} &= -2\,\mathrm{Re}(p_k) \\ b_{2k} &= |z_k|^2, \qquad & a_{2k} &= |p_k|^2 \end{aligned} \tag{3.4.47}$$

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

$$y_{\text{nr}}(n) = 6.3(0.5)^n u(n)$$

and the forced or steady-state response is

$$\begin{aligned} y_{\text{fr}}(n) &= [6.78e^{-j28.7}(e^{j\pi n/4}) + 6.78e^{j28.7}e^{-j\pi n/4}]u(n) \\ &= 13.56\cos\left(\frac{\pi}{4}n - 28.7^\circ\right)u(n) \end{aligned}$$

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

$$\begin{aligned} H(z) &= \frac{3 - 4z^{-1}}{1 - 3.5z^{-1} + 1.5z^{-2}} \\ &= \frac{1}{1 - \frac{1}{2}z^{-1}} + \frac{2}{1 - 3z^{-1}} \end{aligned}$$

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.

**Solution.** The system has poles at $z = \frac{1}{2}$ and $z = 3$.

**(a)** Since the system is stable, its ROC must include the unit circle and hence it is $\frac{1}{2} < |z| < 3$. Consequently, $h(n)$ is noncausal and is given as

$$h(n) = (\frac{1}{2})^n u(n) - 2(3)^n u(-n-1)$$

**(b)** Since the system is causal, its ROC is $|z| > 3$. In this case

$$h(n) = (\frac{1}{2})^n u(n) + 2(3)^n u(n)$$

This system is unstable.

**(c)** If the system is anticausal, its ROC is $|z| < 0.5$. Hence

$$h(n) = -[(\frac{1}{2})^n + 2(3)^n]u(-n-1)$$

In this case the system is unstable.

### 3.5.4 Pole–Zero Cancellations

When a $z$-transform has a pole that is at the same location as a zero, the pole is canceled by the zero and, consequently, the term containing that pole in the inverse $z$-transform vanishes. Such pole–zero cancellations are very important in the analysis of pole–zero systems.

Pole–zero cancellations can occur either in the system function itself or in the product of the system function with the $z$-transform of the input signal. In the first case we say that the order of the system is reduced by one. In the latter case we say that the pole of the system is suppressed by the zero in the input signal, or vice versa. Thus, by properly selecting the position of the zeros of the input signal, it is possible to suppress one or more system modes (pole factors) in the response of the system. Similarly, by proper selection of the zeros of the system function, it is possible to suppress one or more modes of the input signal from the response of the system.

When the zero is located very near the pole but not exactly at the same location, the term in the response has a very small amplitude. For example, nonexact pole–zero cancellations can occur in practice as a result of insufficiant numerical precision used in representing the coefficients of the system. Consequently, one should not attempt to stabilize an inherently unstable system by placing a zero in the input signal at the location of the pole.

#### EXAMPLE 3.5.3

Determine the unit sample response of the system characterized by the difference equation

$$y(n) = 2.5y(n-1) - y(n-2) + x(n) - 5x(n-1) + 6x(n-2)$$

**Solution.** The system function is

$$H(z) = \frac{1 - 5z^{-1} + 6z^{-2}}{1 - 2.5z^{-1} + z^{-2}}$$
$$= \frac{1 - 5z^{-1} + 6z^{-2}}{(1 - \frac{1}{2}z^{-1})(1 - 2z^{-1})}$$

This system has poles at $p_1 = 2$ and $p_1 = \frac{1}{2}$. Consequently, at first glance it appears that the unit sample response is

$$Y(z) = H(z)X(z) = \frac{1 - 5z^{-1} + 6z^{-2}}{(1 - \frac{1}{2}z^{-1})(1 - 2z^{-1})}$$
$$= z\left(\frac{A}{z - \frac{1}{2}} + \frac{B}{z - 2}\right)$$

By evaluating the constants at $z = \frac{1}{2}$ and $z = 2$, we find that

$$A = \frac{5}{2}, \qquad B = 0$$

The fact that $B = 0$ indicates that there exists a zero at $z = 2$ which cancels the pole at $z = 2$. In fact, the zeros occur at $z = 2$ and $z = 3$. Consequently, $H(z)$ reduces to

$$H(z) = \frac{1 - 3z^{-1}}{1 - \frac{1}{2}z^{-1}} = \frac{z - 3}{z - \frac{1}{2}}$$
$$= 1 - \frac{2.5z^{-1}}{1 - \frac{1}{2}z^{-1}}$$

and therefore

$$h(n) = \delta(n) - 2.5(\frac{1}{2})^{n-1}u(n-1)$$

The reduced-order system obtained by canceling the common pole and zero is characterized by the difference equation

$$y(n) = \frac{1}{2}y(n-1) + x(n) - 3x(n-1)$$

Although the original system is also BIBO stable due to the pole–zero cancellation, in a practical implementation of this second-order system, we may encounter an instability due to imperfect cancellation of the pole and the zero.

---

**EXAMPLE 3.5.4**

Determine the response of the system

$$y(n) = \frac{5}{6}y(n-1) - \frac{1}{6}y(n-2) + x(n)$$

to the input signal $x(n) = \delta(n) - \frac{1}{3}\delta(n-1)$.

**Solution.** The system function is

$$H(z) = \frac{1}{1 - \frac{5}{6}z^{-1} + \frac{1}{6}z^{-2}}$$

$$= \frac{1}{\left(1 - \frac{1}{2}z^{-1}\right)\left(1 - \frac{1}{3}z^{-1}\right)}$$

This system has two poles, one at $z = \frac{1}{2}$ and the other at $z = \frac{1}{3}$. The $z$-transform of the input signal is

$$X(z) = 1 - \frac{1}{3}z^{-1}$$

In this case the input signal contains a zero at $z = \frac{1}{3}$ which cancels the pole at $z = \frac{1}{3}$. Consequently,

$$Y(z) = H(z)X(z)$$

$$Y(z) = \frac{1}{1 - \frac{1}{2}z^{-1}}$$

and hence the response of the system is

$$y(n) = (\frac{1}{2})^n u(n)$$

Clearly, the mode $(\frac{1}{3})^n$ is suppressed from the output as a result of the pole–zero cancellation.

### 3.5.5 Multiple-Order Poles and Stability

As we have observed, a necessary and sufficient condition for a causal linear time-invariant system to be BIBO stable is that all its poles lie inside the unit circle. The input signal is bounded if its $z$-transform contains poles $\{q_k\}$, $k = 1, 2, \ldots, L$, which satisfy the condition $|q_k| \leq 1$ for all $k$. We note that the forced response of the system, given in (3.5.6), is also bounded, even when the input signal contains one or more distinct poles on the unit circle.

In view of the fact that a bounded input signal may have poles on the unit circle, it might appear that a stable system may also have poles on the unit circle. This is not the case, however, since such a system produces an unbounded response when excited by an input signal that also has a pole at the same position on the unit circle. The following example illustrates this point.

#### EXAMPLE 3.5.5

Determine the step response of the causal system described by the difference equation

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

**Solution.** The system function for the system is

$$H(z) = \frac{1}{1 - z^{-1}}$$

We note that the system contains a pole on the unit circle at $z = 1$. The $z$-transform of the input signal $x(n) = u(n)$ is

$$X(z) = \frac{1}{1 - z^{-1}}$$

which also contains a pole at $z = 1$. Hence the output signal has the transform

$$\begin{aligned} Y(z) &= H(z)X(z) \\ &= \frac{1}{(1 - z^{-1})^2} \end{aligned}$$

which contains a double pole at $z = 1$.

The inverse $z$-transform of $Y(z)$ is

$$y(n) = (n + 1)u(n)$$

which is a ramp sequence. Thus $y(n)$ is unbounded, even when the input is bounded. Consequently, the system is unstable.

---

Example 3.5.5 demonstrates clearly that BIBO stability requires that the system poles be strictly inside the unit circle. If the system poles are all inside the unit circle and the excitation sequence $x(n)$ contains one or more poles that coincide with the poles of the system, the output $Y(z)$ will contain multiple-order poles. As indicated previously, such multiple-order poles result in an output sequence that contains terms of the form

$$A_k n^b (p_k)^n u(n)$$

where $0 \le b \le m - 1$ and $m$ is the order of the pole. If $|p_k| < 1$, these terms decay to zero as $n$ approaches infinity because the exponential factor $(p_k)^n$ dominates the term $n^b$. Consequently, no bounded input signal can produce an unbounded output signal if the system poles are all inside the unit circle.

Finally, we should state that the only useful systems which contain poles on the unit circle are the digital oscillators discussed in Chapter 5. We call such systems *marginally stable*.

### 3.5.6 Stability of Second-Order Systems

In this section we provide a detailed analysis of a system having two poles. As we shall see in Chapter 9, two-pole systems form the basic building blocks for the realization of higher-order systems.

Let us consider a causal two-pole system described by the second-order difference equation

$$y(n) = -a_1 y(n - 1) - a_2 y(n - 2) + b_0 x(n) \tag{3.5.7}$$

The system function is

$$\begin{aligned} H(z) = \frac{Y(z)}{X(z)} &= \frac{b_0}{1 + a_1 z^{-1} + a_2 z^{-1}} \\ &= \frac{b_0 z^2}{z^2 + a_1 z + a_2} \end{aligned} \tag{3.5.8}$$

This system has two zeros at the origin and poles at

$$p_1, p_2 = -\frac{a_1}{2} \pm \sqrt{\frac{a_1^2 - 4a_2}{4}} \tag{3.5.9}$$

The system is BIBO stable if the poles lie inside the unit circle, that is, if $|p_1| < 1$ and $|p_2| < 1$. These conditions can be related to the values of the coefficients $a_1$ and $a_2$. In particular, the roots of a quadratic equation satisfy the relations

$$a_1 = -(p_1 + p_2) \tag{3.5.10}$$

$$a_2 = p_1 p_2 \tag{3.5.11}$$

From (3.5.10) and (3.5.11) we easily obtain the conditions that $a_1$ and $a_2$ must satisfy for stability. First, $a_2$ must satisfy the condition

$$|a_2| = |p_1 p_2| = |p_1||p_2| < 1 \tag{3.5.12}$$

The condition for $a_1$ can be expressed as

$$|a_1| < 1 + a_2 \tag{3.5.13}$$

Therefore, a two-pole system is stable if and only if the coefficients $a_1$ and $a_2$ satisfy the conditions in (3.5.12) and (3.5.13).

The stability conditions given in (3.5.12) and (3.5.13) define a region in the coefficient plane $(a_1, a_2)$, which is in the form of a triangle, as shown in Fig. 3.5.1. The system is stable if and only if the point $(a_1, a_2)$ lies inside the triangle, which we call the *stability triangle*.

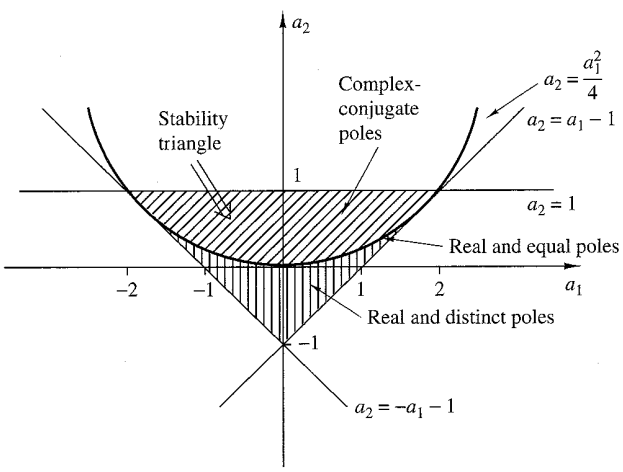



**Figure 3.5.1** Region of stability (stability triangle) in the $(a_1, a_2)$ coefficient plane for a second-order system.

The characteristics of the two-pole system depend on the location of the poles or, equivalently, on the location of the point $(a_1, a_2)$ in the stability triangle. The poles of the system may be real or complex conjugate, depending on the value of the discriminant $\Delta = a_1^2 - 4a_2$. The parabola $a_2 = a_1^2/4$ splits the stability triangle into two regions, as illustrated in Fig. 3.5.1. The region below the parabola $(a_1^2 > 4a_2)$ corresponds to real and distinct poles. The points on the parabola $(a_1^2 = 4a_2)$ result in real and equal (double) poles. Finally, the points above the parabola correspond to complex-conjugate poles.

Additional insight into the behavior of the system can be obtained from the unit sample responses for these three cases.

**Real and distinct poles ($a_1^2 > 4a_2$).** Since $p_1$, $p_2$ are real and $p_1 \neq p_2$, the system function can be expressed in the form

$$H(z) = \frac{A_1}{1 - p_1 z^{-1}} + \frac{A_2}{1 - p_2 z^{-1}} \tag{3.5.14}$$

where

$$A_1 = \frac{b_0 p_1}{p_1 - p_2}, \qquad A_2 = \frac{-b_0 p_2}{p_1 - p_2} \tag{3.5.15}$$

Consequently, the unit sample response is

$$h(n) = \frac{b_0}{p_1 - p_2}(p_1^{n+1} - p_2^{n+1})u(n) \tag{3.5.16}$$

Therefore, the unit sample response is the difference of two decaying exponential sequences. Figure 3.5.2 illustrates a typical graph for $h(n)$ when the poles are distinct.

**Real and equal poles ($a_1^2 = 4a_2$).** In this case $p_1 = p_2 = p = -a_1/2$. The system function is

$$H(z) = \frac{b_0}{(1 - pz^{-1})^2} \tag{3.5.17}$$

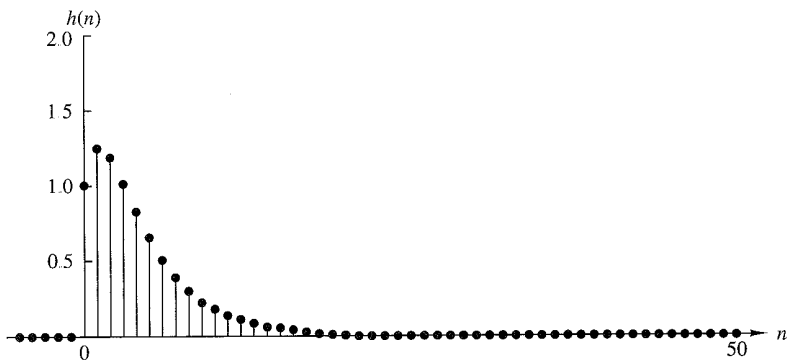



**Figure 3.5.2** Plot of $h(n)$ given by (3.5.16) with $p_1 = 0.5$, $p_2 = 0.75$; $h(n) = [1/(p_1 - p_2)](p_1^{n+1} - p_2^{n+1})u(n)$.

and hence the unit sample response of the system is

$$h(n) = b_0(n+1)p^n u(n) \tag{3.5.18}$$

We observe that $h(n)$ is the product of a ramp sequence and a real decaying exponential sequence. The graph of $h(n)$ is shown in Fig. 3.5.3.

**Complex-conjugate poles ($a_1^2 < 4a_2$).** Since the poles are complex conjugate, the system function can be factored and expressed as

$$\begin{aligned} H(z) &= \frac{A}{1-pz^{-1}} + \frac{A^*}{1-p^*z^{-1}} \\ &= \frac{A}{1-re^{j\omega_0}z^{-1}} + \frac{A^*}{1-re^{-j\omega_0}z^{-1}} \end{aligned} \tag{3.5.19}$$

where $p = re^{j\omega}$ and $0 < \omega_0 < \pi$. Note that when the poles are complex conjugates, the parameters $a_1$ and $a_2$ are related to $r$ and $\omega_0$ according to

$$\begin{aligned} a_1 &= -2r\cos\omega_0 \\ a_2 &= r^2 \end{aligned} \tag{3.5.20}$$

The constant $A$ in the partial-fraction expansion of $H(z)$ is easily shown to be

$$\begin{aligned} A &= \frac{b_0 p}{p-p^*} = \frac{b_0 r e^{j\omega_0}}{r(e^{j\omega_0} - e^{-j\omega_0})} \\ &= \frac{b_0 e^{j\omega_0}}{j2\sin\omega_0} \end{aligned} \tag{3.5.21}$$

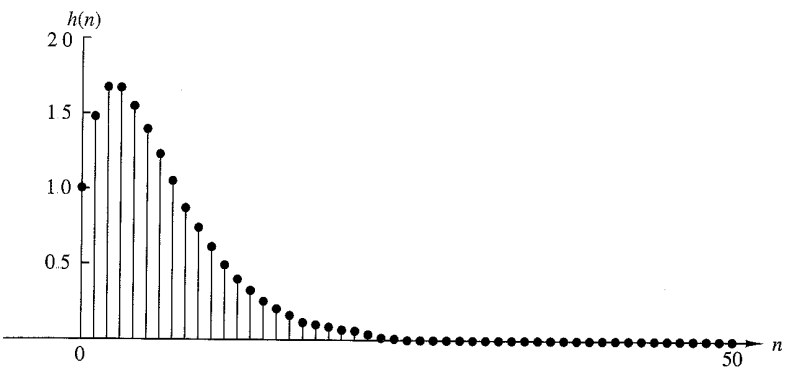



**Figure 3.5.3** Plot of $h(n)$ given by (3.5.18) with $p = \frac{3}{4}$; $h(n) = (n+1)p^n u(n)$.

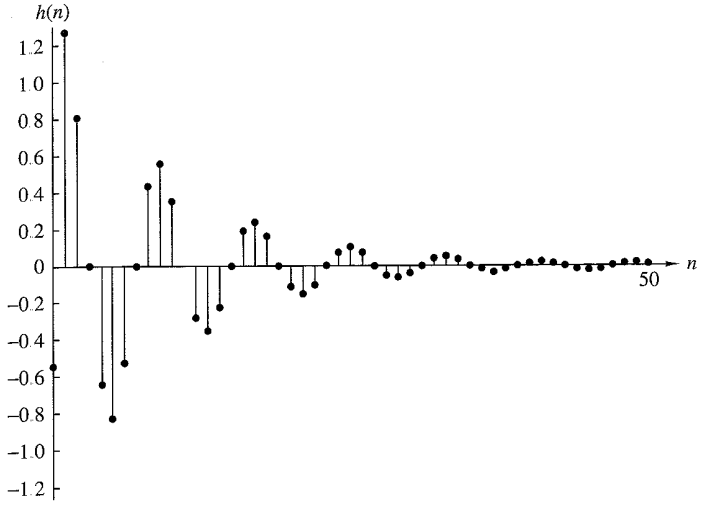



**Figure 3.5.4** Plot of $h(n)$ given by (3.5.22) with $b_0 = 1$, $\omega_0 = \pi/4$, $r = 0.9$; $h(n) = [b_0 r^n/(\sin\omega_0)]\sin[(n+1)\omega_0]u(n)$.

Consequently, the unit sample response of a system with complex-conjugate poles is

$$\begin{aligned} h(n) &= \frac{b_0 r^n}{\sin\omega_0}\,\frac{e^{j(n+1)\omega_0} - e^{-j(n+1)\omega_0}}{2j}\,u(n) \\ &= \frac{b_0 r^n}{\sin\omega_0}\sin(n+1)\omega_0 u(n) \end{aligned} \tag{3.5.22}$$

In this case $h(n)$ has an oscillatory behavior with an exponentially decaying envelope when $r < 1$. The angle $\omega_0$ of the poles determines the frequency of oscillation and the distance $r$ of the poles from the origin determines the rate of decay. When $r$ is close to unity, the decay is slow. When $r$ is close to the origin, the decay is fast. A typical graph of $h(n)$ is illustrated in Fig. 3.5.4.

## 3.6 The One-sided $z$-Transform

The two-sided $z$-transform requires that the corresponding signals be specified for the entire time range $-\infty < n < \infty$. This requirement prevents its use for a very useful family of practical problems, namely the evaluation of the output of nonrelaxed systems. As we recall, these systems are described by difference equations with nonzero initial conditions. Since the input is applied at a finite time, say $n_0$, both input and output signals are specified for $n \geq n_0$, but by no means are zero for $n < n_0$. Thus the two-sided $z$-transform cannot be used. In this section we develop the one-sided $z$-transform which can be used to solve difference equations with initial conditions.

### 3.6.1 Definition and Properties

The *one-sided* or *unilateral* $z$-transform of a signal $x(n)$ is defined by

$$X^+(z) \equiv \sum_{n=0}^{\infty} x(n)z^{-n} \tag{3.6.1}$$

We also use the notations $Z^+\{x(n)\}$ and

$$x(n) \overset{z^+}{\longleftrightarrow} X^+(z)$$

The one-sided $z$-transform differs from the two-sided transform in the lower limit of the summation, which is always zero, whether or not the signal $x(n)$ is zero for $n < 0$ (i.e., causal). Due to this choice of lower limit, the one-sided $z$-transform has the following characteristics:

1. It does not contain information about the signal $x(n)$ for negative values of time (i.e., for $n < 0$).
2. It is *unique* only for causal signals, because only these signals are zero for $n < 0$.
3. The one-sided $z$-transform $X^+(z)$ of $x(n)$ is identical to the two-sided $z$-transform of the signal $x(n)u(n)$. Since $x(n)u(n)$ is causal, the ROC of its transform, and hence the ROC of $X^+(z)$, is always the exterior of a circle. Thus when we deal with one-sided $z$-transforms, it is not necessary to refer to their ROC.

**EXAMPLE 3.6.1**

Determine the one-sided $z$-transform of the signals in Example 3.1.1.

**Solution.** From the definition (3.6.1), we obtain

$$x_1(n) = \{\underset{\uparrow}{1}, 2, 5, 7, 0, 1\} \overset{z^+}{\longleftrightarrow} X_1^+(z) = 1 + 2z^{-1} + 5z^{-2} + 7z^{-3} + z^{-5}$$

$$x_2(n) = \{1, 2, \underset{\uparrow}{5}, 7, 0, 1\} \overset{z^+}{\longleftrightarrow} X_2^+(z) = 5 + 7z^{-1} + z^{-3}$$

$$x_3(n) = \{\underset{\uparrow}{0}, 0, 1, 2, 5, 7, 0, 1\} \overset{z^+}{\longleftrightarrow} X_3^+(z) = z^{-2} + 2z^{-3} + 5z^{-4} + 7z^{-5} + z^{-7}$$

$$x_4(n) = \{2, 4, \underset{\uparrow}{5}, 7, 0, 1\} \overset{z^+}{\longleftrightarrow} X_4^+(z) = 5 + 7z^{-1} + z^{-3}$$

$$x_5(n) = \delta(n) \overset{z^+}{\longleftrightarrow} X_5^+(z) = 1$$

$$x_6(n) = \delta(n-k), \qquad k > 0 \overset{z^+}{\longleftrightarrow} X_6^+(z) = z^{-k}$$

$$x_7(n) = \delta(n+k), \qquad k > 0 \overset{z^+}{\longleftrightarrow} X_7^+(z) = 0$$

Note that for a noncausal signal, the one-sided $z$-transform is not unique. Indeed, $X_2^+(z) = X_4^+(z)$ but $x_2(n) \neq x_4(n)$. Also for anticausal signals, $X^+(z)$ is always zero.

Almost all properties we have studied for the two-sided $z$-transform carry over to the one-sided $z$-transform with the exception of the *shifting* property.

**Shfiting Property**

**Case 1: Time delay** If

$$x(n) \overset{z^+}{\longleftrightarrow} X^+(z)$$

then

$$x(n-k) \overset{z^+}{\longleftrightarrow} z^{-k}[X^+(z) + \sum_{n=1}^{k} x(-n)z^n], \qquad k > 0 \tag{3.6.2}$$

In case $x(n)$ is causal, then

$$x(n-k) \overset{z^+}{\longleftrightarrow} z^{-k}X^+(z) \tag{3.6.3}$$

*Proof* From the definition (3.6.1) we have

$$Z^+\{x(n-k)\} = z^{-k}\left[\sum_{l=-k}^{-1} x(l)z^{-l} + \sum_{l=0}^{\infty} x(l)z^{-l}\right]$$

$$= z^{-k}\left[\sum_{l=-1}^{-k} x(l)z^{-l} + X^+(z)\right]$$

By changing the index from $l$ to $n = -l$, the result in (3.6.2) is easily obtained.

EXAMPLE 3.6.2

Determine the one-sided $z$-transform of the signals

**(a)** $x(n) = a^n u(n)$

**(b)** $x_1(n) = x(n-2)$ where $x(n) = a^n$

**Solution.**

**(a)** From (3.6.1) we easily obtain

$$X^+(z) = \frac{1}{1 - az^{-1}}$$

**(b)** We will apply the shifting property for $k = 2$. Indeed, we have

$$Z^+\{x(n-2)\} = z^{-2}[X^+(z) + x(-1)z + x(-2)z^2]$$

$$= z^{-2}X^+(z) + x(-1)z^{-1} + x(-2)$$

Since $x(-1) = a^{-1}$, $x(-2) = a^{-2}$, we obtain

$$X_1^+(z) = \frac{z^{-2}}{1 - az^{-1}} + a^{-1}z^{-1} + a^{-2}$$

The meaning of the shifting property can be intuitively explained if we write (3.6.2) as follows:

$$Z^+\{x(n-k)\} = [x(-k) + x(-k+1)z^{-1} + \cdots + x(-1)z^{-k+1}] + z^{-k}X^+(z), \qquad k > 0 \tag{3.6.4}$$

To obtain $x(n-k)(k>0)$ from $x(n)$, we should shift $x(n)$ by $k$ samples to the right. Then $k$ "new" samples, $x(-k)$, $x(-k+1), \ldots, x(-1)$, enter the positive time axis with $x(-k)$ located at time zero. The first term in (3.6.4) stands for the $z$-transform of these samples. The "old" samples of $x(n-k)$ are the same as those of $x(n)$ simply shifted by $k$ samples to the right. Their $z$-transform is obviously $z^{-k}X^+(z)$, which is the second term in (3.6.4).

**Case 2: Time advance** If

$$x(n) \overset{z^+}{\longleftrightarrow} X^+(z)$$

then

$$x(n+k) \overset{z^+}{\longleftrightarrow} z^k\left[X^+(z) - \sum_{n=0}^{k-1} x(n)z^{-n}\right], \qquad k > 0 \tag{3.6.5}$$

*Proof* From (3.6.1) we have

$$Z^+\{x(n+k)\} = \sum_{n=0}^{\infty} x(n+k)z^{-n} = z^k \sum_{l=k}^{\infty} x(l)z^{-l}$$

where we have changed the index of summation from $n$ to $l = n + k$. Now, from (3.6.1) we obtain

$$X^+(z) = \sum_{l=0}^{\infty} x(l)z^{-l} = \sum_{l=0}^{k-1} x(l)z^{-l} + \sum_{l=k}^{\infty} x(l)z^{-l}$$

By combining the last two relations, we easily obtain (3.6.5).

EXAMPLE 3.6.3

With $x(n)$, as given in Example 3.6.2, determine the one-sided $z$-transform of the signal

$$x_2(n) = x(n+2)$$

**Solution.** We will apply the shifting theorem for $k = 2$. From (3.6.5), with $k = 2$, we obtain

$$Z^+\{x(n+2)\} = z^2X^+(z) - x(0)z^2 - x(1)z$$

But $x(0) = 1$, $x(1) = a$, and $X^+(z) = 1/(1 - az^{-1})$. Thus

$$Z^+\{x(n+2)\} = \frac{z^2}{1 - az^{-1}} - z^2 - az$$

The case of a time advance can be intuitively explained as follows. To obtain $x(n+k)$, $k > 0$, we should shift $x(n)$ by $k$ samples to the left. As a result, the samples $x(0), x(1), \ldots, x(k-1)$ "leave" the positive time axis. Thus we first remove their contribution to the $X^+(z)$, and then multiply what remains by $z^k$ to compensate for the shifting of the signal by $k$ samples.

The importance of the shifting property lies in its application to the solution of difference equations with constant coefficients and nonzero initial conditions. This makes the one-sided $z$-transform a very useful tool for the analysis of recursive linear time-invariant discrete-time systems.

An important theorem useful in the analysis of signals and systems is the final value theorem.

**Final Value Theorem.** If

$$x(n) \overset{z^+}{\longleftrightarrow} X^+(z)$$

then

$$\lim_{n\to\infty} x(n) = \lim_{z\to 1}(z-1)X^+(z) \tag{3.6.6}$$

The limit in (3.6.6) exists if the ROC of $(z-1)X^+(z)$ includes the unit circle.

The proof of this theorem is left as an exercise for the reader.

This theorem is useful when we are interested in the asymptotic behavior of a signal $x(n)$ and we know its $z$-transform, but not the signal itself. In such cases, especially if it is complicated to invert $X^+(z)$, we can use the final value theorem to determine the limit of $x(n)$ as $n$ goes to infinity.

EXAMPLE 3.6.4

The impulse response of a relaxed linear time-invariant system is $h(n) = \alpha^n u(n)$, $|\alpha| < 1$. Determine the value of the step response of the system as $n \to \infty$.

**Solution.** The step response of the system is

$$y(n) = h(n) * x(n)$$

where

$$x(n) = u(n)$$

Obviously, if we excite a causal system with a causal input the output will be causal. Since $h(n)$, $x(n)$, $y(n)$ are causal signals, the one-sided and two-sided $z$-transforms are identical. From the convolution property (3.2.17) we know that the $z$-transforms of $h(n)$ and $x(n)$ must be multiplied to yield the $z$-transform of the output. Thus

$$Y(z) = \frac{1}{1-\alpha z^{-1}}\frac{1}{1-z^{-1}} = \frac{z^2}{(z-1)(z-\alpha)}, \qquad \text{ROC: } |z| > |\alpha|$$

Now

$$(z-1)Y(z) = \frac{z^2}{z-\alpha}, \qquad \text{ROC: } |z| < |\alpha|$$

Since $|\alpha| < 1$, the ROC of $(z-1)Y(z)$ includes the unit circle. Consequently, we can apply (3.6.6) and obtain

$$\lim_{n\to\infty} y(n) = \lim_{z\to 1}\frac{z^2}{z-\alpha} = \frac{1}{1-\alpha}$$

### 3.6.2 Solution of Difference Equations

The one-sided $z$-transform is a very efficient tool for the solution of difference equations with nonzero initial conditions. It achieves that by reducing the difference equation relating the two time-domain signals to an equivalent algebraic equation relating their one-sided $z$-transforms. This equation can be easily solved to obtain the transform of the desired signal. The signal in the time domain is obtained by inverting the resulting $z$-transform. We will illustrate this approach with two examples.

EXAMPLE 3.6.5

The well-known Fibonacci sequence of integer numbers is obtained by computing each term as the sum of the two previous ones. The first few terms of the sequence are

$$1, 1, 2, 3, 5, 8, \ldots$$

Determine a closed-form expression for the $n$th term of the Fibonacci sequence.

**Solution.** Let $y(n)$ be the $n$th term of the Fibonacci sequence. Clearly, $y(n)$ satisfies the difference equation

$$y(n) = y(n-1) + y(n-2) \tag{3.6.7}$$

with initial conditions

$$y(0) = y(-1) + y(-2) = 1 \tag{3.6.8a}$$

$$y(1) = y(0) + y(-1) = 1 \tag{3.6.8b}$$

From (3.6.8b) we have $y(-1) = 0$. Then (3.6.8a) gives $y(-2) = 1$. Thus we have to determine $y(n)$, $n \geq 0$, which satisfies (3.6.7), with initial conditions $y(-1) = 0$ and $y(-2) = 1$.

By taking the one-sided $z$-transform of (3.6.7) and using the shifting property (3.6.2), we obtain

$$Y^+(z) = [z^{-1}Y^+(z) + y(-1)] + [z^{-2}Y^+(z) + y(-2) + y(-1)z^{-1}]$$

or

$$Y^+(z) = \frac{1}{1 - z^{-1} - z^2} = \frac{z^2}{z^2 - z - 1} \tag{3.6.9}$$

where we have used the fact that $y(-1) = 0$ and $y(-2) = 1$.

We can invert $Y^+(z)$ by the partial-fraction expansion method. The poles of $Y^+(z)$ are

$$p_1 = \frac{1+\sqrt{5}}{2}, \qquad p_2 = \frac{1-\sqrt{5}}{2}$$

and the corresponding coefficients are $A_1 = p_1/\sqrt{5}$ and $A_2 = -p_2/\sqrt{5}$. Therefore,

$$y(n) = \left[\frac{1+\sqrt{5}}{2\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1-\sqrt{5}}{2\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n\right]u(n)$$

or, equivalently,

$$y(n) = \frac{1}{\sqrt{5}}\left(\frac{1}{2}\right)^{n+1}\left[\left(1+\sqrt{5}\right)^{n+1} - \left(1-\sqrt{5}\right)^{n+1}\right]u(n) \tag{3.6.10}$$

EXAMPLE 3.6.6

Determine the step response of the system

$$y(n) = \alpha y(n-1) + x(n), \qquad -1 < \alpha < 1 \tag{3.6.11}$$

when the initial condition is $y(-1) = 1$.

**Solution.** By taking the one-sided $z$-transform of both sides of (3.6.11), we obtain

$$Y^+(z) = \alpha[z^{-1}Y^+(z) + y(-1)] + X^+(z)$$

Upon substitution for $y(-1)$ and $X^+(z)$ and solving for $Y^+(z)$, we obtain the result

$$Y^+(z) = \frac{\alpha}{1-\alpha z^{-1}} + \frac{1}{(1-\alpha z^{-1})(1-z^{-1})} \tag{3.6.12}$$

By performing a partial-fraction expansion and inverse transforming the result, we have

$$\begin{aligned} y(n) &= \alpha^{n+1}u(n) + \frac{1-\alpha^{n+1}}{1-\alpha}u(n) \\ &= \frac{1}{1-\alpha}(1-\alpha^{n+2})u(n) \end{aligned} \tag{3.6.13}$$

---

### 3.6.3 Response of Pole–Zero Systems with Nonzero Initial Conditions

Suppose that the signal $x(n)$ is applied to the pole–zero system at $n = 0$. Thus the signal $x(n)$ is assumed to be causal. The effects of all previous input signals to the system are reflected in the initial conditions $y(-1), y(-2), \ldots, y(-N)$. Since the input $x(n)$ is causal and since we are interested in determining the output $y(n)$ for $n \geq 0$, we can use the one-sided $z$-transform, which allows us to deal with the initial conditions. Thus the one-sided $z$-transform of (3.3.7) becomes

$$Y^+(z) = -\sum_{k=1}^{N} a_k z^{-k}\left[Y^+(z) + \sum_{n=1}^{k} y(-n)z^n\right] + \sum_{k=0}^{M} b_k z^{-k} X^+(z) \tag{3.6.14}$$

Since $x(n)$ is causal, we can set $X^+(z) = X(z)$. In any case (3.6.14) may be expressed as

$$\begin{aligned} Y^+(z) &= \frac{\sum_{k=0}^{M} b_k z^{-k}}{1+\sum_{k=1}^{N} a_k z^{-k}} X(z) - \frac{\sum_{k=1}^{N} a_k z^{-k} \sum_{n=1}^{k} y(-n)z^n}{1+\sum_{k=1}^{N} a_k z^{-k}} \\ &= H(z)X(z) + \frac{N_0(z)}{A(z)} \end{aligned} \tag{3.6.15}$$

where

$$N_0(z) = -\sum_{k=1}^{N} a_k z^{-k} \sum_{n=1}^{k} y(-n) z^n \tag{3.6.16}$$

From (3.6.15) it is apparent that the output of the system with nonzero initial conditions can be subdivided into two parts. The first is the zero-state response of the system, defined in the $z$-domain as

$$Y_{zs}(z) = H(z)X(z) \tag{3.6.17}$$

The second component corresponds to the output resulting from the nonzero initial conditions. This output is the zero-input response of the system, which is defined in the $z$-domain as

$$Y_{zi}^{+}(z) = \frac{N_0(z)}{A(z)} \tag{3.6.18}$$

Hence the total response is the sum of these two output components, which can be expressed in the time domain by determining the inverse $z$-transforms of $Y_{zs}(z)$ and $Y_{zi}(z)$ separately, and then adding the results. Thus

$$y(n) = y_{zs}(n) + y_{zi}(n) \tag{3.6.19}$$

Since the denominator of $Y_{zi}^{+}(z)$, is $A(z)$, its poles are $p_1, p_2, \ldots, p_N$. Consequently, the zero-input response has the form

$$y_{zi}(n) = \sum_{k=1}^{N} D_k (p_k)^n u(n) \tag{3.6.20}$$

This can be added to (3.6.4) and the terms involving the poles $\{p_k\}$ can be combined to yield the total response in the form

$$y(n) = \sum_{k=1}^{N} A_k' (p_k)^n u(n) + \sum_{k=1}^{L} Q_k (q_k)^n u(n) \tag{3.6.21}$$

where, by definition,

$$A_k' = A_k + D_k \tag{3.6.22}$$

This development indicates clearly that the effect of the initial conditions is to alter the natural response of the system through modification of the scale factors $\{A_k\}$. There are no new poles introduced by the nonzero initial conditions. Furthermore, there is no effect on the forced response of the system. These important points are reinforced in the following example.

EXAMPLE 3.6.7

Determine the unit step response of the system described by the difference equation

$$y(n) = 0.9y(n-1) - 0.81y(n-2) + x(n)$$

under the following initial conditions $y(-1) = y(-2) = 1$.

**Solution.** The system function is

$$H(z) = \frac{1}{1 - 0.9z^{-1} + 0.81z^{-2}}$$

This system has two complex-conjugate poles at

$$p_1 = 0.9e^{j\pi/3}, \qquad p_2 = 0.9e^{-j\pi/3}$$

The $z$-transform of the unit step sequence is

$$X(z) = \frac{1}{1 - z^{-1}}$$

Therefore,

$$\begin{aligned} Y_{zs}(z) &= \frac{1}{(1 - 0.9e^{j\pi/3}z^{-1})(1 - 0.9e^{-j\pi/3}z^{-1})(1 - z^{-1})} \\ &= \frac{0.0496 - j0.542}{1 - 0.9e^{j\pi/3}z^{-1}} + \frac{0.0496 + j0.542}{1 - 0.9e^{-j\pi/3}z^{-1}} + \frac{1.099}{1 - z^{-1}} \end{aligned}$$

and hence the zero-state response is

$$y_{zs}(n) = \left[1.099 + 1.088(0.9)^n \cos\left(\frac{\pi}{3}n - 5.2^\circ\right)\right]u(n)$$

For the initial conditions $y(-1) = y(-2) = 1$, the additional component in the $z$-transform is

$$\begin{aligned} Y_{zi}(z) &= \frac{N_0(z)}{A(z)} = \frac{0.09 - 0.81z^{-1}}{1 - 0.9z^{-1} + 0.81z^{-2}} \\ &= \frac{0.045 + j0.4936}{1 - 0.9e^{j\pi/3}z^{-1}} + \frac{0.045 - j0.4936}{1 - 0.9e^{-j\pi/3}z^{-1}} \end{aligned}$$

Consequently, the zero-input response is

$$y_{zi}(n) = 0.988(0.9)^n \cos\left(\frac{\pi}{3}n + 87^\circ\right)u(n)$$

In this case the total response has the $z$-transform

$$\begin{aligned} Y(z) &= Y_{zs}(z) + Y_{zi}(z) \\ &= \frac{1.099}{1 - z^{-1}} + \frac{0.568 + j0.445}{1 - 0.9e^{j\pi/3}z^{-1}} + \frac{0.568 - j0.445}{1 - 0.9e^{-j\pi/3}z^{-1}} \end{aligned}$$

The inverse transform yields the total response in the form

$$y(n) = 1.099u(n) + 1.44(0.9)^n \cos\left(\frac{\pi}{3}n + 38^\circ\right)u(n)$$

## 3.7 Summary and References

The $z$-transform plays the same role in discrete-time signals and systems as the Laplace transform does in continuous-time signals and systems. In this chapter we derived the important properties of the $z$-transform, which are extremely useful in the analysis of discrete-time systems. Of particular importance is the convolution property, which transforms the convolution of two sequences into a product of their $z$-transforms.

In the context of LTI systems, the convolution property results in the product of the $z$-transform $X(z)$ of the input signal with the system function $H(z)$, where the latter is the $z$-transform of the unit sample response of the system. This relationship allows us to determine the output of an LTI system in response to an input with transform $X(z)$ by computing the product $Y(z) = H(z)X(z)$ and then determining the inverse $z$-transform of $Y(z)$ to obtain the output sequence $y(n)$.

We observed that many signals of practical interest have rational $z$-transforms. Moreover, LTI systems characterized by constant-coefficient linear difference equations also possess rational system functions. Consequently, in determining the inverse $z$-transform, we naturally emphasized the inversion of rational transforms. For such transforms, the partial-fraction expansion method is relatively easy to apply, in conjunction with the ROC, to determine the corresponding sequence in the time domain.

We considered the characterization of LTI systems in the $z$-transform domain. In particular, we related the pole–zero locations of a system to its time-domain characteristics and restated the requirements for stability and causality of LTI systems in terms of the pole locations. We demonstrated that a causal system has a system function $H(z)$ with a ROC $|z| > r_1$, where $0 < r_1 \leq \infty$. In a stable and causal system, the poles of $H(z)$ lie inside the unit circle. On the other hand, if the system is noncausal, the condition for stability requires that the unit circle be contained in the ROC of $H(z)$. Hence a noncausal stable LTI system has a system function with poles both inside and outside the unit circle with an annular ROC that includes the unit circle. Finally, the one-sided $z$-transform was introduced to solve for the response of causal systems excited by causal input signals with nonzero initial conditions.

## Problems

**3.1** Determine the $z$-transform of the following signals.

**(a)** $x(n) = \{3, 0, 0, 0, 0, \underset{\uparrow}{6}, 1, -4\}$

**(b)** $x(n) = \begin{cases} (\frac{1}{2})^n, & n \geq 5 \\ 0, & n \leq 4 \end{cases}$

**3.2** Determine the $z$-transforms of the following signals and sketch the corresponding pole–zero patterns.

**(a)** $x(n) = (1 + n)u(n)$

**(b)** $x(n) = (a^n + a^{-n})u(n)$, $a$ real

**(c)** $x(n) = (-1)^n 2^{-n} u(n)$

**(d)** $x(n) = (na^n \sin \omega_0 n)u(n)$

**(e)** $x(n) = (na^n \cos \omega_0 n)u(n)$

**(f)** $x(n) = Ar^n \cos(\omega_0 n + \phi)u(n), 0 < r < 1$

**(g)** $x(n) = \frac{1}{2}(n^2 + n)(\frac{1}{3})^{n-1}u(n-1)$

**(h)** $x(n) = (\frac{1}{2})^n[u(n) - u(n-10)]$

**3.3** Determine the $z$-transforms and sketch the ROC of the following signals.

**(a)** $x_1(n) = \begin{cases} (\frac{1}{3})^n, & n \geq 0 \\ (\frac{1}{2})^{-n}, & n < 0 \end{cases}$

**(b)** $x_2(n) = \begin{cases} (\frac{1}{3})^n - 2^n, & n \geq 0 \\ 0, & n < 0 \end{cases}$

**(c)** $x_3(n) = x_1(n+4)$

**(d)** $x_4(n) = x_1(-n)$

**3.4** Determine the $z$-transform of the following signals.

**(a)** $x(n) = n(-1)^n u(n)$

**(b)** $x(n) = n^2 u(n)$

**(c)** $x(n) = -na^n u(-n-1)$

**(d)** $x(n) = (-1)^n \left(\cos \frac{\pi}{3} n\right) u(n)$

**(e)** $x(n) = (-1)^n u(n)$

**(f)** $x(n) = \{\underset{\uparrow}{1}, 0, -1, 0, 1, -1, \ldots\}$

**3.5** Determine the regions of convergence of right-sided, left-sided, and finite-duration two-sided sequences.

**3.6** Express the $z$-transform of

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

in terms of $X(z)$. [*Hint*: Find the difference $y(n) - y(n-1)$.]

**3.7** Compute the convolution of the following signals by means of the $z$-transform.

$$x_1(n) = \begin{cases} (\frac{1}{3})^n, & n \geq 0 \\ (\frac{1}{2})^{-n}, & n < 0 \end{cases}$$

$$x_2(n) = (\frac{1}{2})^n u(n)$$

**3.8** Use the convolution property to:

**(a)** Express the $z$-transform of

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

in terms of $X(z)$.

**(b)** Determine the $z$-transform of $x(n) = (n+1)u(n)$. [*Hint*: Show first that $x(n) = u(n) * u(n)$.]

**3.9** The $z$-transform $X(z)$ of a real signal $x(n)$ includes a pair of complex-conjugate zeros and a pair of complex-conjugate poles. What happens to these pairs if we multiply $x(n)$ by $e^{j\omega_0 n}$? (*Hint*: Use the scaling theorem in the $z$-domain.)

**3.10** Apply the final value theorem to determine $x(\infty)$ for the signal

$$x(n) = \begin{cases} 1, & \text{if } n \text{ is even} \\ 0, & \text{otherwise} \end{cases}$$

**3.11** Using long division, determine the inverse $z$-transform of

$$X(z) = \frac{1 + 2z^{-1}}{1 - 2z^{-1} + z^{-2}}$$

if **(a)** $x(n)$ is causal and **(b)** $x(n)$ is anticausal.

**3.12** Determine the causal signal $x(n)$ having the $z$-transform

$$X(z) = \frac{1}{(1 - 2z^{-1})(1 - z^{-1})2}$$

**3.13** Let $x(n)$ be a sequence with $z$-transform $X(z)$. Determine, in terms of $X(z)$, the $z$-transforms of the following signals.

**(a)** $x_1(n) = \begin{cases} x\left(\frac{n}{2}\right), & \text{if } n \text{ even} \\ 0, & \text{if } n \text{ odd} \end{cases}$

**(b)** $x_2(n) = x(2n)$

**3.14** Determine the causal signal $x(n)$ if its $z$-transform $X(z)$ is given by:

**(a)** $X(z) = \frac{1+3z^{-1}}{1+3z^{-1}+2z^{-2}}$

**(b)** $X(z) = \frac{1}{1-z^{-1}+\frac{1}{2}z^{-2}}$

**(c)** $X(z) = \frac{z^{-6}+z^{-7}}{1-z^{-1}}$

**(d)** $X(z) = \frac{1+2z^{-2}}{1+z^{-2}}$

**(e)** $X(z) = \frac{1}{4}\frac{1+6z^{-1}+z^{-2}}{(1-2z^{-1}+2z^{-2})(1-0.5z^{-1})}$

**(f)** $X(z) = \frac{2-1.5z^{-1}}{1-1.5z^{-1}+0.5z^{-2}}$

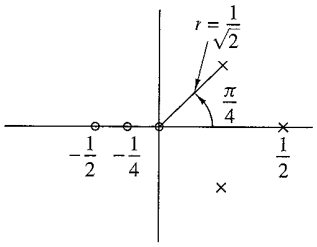



**Figure P3.14**

**(g)** $X(z) = \frac{1+2z^{-1}+z^{-2}}{1+4z^{-1}+4z^{-2}}$

**(h)** $X(z)$ is specified by a pole–zero pattern in Fig. P3.14. The constant $G = \frac{1}{4}$.

**(i)** $X(z) = \frac{1-\frac{1}{2}z^{-1}}{1+\frac{1}{2}z^{-1}}$

**(j)** $X(z) = \frac{1-az^{-1}}{z^{-1}-a}$

**3.15** Determine all possible signals $x(n)$ associated with the $z$-transform

$$X(z) = \frac{5z^{-1}}{(1-2z^{-1})(3-z^{-1})}$$

**3.16** Determine the convolution of the following pairs of signals by means of the $z$-transform.

**(a)** $x_1(n) = (\frac{1}{4})^n u(n-1), \quad x_2(n) = [1 + (\frac{1}{2})^n]u(n)$

**(b)** $x_1(n) = u(n), \quad x_2(n) = \delta(n) + (\frac{1}{2})^n u(n)$

**(c)** $x_1(n) = (\frac{1}{2})^n u(n), \quad x_2(n) = \cos \pi n u(n)$

**(d)** $x_1(n) = nu(n), \quad x_2(n) = 2^n u(n-1)$

**3.17** Prove the final value theorem for the one-sided $z$-transform.

**3.18** If $X(z)$ is the $z$-transform of $x(n)$, show that:

**(a)** $Z\{x^*(n)\} = X^*(z^*)$

**(b)** $Z\{\text{Re}[x(n)]\} = \frac{1}{2}[X(z) + X^*(z^*)]$

**(c)** $Z\{\text{Im}[x(n)]\} = \frac{1}{2}[X(z) - X^*(z^*)]$

**(d)** If

$$x_k(n) = \begin{cases} x\left(\frac{n}{k}\right), & \text{if } n/k \text{ integer} \\ 0, & \text{otherwise} \end{cases}$$

then

$$X_k(z) = X(z^k)$$

**(e)** $Z\{e^{j\omega_0 n}x(n)\} = X(ze^{-j\omega_0})$

**3.19** By first differentiating $X(z)$ and then using appropriate properties of the $z$-transform, determine $x(n)$ for the following transforms.

**(a)** $X(z) = \log(1-2z), \quad |z| < \frac{1}{2}$

**(b)** $X(z) = \log(1-z^{-1}), \quad |z| > \frac{1}{2}$

**3.20**

**(a)** Draw the pole–zero pattern for the signal

$$x_1(n) = (r^n \sin \omega_0 n)u(n), \qquad 0 < r < 1$$

**(b)** Compute the $z$-transform $X_2(z)$, which corresponds to the pole–zero pattern in part (a).

**(c)** Compare $X_1(z)$ with $X_2(z)$. Are they indentical? If not, indicate a method to derive $X_1(z)$ from the pole–zero pattern.

**3.21** Show that the roots of a polynomial with real coefficients are real or form complex-conjugate pairs. The inverse is not true, in general.

**3.22** Prove the convolution and correlation properties of the $z$-transform using only its definition.

**3.23** Determine the signal $x(n)$ with $z$-transform

$$X(z) = e^{z} + e^{1/z}, \qquad |z| \neq 0$$

**3.24** Determine, in closed form, the causal signals $x(n)$ whose $z$-transforms are given by:

**(a)** $X(z) = \frac{1}{1+1.5z^{-1}-0.5z^{-2}}$

**(b)** $X(z) = \frac{1}{1-0.5z^{-1}+0.6z^{-2}}$

Partially check your results by computing $x(0)$, $x(1)$, $x(2)$, and $x(\infty)$ by an alternative method.

**3.25** Determine all possible signals that can have the following $z$-transforms.

**(a)** $X(z) = \frac{1}{1-1.5z^{-1}+0.5z^{-2}}$

**(b)** $X(z) = \frac{1}{1-\frac{1}{2}z^{-1}+\frac{1}{4}z^{-2}}$

**3.26** Determine the signal $x(n)$ with $z$-transform

$$X(z) = \frac{3}{1-\frac{10}{3}z^{-1}+z^{-2}}$$

if $X(z)$ converges on the unit circle.

**3.27** Prove the complex convolution relation given by (3.2.22).

**3.28** Prove the conjugation properties and Parseval's relation for the $z$-transform given in Table 3.2.

**3.29** In Example 3.4.1 we solved for $x(n)$, $n < 0$, by performing contour integrations for each value of $n$. In general, this procedure proves to be tedious. It can be avoided by making a transformation in the contour integral from $z$-plane to the $w = 1/z$ plane. Thus a circle of radius $R$ in the $z$-plane is mapped into a circle of radius $1/R$ in the $w$-plane. As a consequence, a pole inside the unit circle in the $z$-plane is mapped into a pole outside the unit circle in the $w$-plane. By making the change of variable $w = 1/z$ in the contour integral, determine the sequence $x(n)$ for $n < 0$ in Example 3.4.1.

**3.30** Let $x(n)$, $0 \leq n \leq N-1$ be a finite-duration sequence, which is also real-valued and even. Show that the zeros of the polynomial $X(z)$ occur in mirror-image pairs about the unit circle. That is, if $z = re^{j\theta}$ is a zero of $X(z)$, then $z = (1/r)e^{j\theta}$ is also a zero.

**3.31** Prove that the Fibonacci sequence can be thought of as the impulse response of the system described by the difference equation $y(n) = y(n-1) + y(n-2) + x(n)$. Then determine $h(n)$ using $z$-transform techniques.

**3.32** Show that the following systems are equivalent.

**(a)** $y(n) = 0.2y(n-1) + x(n) - 0.3x(n-1) + 0.02x(n-2)$

**(b)** $y(n) = x(n) - 0.1x(n-1)$

**3.33** Consider the sequence $x(n) = a^n u(n)$, $-1 < a < 1$. Determine at least two sequences that are not equal to $x(n)$ but have the same autocorrelation.

**3.34** Compute the unit step response of the system with impulse response

$$h(n) = \begin{cases} 3^n, & n < 0 \\ (\frac{2}{5})^n, & n \geq 0 \end{cases}$$

**3.35** Compute the zero-state response for the following pairs of systems and input signals.

**(a)** $h(n) = (\frac{1}{3})^n u(n), \quad x(n) = (\frac{1}{2})^n (\cos \frac{\pi}{3} n) u(n)$

**(b)** $h(n) = (\frac{1}{2})^n u(n), \quad x(n) = (\frac{1}{3})^n u(n) + (\frac{1}{2})^{-n} u(-n-1)$

**(c)** $y(n) = -0.1y(n-1) + 0.2y(n-2) + x(n) + x(n-1) x(n) = (\frac{1}{3})^n u(n)$

**(d)** $y(n) = \frac{1}{2}x(n) - \frac{1}{2}x(n-1) x(n) = 10(\cos \frac{\pi}{2} n) u(n)$

**(e)** $y(n) = -y(n-2) + 10x(n) x(n) = 10(\cos \frac{\pi}{2} n) u(n)$

**(f)** $h(n) = (\frac{2}{5})^n u(n), \quad x(n) = u(n) - u(n-7)$

**(g)** $h(n) = (\frac{1}{2})^n u(n), \quad x(n) = (-1)^n, \quad -\infty < n < \infty$

**(h)** $h(n) = (\frac{1}{2})^n u(n), \quad x(n) = (n+1)(\frac{1}{4})^n u(n)$

**3.36** Consider the system

$$H(z) = \frac{1 - 2z^{-1} + 2z^{-2} - z^{-3}}{(1 - z^{-1})(1 - 0.5z^{-1})(1 - 0.2z^{-1})}, \qquad \text{ROC: } 0.5|z| > 1$$

**(a)** Sketch the pole–zero pattern. Is the system stable?

**(b)** Determine the impulse response of the system.

**3.37** Compute the response of the system

$$y(n) = 0.7y(n-1) - 0.12y(n-2) + x(n-1) + x(n-2)$$

to the input $x(n) = nu(n)$. Is the system stable?

**3.38** Determine the impulse response and the step response of the following causal systems. Plot the pole–zero patterns and determine which of the systems are stable.

**(a)** $y(n) = \frac{3}{4}y(n-1) - \frac{1}{8}y(n-2) + x(n)$

**(b)** $y(n) = y(n-1) - 0.5y(n-2) + x(n) + x(n-1)$

**(c)** $H(z) = \frac{z^{-1}(1+z^{-1})}{(1-z^{-1})^3}$

**(d)** $y(n) = 0.6y(n-1) - 0.08y(n-2) + x(n)$

**(e)** $y(n) = 0.7y(n-1) - 0.1y(n-2) + 2x(n) - x(n-2)$

**3.39** Let $x(n)$ be a causal sequence with $z$-transform $X(z)$ whose pole–zero plot is shown in Fig. P3.39. Sketch the pole–zero plots and the ROC of the following sequence:

**(a)** $x_1(n) = x(-n+2)$

**(b)** $x_2(n) = e^{j(\pi/3)n} x(n)$

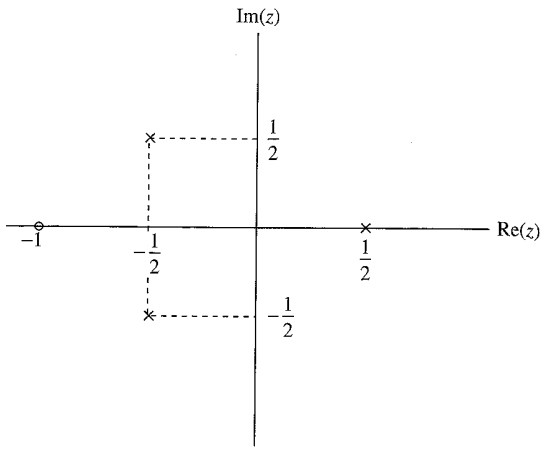



**Figure P3.39**

**3.40** We want to design a causal discrete-time LTI system with the property that if the input is

$$x(n) = (\frac{1}{2})^n u(n) - \frac{1}{4}(\frac{1}{2})^{n-1} u(n-1)$$

then the output is

$$y(n) = (\frac{1}{3})^n u(n)$$

**(a)** Determine the impulse response $h(n)$ and the system function $H(z)$ of a system that satisfies the foregoing conditions.

**(b)** Find the difference equation that characterizes this system.

**(c)** Determine a realization of the system that requires the minimum possible amount of memory.

**(d)** Determine if the system is stable.

**3.41** Determine the stability region for the causal system

$$H(z) = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2}}$$

by computing its poles and restricting them to be inside the unit circle.

**3.42** Consider the system

$$H(z) = \frac{z^{-1} + \frac{1}{2} z^{-2}}{1 - \frac{3}{5} z^{-1} + \frac{2}{25} z^{-2}}$$

Determine:

**(a)** The impulse response

**(b)** The zero-state step response

**(c)** The step response if $y(-1) = 1$ and $y(-2) = 2$

**3.43** Determine the system function, impulse response, and zero-state step response of the system shown in Fig P3.43.

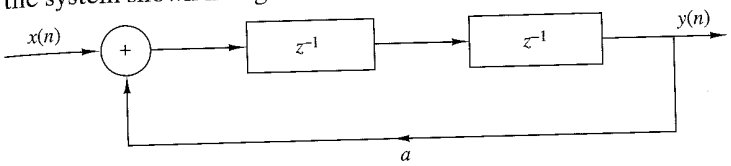



**Figure P3.43**

**3.44** Consider the causal system

$$y(n) = -a_1 y(n-1) + b_0 x(n) + b_1 x(n-1)$$

Determine:

**(a)** The impulse response

**(b)** The zero-state step response

**(c)** The step response if $y(-1) = A \neq 0$

**(d)** The response to the input

$$x(n) = \cos \omega_0 n, \qquad 0 \leq n < \infty$$

**3.45** Determine the zero-state response of the system

$$y(n) = \frac{1}{2} y(n-1) + 4x(n) + 3x(n-1)$$

to the input

$$x(n) = e^{j\omega_0 n} u(n)$$

What is the steady-state response of the system?

**3.46** Consider the causal system defined by the pole–zero pattern shown in Fig. P3.46.

**(a)** Determine the system function and the impulse response of the system given that $H(z)|_{z=1} = 1$.

**(b)** Is the system stable?

**(c)** Sketch a possible implementation of the system and determine the corresponding difference equations.

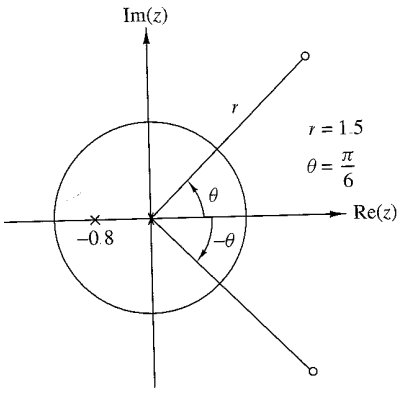



**Figure P3.46**

**3.47** Compute the convolution of the following pair of signals in the time domain and by using the one-sided $z$-transform.

**(a)** $x_1(n) = \{1, 1, \underset{\uparrow}{1}, 1, 1\}$, $\quad x_2(n) = \{\underset{\uparrow}{1}, 1, 1\}$

**(b)** $x_1(n) = (\frac{1}{2})^n u(n)$, $\quad x_2(n) = (\frac{1}{3})^n u(n)$

**(c)** $x_1(n) = \{1, \underset{\uparrow}{2}, 3, 4\}$, $\quad x_2(n) = \{4, 3, \underset{\uparrow}{2}, 1\}$

**(d)** $x_1(n) = \{\underset{\uparrow}{1}, 1, 1, 1, 1\}$, $\quad x_2(n) = \{\underset{\uparrow}{1}, 1, 1\}$

Did you obtain the same results by both methods? Explain.

**3.48** Determine the one-sided $z$-transform of the constant signal $x(n) = 1$, $-\infty < n < \infty$.

**3.49** Use the one-sided $z$-transform to determine $y(n)$, $n \geq 0$ in the following cases.

**(a)** $y(n) + \frac{1}{2}y(n-1) - \frac{1}{4}y(n-2) = 0$; $\quad y(-1) = y(-2) = 1$

**(b)** $y(n) - 1.5y(n-1) + 0.5y(n-2) = 0$; $\quad y(-1) = 1, y(-2) = 0$

**(c)** $y(n) = \frac{1}{2}y(n-1) + x(n)x(n) = (\frac{1}{3})^n u(n)$, $\quad y(-1) = 1$

**(d)** $y(n) = \frac{1}{4}y(n-2) + x(n)x(n) = u(n)y(-1) = 0$; $\quad y(-2) = 1$

**3.50** An FIR LTI system has an impulse response $h(n)$, which is real valued, even, and has finite duration of $2N + 1$. Show that if $z_1 = re^{j\omega_0}$ is a zero of the system, then $z_1 = (1/r)e^{j\omega_0}$ is also a zero.

**3.51** Consider an LTI discrete-time system whose pole–zero pattern is shown in Fig. P3.51.

**(a)** Determine the ROC of the system function $H(z)$ if the system is known to be stable.

**(b)** It is possible for the given pole–zero plot to correspond to a causal and stable system? If so, what is the appropriate ROC?

**(c)** How many possible systems can be associated with this pole–zero pattern?

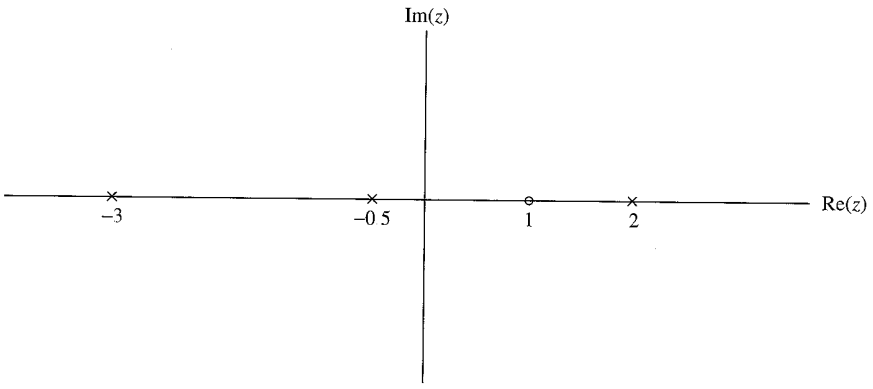



Figure P3.51

**3.52** Let $x(n)$ be a causal sequence.

**(a)** What conclusion can you draw about the value of its $z$-transform $X(z)$ at $z = \infty$?

**(b)** Use the result in part (a) to check which of the following transforms cannot be associated with a causal sequence.

$$\text{(i)} X(z) = \frac{(z-\frac{1}{2})^4}{(z-\frac{1}{3})^3} \qquad \text{(ii)} X(z) = \frac{(1-\frac{1}{2}z^{-1})^2}{(1-\frac{1}{3}z^{-1})} \qquad \text{(iii)} X(z) = \frac{(z-\frac{1}{3})^2}{(z-\frac{1}{2})^3}$$

**3.53** A causal pole–zero system is BIBO stable if its poles are inside the unit circle. Consider now a pole–zero system that is BIBO stable and has its poles inside the unit circle. Is the system always causal? [*Hint:* Consider the systems $h_1(n) = a^n u(n)$ and $h_2(n) = a^n u(n+3)$, $|a| < 1$.]

**3.54** Let $x(n)$ be an anticausal signal [i.e., $x(n) = 0$ for $n > 0$]. Formulate and prove an initial value theorem for anticausal signals.

**3.55** The step response of an LTI system is

$$s(n) = (\frac{1}{3})^{n-2} u(n+2)$$

**(a)** Find the system function $H(z)$ and sketch the pole–zero plot.

**(b)** Determine the impulse response $h(n)$.

**(c)** Check if the system is causal and stable.

**3.56** Use contour integration to determine the sequence $x(n)$ whose $z$-transform is given by

**(a)** $X(z) = \frac{1}{1-\frac{1}{2}z^{-1}}, \qquad |z| > \frac{1}{2}$

**(b)** $X(z) = \frac{1}{1-\frac{1}{2}z^{-1}}, \qquad |z| < \frac{1}{2}$

**(c)** $X(z) = \frac{z-a}{1-az}, \qquad |z| > |1/a|$

**(d)** $X(z) = \frac{1-\frac{1}{4}z^{-1}}{1-\frac{1}{6}z^{-1}-\frac{1}{6}z^{-2}}, \qquad |z| > \frac{1}{2}$

**3.57** Let $x(n)$ be a sequence with $z$-transform

$$X(z) = \frac{1-a^2}{(1-az)(1-az^{-1})}, \qquad \text{ROC: } a > |z| > 1/a$$

with $0 < a < 1$. Determine $x(n)$ by using contour integration.

**3.58** The $z$-transform of a sequence $x(n)$ is given by

$$X(z) = \frac{z^{20}}{(z-\frac{1}{2})(z-2)^5(z+\frac{5}{2})^2(z+3)}$$

Furthermore it is known that $X(z)$ converges for $|z| = 1$.

**(a)** Determine the ROC of $X(z)$.

**(b)** Determine $x(n)$ at $n = -18$. (*Hint:* Use contour integration.)

ElCoM
Electrical | Computer | Mechatronics
ELCOM-HU.COM

CHAPTER

4

# Frequency Analysis of Signals

The Fourier transform is one of several mathematical tools that is useful in the analysis and design of LTI systems. Another is the Fourier series. These signal representations basically involve the decomposition of the signals in terms of sinusoidal (or complex exponential) components. With such a decomposition, a signal is said to be represented in the *frequency domain*.

As we shall demonstrate, most signals of practical interest can be decomposed into a sum of sinusoidal signal components. For the class of periodic signals, such a decomposition is called a *Fourier series*. For the class of finite energy signals, the decomposition is called the *Fourier transform*. These decompositions are extremely important in the analysis of LTI systems because the response of an LTI system to a sinusoidal input signal is a sinusoid of the same frequency but of different amplitude and phase. Furthermore, the linearity property of the LTI system implies that a linear sum of sinusoidal components at the input produces a similar linear sum of sinusoidal components at the output, which differ only in the amplitudes and phases from the input sinusoids. This characteristic behavior of LTI systems renders the sinusoidal decomposition of signals very important. Although many other decompositions of signals are possible, only the class of sinusoidal (or complex exponential) signals possess this desirable property in passing through an LTI system.

We begin our study of frequency analysis of signals with the representation of continuous-time periodic and aperiodic signals by means of the Fourier series and the Fourier transform, respectively. This is followed by a parallel treatment of discrete-time periodic and aperiodic signals. The properties of the Fourier transform are described in detail and a number of time-frequency dualities are presented.

## 4.1 Frequency Analysis of Continuous-Time Signals

It is well known that a prism can be used to break up white light (sunlight) into the colors of the rainbow (see Fig. 4.1.1(a)). In a paper submitted in 1672 to the Royal Society, Isaac Newton used the term *spectrum* to describe the *continuous* bands of colors produced by this apparatus. To understand this phenomenon, Newton placed another prism upside-down with respect to the first, and showed that the colors blended back into white light, as in Fig. 4.1.1(b). By inserting a slit between the two prisms and blocking one or more colors from hitting the second prism, he showed that the remixed light is no longer white. Hence the light passing through the first prism is simply analyzed into its component colors without any other change. However, only if we mix again all of these colors do we obtain the original white light.

Later, Joseph Fraunhofer (1787–1826), in making measurements of light emitted by the sun and stars, discovered that the spectrum of the observed light consists of distinct color lines. A few years later (mid-1800s) Gustav Kirchhoff and Robert Bunsen found that each chemical element, when heated to incandescence, radiated its own distinct color of light. As a consequence, each chemical element can be identified by its own *line spectrum*.

From physics we know that each color corresponds to a specific frequency of the visible spectrum. Hence the analysis of light into colors is actually a form of *frequency analysis*.

Frequency analysis of a signal involves the resolution of the signal into its frequency (sinusoidal) components. Instead of light, our signal waveforms are basically functions of time. The role of the prism is played by the Fourier analysis tools that we will develop: the Fourier series and the Fourier transform. The recombination of the sinusoidal components to reconstruct the original signal is basically a Fourier synthesis problem. The problem of signal analysis is basically the same for the case of a signal waveform and for the case of the light from heated chemical composi-

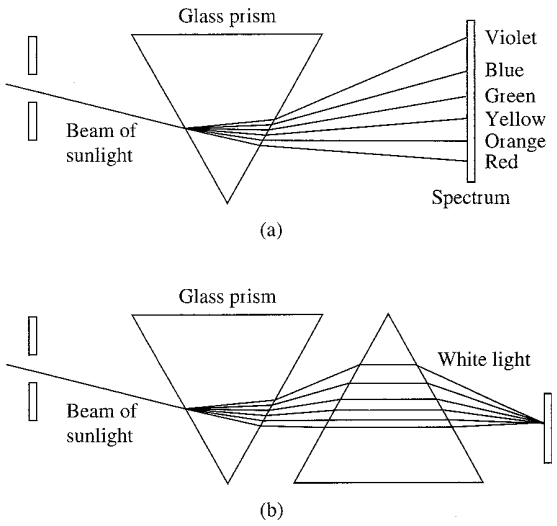



**Figure 4.1.1**
(a) Analysis and
(b) synthesis of the white light (sunlight) using glass prisms.

tions. Just as in the case of chemical compositions, different signal waveforms have different spectra. Thus the spectrum provides an "identity" or a signature for the signal in the sense that no other signal has the same spectrum. As we will see, this attribute is related to the mathematical treatment of frequency-domain techniques.

If we decompose a waveform into sinusoidal components, in much the same way that a prism separates white light into different colors, the sum of these sinusoidal components results in the original waveform. On the other hand, if any of these components is missing, the result is a different signal.

In our treatment of frequency analysis, we will develop the proper mathematical tools ("prisms") for the decomposition of signals ("light") into sinusoidal frequency components (colors). Furthermore, the tools ("inverse prisms") for synthesis of a given signal from its frequency components will also be developed.

The basic motivation for developing the frequency analysis tools is to provide a mathematical and pictorial representation for the frequency components that are contained in any given signal. As in physics, the term *spectrum* is used when referring to the frequency content of a signal. The process of obtaining the spectrum of a given signal using the basic mathematical tools described in this chapter is known as *frequency* or *spectral analysis*. In contrast, the process of determining the spectrum of a signal in practice, based on actual measurements of the signal, is called *spectrum estimation*. This distinction is very important. In a practical problem the signal to be analyzed does not lend itself to an exact mathematical description. The signal is usually some information-bearing signal from which we are attempting to extract the relevant information. If the information that we wish to extract can be obtained either directly or indirectly from the spectral content of the signal, we can perform *spectrum estimation*on the information-bearing signal, and thus obtain an estimate of the signal spectrum. In fact, we can view spectral estimation as a type of spectral analysis performed on signals obtained from physical sources (e.g., speech, EEG, ECG, etc.). The instruments or software programs used to obtain spectral estimates of such signals are known as *spectrum analyzers*.

Here, we will deal with spectral analysis. However, in Chapter 14 we shall treat the subject of power spectrum estimation.

### 4.1.1 The Fourier Series for Continuous-Time Periodic Signals

In this section we present the frequency analysis tools for continuous-time periodic signals. Examples of periodic signals encountered in practice are square waves, rectangular waves, triangular waves, and of course, sinusoids and complex exponentials.

The basic mathematical representation of periodic signals is the Fourier series, which is a linear weighted sum of harmonically related sinusoids or complex exponentials. Jean Baptiste Joseph Fourier (1768–1830), a French mathematician, used such trigonometric series expansions in describing the phenomenon of heat conduction and temperature distribution through bodies. Although his work was motivated by the problem of heat conduction, the mathematical techniques that he developed during the early part of the nineteenth century now find application in a variety of problems encompassing many different fields, including optics, vibrations in mechanical systems, system theory, and electromagnetics.

From Chapter 1 we recall that a linear combination of harmonically related complex exponentials of the form

$$x(t) = \sum_{k=-\infty}^{\infty} c_k e^{j2\pi k F_0 t} \tag{4.1.1}$$

is a periodic signal with fundamental period $T_p = 1/F_0$. Hence we can think of the exponential signals

$$\{e^{j2\pi k F_0 t}, \qquad k = 0, \pm 1, \pm 2, \ldots\}$$

as the basic "building blocks" from which we can construct periodic signals of various types by proper choice of the fundamental frequency and the coefficients $\{c_k\}$. $F_0$ determines the fundamental period of $x(t)$ and the coefficients $\{c_k\}$ specify the shape of the waveform.

Suppose that we are given a periodic signal $x(t)$ with period $T_p$. We can represent the periodic signal by the series (4.1.1), called a *Fourier series*, where the fundamental frequency $F_0$ is selected to be the reciprocal of the given period $T_p$. To determine the expression for the coefficients $\{c_k\}$, we first multiply both sides of (4.1.1) by the complex exponential

$$e^{-j2\pi F_0 l t}$$

where $l$ is an integer and then integrate both sides of the resulting equation over a single period, say from 0 to $T_p$, or more generally, from $t_0$ to $t_0 + T_p$, where $t_0$ is an arbitrary but mathematically convenient starting value. Thus we obtain

$$\int_{t_0}^{t_0+T_p} x(t) e^{-j2\pi l F_0 t}\, dt = \int_{t_0}^{t_0+T_p} e^{-j2\pi l F_0 t} \left( \sum_{k=-\infty}^{\infty} c_k e^{+j2\pi k F_0 t} \right) dt \tag{4.1.2}$$

To evaluate the integral on the right-hand side of (4.1.2), we interchange the order of the summation and integration and combine the two exponentials. Hence

$$\sum_{k=-\infty}^{\infty} c_k \int_{t_0}^{t_0+T_p} e^{j2\pi F_0 (k-l) t}\, dt = \sum_{k=-\infty}^{\infty} c_k \left[ \frac{e^{j2\pi F_0 (k-l) t}}{j2\pi F_0 (k-l)} \right]_{t_0}^{t_0+T_p} \tag{4.1.3}$$

For $k \neq l$, the right-hand side of (4.1.3) evaluated at the lower and upper limits, $t_0$ and $t_0 + T_p$, respectively, yields zero. On the other hand, if $k = l$, we have

$$\int_{t_0}^{t_0+T_p} dt = t \Big|_{t_0}^{t_0+T_p} = T_p$$

Consequently, (4.1.2) reduces to

$$\int_{t_0}^{t_0+T_p} x(t) e^{-j2\pi l F_0 t}\, dt = c_l T_p$$

and therefore the expression for the Fourier coefficients in terms of the given periodic signal becomes

$$c_l = \frac{1}{T_p} \int_{t_0}^{t_0+T_p} x(t) e^{-j2\pi l F_0 t}\, dt$$

Since $t_0$ is arbitrary, this integral can be evaluated over any interval of length $T_p$, that is, over any interval equal to the period of the signal $x(t)$. Consequently, the integral for the Fourier series coefficients will be written as

$$c_l = \frac{1}{T_p} \int_{T_p} x(t) e^{-j2\pi l F_0 t}\, dt \tag{4.1.4}$$

An important issue that arises in the representation of the periodic signal $x(t)$ by the Fourier series is whether or not the series converges to $x(t)$ for every value of $t$, that is, whether the signal $x(t)$ and its Fourier series representation

$$\sum_{k=-\infty}^{\infty} c_k e^{j2\pi k F_0 t} \tag{4.1.5}$$

are equal at every value of $t$. The so-called *Dirichlet conditions* guarantee that the series (4.1.5) will be equal to $x(t)$, except at the values of $t$ for which $x(t)$ is discontinuous. At these values of $t$, (4.1.5) converges to the midpoint (average value) of the discontinuity. The Dirichlet conditions are:

1. The signal $x(t)$ has a finite number of discontinuities in any period.
2. The signal $x(t)$ contains a finite number of maxima and minima during any period.
3. The signal $x(t)$ is absolutely integrable in any period, that is,

$$\int_{T_p} |x(t)|\, dt < \infty \tag{4.1.6}$$

All periodic signals of practical interest satisfy these conditions.

The weaker condition, that the signal has finite energy in one period,

$$\int_{T_p} |x(t)|^2\, dt < \infty \tag{4.1.7}$$

guarantees that the energy in the difference signal

$$e(t) = x(t) - \sum_{k=-\infty}^{\infty} c_k e^{j2\pi k F_0 t}$$

is zero, although $x(t)$ and its Fourier series may not be equal for all values of $t$. Note that (4.1.6) implies (4.1.7), but not vice versa. Also, both (4.1.7) and the Dirichlet

conditions are sufficient but not necessary conditions (i.e., there are signals that have a Fourier series representation but do not satisfy these conditions).

In summary, if $x(t)$ is periodic and satisfies the Dirichlet conditions, it can be represented in a Fourier series as in (4.1.1), where the coefficients are specified by (4.1.4). These relations are summarized below.

Frequency Analysis of Continuous-Time Periodic Signals

| | | |
|---|---|---|
| Synthesis equation | $x(t) = \sum_{k=-\infty}^{\infty} c_k e^{j2\pi k F_0 t}$ | (4.1.8) |
| Analysis equation | $c_k = \frac{1}{T_p} \int_{T_p} x(t) e^{-j2\pi k F_0 t} dt$ | (4.1.9) |

In general, the Fourier coefficients $c_k$ are complex valued. Moreover, it is easily shown that if the periodic signal is real, $c_k$ and $c_{-k}$ are complex conjugates. As a result, if

$$c_k = |c_k| e^{j\theta_k}$$

then

$$c_{-k} = |c_k|^{-j\theta_k}$$

Consequently, the Fourier series may also be represented in the form

$$x(t) = c_0 + 2\sum_{k=1}^{\infty} |c_k| \cos(2\pi k F_0 t + \theta_k) \tag{4.1.10}$$

where $c_0$ is real valued when $x(t)$ is real.

Finally, we should indicate that yet another form for the Fourier series can be obtained by expanding the cosine function in (4.1.10) as

$$\cos(2\pi k F_0 t + \theta_k) = \cos 2\pi k F_0 t \cos\theta_k - \sin 2\pi k F_0 t \sin\theta_k$$

Consequently, we can rewrite (4.1.10) in the form

$$x(t) = a_0 + \sum_{k=1}^{\infty} (a_k \cos 2\pi k F_0 t - b_k \sin 2\pi k F_0 t) \tag{4.1.11}$$

where

$$a_0 = c_0$$

$$a_k = 2|c_k| \cos\theta_k$$

$$b_k = 2|c_k| \sin\theta_k$$

The expressions in (4.1.8), (4.1.10), and (4.1.11) constitute three equivalent forms for the Fourier series representation of a real periodic signal.

### 4.1.2 Power Density Spectrum of Periodic Signals

A periodic signal has infinite energy and a finite average power, which is given as

$$P_x = \frac{1}{T_p} \int_{T_p} |x(t)|^2 \, dt \tag{4.1.12}$$

If we take the complex conjugate of (4.1.8) and substitute for $x^*(t)$ in (4.1.12), we obtain

$$\begin{aligned} P_x &= \frac{1}{T_p} \int_{T_p} x(t) \sum_{k=-\infty}^{\infty} c_k^* e^{-j2\pi k F_0 t} \, dt \\ &= \sum_{k=-\infty}^{\infty} c_k^* \left[ \frac{1}{T_p} \int_{T_p} x(t) e^{-j2\pi k F_0 t} \, dt \right] \\ &= \sum_{k=-\infty}^{\infty} |c_k|^2 \end{aligned} \tag{4.1.13}$$

Therefore, we have established the relation

$$P_x = \frac{1}{T_p} \int_{T_p} |x(t)|^2 \, dt = \sum_{k=-\infty}^{\infty} |c_k|^2 \tag{4.1.14}$$

which is called *Parseval's relation*for power signals.

To illustrate the physical meaning of (4.1.14), suppose that $x(t)$ consists of a single complex exponential

$$x(t) = c_k e^{j2\pi k F_0 t}$$

In this case, all the Fourier series coefficients except $c_k$ are zero. Consequently, the average power in the signal is

$$P_x = |c_k|^2$$

It is obvious that $|c_k|^2$ represents the power in the $k$th harmonic component of the signal. Hence the total average power in the periodic signal is simply the sum of the average powers in all the harmonics.

If we plot the $|c_k|^2$ as a function of the frequencies $kF_0$, $k = 0, \pm 1, \pm 2, \ldots$, the diagram that we obtain shows how the power of the periodic signal is distributed among the various frequency components. This diagram, which is illustrated in Fig. 4.1.2, is called the *power density spectrum*[1] of the periodic signal $x(t)$. Since the power in a periodic signal exists only at discrete values of frequencies (i.e., $F = 0, \pm F_0, \pm 2F_0, \ldots$), the signal is said to have a *line spectrum*. The spacing between two consecutive spectral lines is equal to the reciprocal of the fundamental period $T_p$, whereas the shape of the spectrum (i.e., the power distribution of the signal), depends on the time-domain characteristics of the signal.

[1] This function is also called the *power spectral density* or, simply, the *power spectrum*.

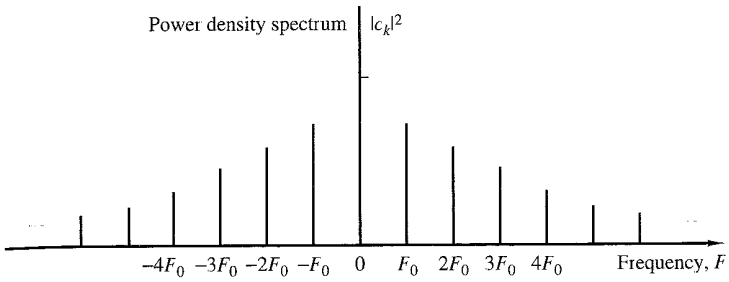



**Figure 4.1.2** Power density spectrum of a continuous-time periodic signal.

As indicated in the preceding section, the Fourier series coefficients $\{c_k\}$ are complex valued, that is, they can be represented as

$$c_k = |c_k|e^{j\theta_k}$$

where

$$\theta_k = \measuredangle c_k$$

Instead of plotting the power density spectrum, we can plot the magnitude voltage spectrum $\{|c_k|\}$ and the phase spectrum $\{\theta_k\}$ as a function of frequency. Clearly, the power spectral density in the periodic signal is simply the square of the magnitude spectrum. The phase information is totally destroyed (or does not appear) in the power spectral density.

If the periodic signal is real valued, the Fourier series coefficients $\{c_k\}$ satisfy the condition

$$c_{-k} = c_k^*$$

Consequently, $|c_k|^2 = |c_k^*|^2$. Hence the power spectrum is a symmetric function of frequency. This condition also means that the magnitude spectrum is symmetric (even function) about the origin and the phase spectrum is an odd function. As a consequence of the symmetry, it is sufficient to specify the spectrum of a real periodic signal for positive frequencies only. Furthermore, the total average power can be expressed as

$$P_x = c_0^2 + 2\sum_{k=1}^{\infty} |c_k|^2 \tag{4.1.15}$$

$$= a_0^2 + \frac{1}{2}\sum_{k=1}^{\infty}(a_k^2 + b_k^2) \tag{4.1.16}$$

which follows directly from the relationships given in Section 4.1.1 among $\{a_k\}$, $\{b_k\}$, and $\{c_k\}$ coefficients in the Fourier series expressions.

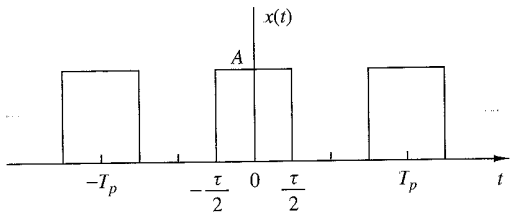



**Figure 4.1.3** Continuous-time periodic train of rectangular pulses.

## EXAMPLE 4.1.1

Determine the Fourier series and the power density spectrum of the rectangular pulse train signal illustrated in Fig 4.1.3.

**Solution.** The signal is periodic with fundamental period $T_p$ and, clearly, satisfies the Dirichlet conditions. Consequently, we can represent the signal in the Fourier series given by (4.1.8) with the Fourier coefficients specified by (4.1.9).

Since $x(t)$ is an even signal [i.e., $x(t) = x(-t)$], it is convenient to select the integration interval from $-T_p/2$ to $T_p/2$. Thus (4.1.9) evaluated for $k = 0$ yields

$$c_0 = \frac{1}{T_p}\int_{-T_p/2}^{T_p/2} x(t)\,dt = \frac{1}{T_p}\int_{-\tau/2}^{\tau/2} A\,dt = \frac{A\tau}{T_p} \tag{4.1.17}$$

The term $c_0$ represents the average value (dc component) of the signal $x(t)$. For $k \neq 0$ we have

$$\begin{aligned} c_k &= \frac{1}{T_p}\int_{-\tau/2}^{\tau/2} Ae^{-j2\pi kF_0t}\,dt = \frac{A}{T_p}\left[\frac{e^{-j2\pi F_0kt}}{-j2\pi kF_0}\right]_{-\tau/2}^{\tau/2} \\ &= \frac{A}{\pi F_0kT_p}\,\frac{e^{j\pi kF_0\tau} - e^{-j\pi kF_0\tau}}{j2} \\ &= \frac{A\tau}{T_p}\,\frac{\sin \pi kF_0\tau}{\pi kF_0\tau}, \qquad k = \pm 1, \pm 2, \ldots \end{aligned} \tag{4.1.18}$$

It is interesting to note that the right-hand side of (4.1.18) has the form $(\sin\phi)/\phi$, where $\phi = \pi kF_0\tau$. In this case $\phi$ takes on discrete values since $F_0$ and $\tau$ are fixed and the index $k$ varies. However, if we plot $(\sin\phi)/\phi$ with $\phi$ as a continuous parameter over the range $-\infty < \phi < \infty$, we obtain the graph shown in Fig 4.1.4. We observe that this function decays to zero as $\phi \to \pm\infty$, has a maximum value of unity at $\phi = 0$, and is zero at multiples of $\pi$ (i.e., at $\phi = m\pi$, $m = \pm 1, \pm 2, \ldots$). It is clear that the Fourier coefficients given by (4.1.18) are the sample values of the $(\sin\phi)/\phi$ function for $\phi = \pi kF_0\tau$ and scaled in amplitude by $A\tau/T_p$.

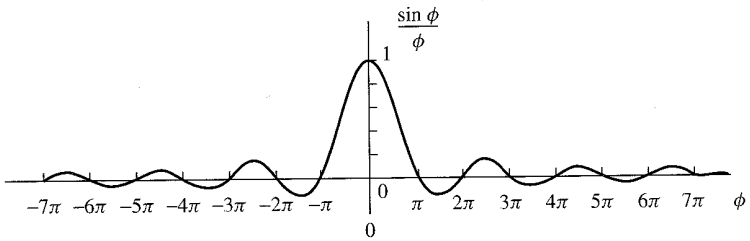



**Figure 4.1.4** The function $(\sin\phi)/\phi$.

Since the periodic function $x(t)$ is even, the Fourier coefficients $c_k$ are real. Consequently, the phase spectrum is either zero, when $c_k$ is positive, or $\pi$ when $c_k$ is negative. Instead of plotting the magnitude and phase spectra separately, we may simply plot $\{c_k\}$ on a single graph, showing both the positive and negative values $c_k$ on the graph. This is commonly done in practice when the Fourier coefficients $\{c_k\}$ are real.

Figure 4.1.5 illustrates the Fourier coefficients of the rectangular pulse train when $T_p$ is fixed and the pulse width $\tau$ is allowed to vary. In this case $T_p = 0.25$ second, so that $F_0 = 1/T_p = 4$ Hz and $\tau = 0.05T_p$, $\tau = 0.1T_p$, and $\tau = 0.2T_p$. We observe that the effect of decreasing $\tau$ while keeping $T_p$ fixed is to spread out the signal power over the frequency range. The spacing between adjacent spectral lines is $F_0 = 4$ Hz, independent of the value of the pulse width $\tau$.

On the other hand, it is also instructive to fix $\tau$ and vary the period $T_p$ when $T_p > \tau$. Figure 4.1.6 illustrates this condition when $T_p = 5\tau$, $T_p = 10\tau$, and $T_p = 20\tau$. In this case, the spacing between adjacent spectral lines decreases as $T_p$ increases. In the limit as $T_p \to \infty$, the Fourier coefficients $c_k$ approach zero due to the factor of $T_p$ in the denominator of (4.1.18). This behavior is consistent with the fact that as $T_p \to \infty$ and $\tau$ remains fixed, the resulting signal is no longer a power signal. Instead, it becomes an energy signal and its average power is zero. The spectra of finite energy signals are described in the next section.

We also note that if $k \neq 0$ and $\sin(\pi k F_0 \tau) = 0$, then $c_k = 0$. The harmonics with zero power occur at frequencies $kF_0$ such that $\pi(kF_0)\tau = m\pi$, $m = \pm 1, \pm 2, \ldots$, or at $kF_0 = m/\tau$. For example, if $F_0 = 4$ Hz and $\tau = 0.2T_p$, it follows that the spectral components at $\pm 20$ Hz, $\pm 40$ Hz, ... have zero power. These frequencies correspond to the Fourier coefficients $c_k$, $k = \pm 5, \pm 10, \pm 15, \ldots$. On the other hand, if $\tau = 0.1T_p$, the spectral components with zero power are $k = \pm 10, \pm 20, \pm 30, \ldots$.

The power density spectrum for the rectangular pulse train is

$$|c_k|^2 = \begin{cases} \left(\dfrac{A\tau}{T_p}\right)^2, & k = 0 \\ \left(\dfrac{A\tau}{T_p}\right)^2 \left(\dfrac{\sin \pi k F_0 \tau}{\pi k F_0 \tau}\right)^2, & k = \pm 1, \pm 2, \ldots \end{cases} \tag{4.1.19}$$

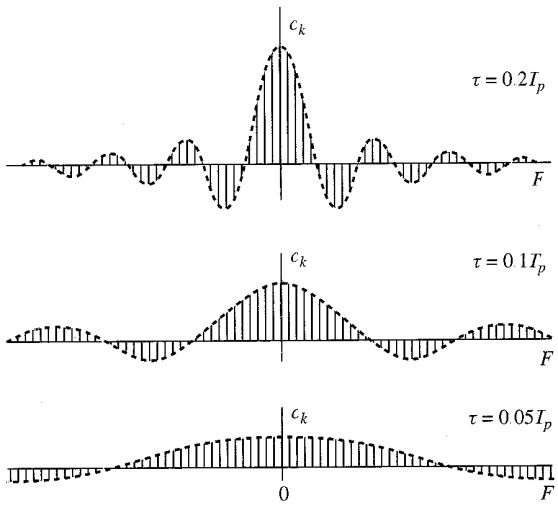



**Figure 4.1.5**
Fourier coefficients of the rectangular pulse train when $T_p$ is fixed and the pulse width $\tau$ varies.

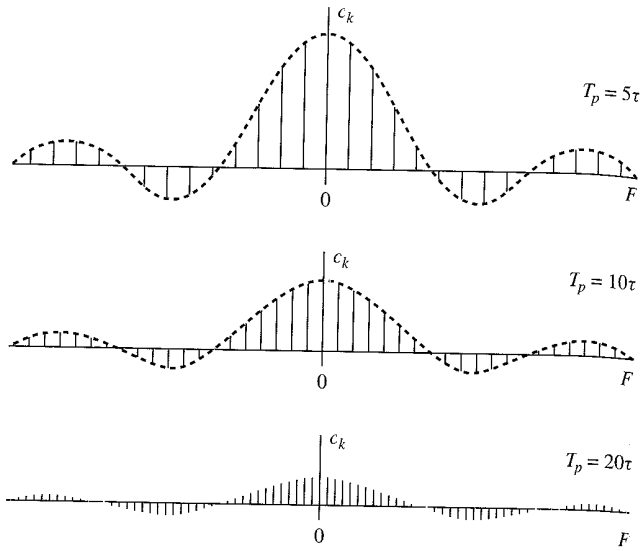



**Figure 4.1.6**
Fourier coefficient of a rectangular pulse train with fixed pulse width $\tau$ and varying period $T_p$.

### 4.1.3 The Fourier Transform for Continuous-Time Aperiodic Signals

In Section 4.1.1 we developed the Fourier series to represent a periodic signal as a linear combination of harmonically related complex exponentials. As a consequence of the periodicity, we saw that these signals possess line spectra with equidistant lines. The line spacing is equal to the fundamental frequency, which in turn is the inverse of the fundamental period of the signal. We can view the fundamental period as providing the number of lines per unit of frequency (line density), as illustrated in Fig 4.1.6.

With this interpretation in mind, it is apparent that if we allow the period to increase without limit, the line spacing tends toward zero. In the limit, when the period becomes infinite, the signal becomes aperiodic and its spectrum becomes continuous. This argument suggests that the spectrum of an aperiodic signal will be the envelope of the line spectrum in the corresponding periodic signal obtained by repeating the aperiodic signal with some period $T_p$.

Let us consider an aperiodic signal $x(t)$ with finite duration as shown in Fig 4.1.7(a). From this aperiodic signal, we can create a periodic signal $x_p(t)$ with period $T_p$, as shown in Fig 4.1.7(b). Clearly, $x_p(t) = x(t)$ in the limit as $T_p \to \infty$, that is,

$$x(t) = \lim_{T_p \to \infty} x_p(t)$$

This interpretation implies that we should be able to obtain the spectrum of $x(t)$ from the spectrum of $x_p(t)$ simply by taking the limit as $T_p \to \infty$.

We begin with the Fourier series representation of $x_p(t)$,

$$x_p(t) = \sum_{k=-\infty}^{\infty} c_k e^{j2\pi k F_0 t}, \qquad F_0 = \frac{1}{T_p} \tag{4.1.20}$$

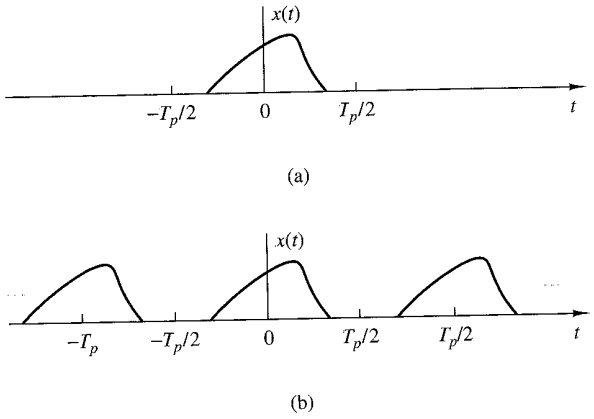



**Figure 4.1.7**
(a) Aperiodic signal $x(t)$ and (b) periodic signal $x_p(t)$ constructed by repeating $x(t)$ with a period $T_p$.

where

$$c_k = \frac{1}{T_p}\int_{-T_p/2}^{T_p/2} x_p(t)e^{-j2\pi kF_0t}\,dt \tag{4.1.21}$$

Since $x_p(t) = x(t)$ for $-T_p/2 \le t \le T_p/2$, (4.1.21) can be expressed as

$$c_k = \frac{1}{T_p}\int_{-T_p/2}^{T_p/2} x(t)e^{-j2\pi kF_0t}\,dt \tag{4.1.22}$$

It is also true that $x(t) = 0$ for $|t| > T_p/2$. Consequently, the limits on the integral in (4.1.22) can be replaced by $-\infty$ and $\infty$. Hence

$$c_k = \frac{1}{T_p}\int_{-\infty}^{\infty} x(t)e^{-j2\pi kF_0t}\,dt \tag{4.1.23}$$

Let us now define a function $X(F)$, called the *Fourier transform* of $x(t)$, as

$$X(F) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi Ft}\,dt \tag{4.1.24}$$

$X(F)$ is a function of the continuous variable $F$. It does not depend on $T_p$ or $F_0$. However, if we compare (4.1.23) and (4.1.24), it is clear that the Fourier coefficients $c_k$ can be expressed in terms of $X(F)$ as

$$c_k = \frac{1}{T_p}X(kF_0)$$

or equivalently,

$$T_pc_k = X(kF_0) = X\left(\frac{k}{T_p}\right) \tag{4.1.25}$$

Thus the Fourier coefficients are samples of $X(F)$ taken at multiples of $F_0$ and scaled by $F_0$ (multiplied by $1/T_p$). Substitution for $c_k$ from (4.1.25) into (4.1.20) yields

$$x_p(t) = \frac{1}{T_p} \sum_{k=-\infty}^{\infty} X\left(\frac{k}{T_p}\right) e^{j2\pi k F_0 t} \tag{4.1.26}$$

We wish to take the limit of (4.1.26) as $T_p$ approaches infinity. First, we define $\Delta F = 1/T_p$. With this substitution, (4.1.26) becomes

$$x_p(t) = \sum_{k=-\infty}^{\infty} X(k\Delta F) e^{j2\pi k \Delta F t} \Delta F \tag{4.1.27}$$

It is clear that in the limit as $T_p$ approaches infinity, $x_p(t)$ reduces to $x(t)$. Also, $\Delta F$ becomes the differential $dF$ and $k\,\Delta F$ becomes the continuous frequency variable $F$. In turn, the summation in (4.1.27) becomes an integral over the frequency variable $F$. Thus

$$\begin{aligned} \lim_{T_p \to \infty} x_p(t) = x(t) &= \lim_{\Delta F \to 0} \sum_{k=-\infty}^{\infty} X(k\Delta F) e^{-j2\pi k \Delta F t} \Delta F \\ x(t) &= \int_{-\infty}^{\infty} X(F) e^{j2\pi F t}\, dF \end{aligned} \tag{4.1.28}$$

This integral relationship yields $x(t)$ when $X(F)$ is known, and it is called the *inverse Fourier transform*.

This concludes our heuristic derivation of the Fourier transform pair given by (4.1.24) and (4.1.28) for an aperiodic signal $x(t)$. Although the derivation is not mathematically rigorous, it led to the desired Fourier transform relationships with relatively simple intuitive arguments. In summary, the frequency analysis of continuous-time aperiodic signals involves the following Fourier transform pair.

Frequency Analysis of Continuous-Time Aperiodic Signals

| | | |
|---|---|---|
| Synthesis equation (inverse transform) | $x(t) = \int_{-\infty}^{\infty} X(F) e^{j2\pi F t}\, dF$ | (4.1.29) |
| Analysis equation (direct transform) | $X(F) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi F t}\, dt$ | (4.1.30) |

It is apparent that the essential difference between the Fourier series and the Fourier transform is that the spectrum in the latter case is continuous and hence the synthesis of an aperiodic signal from its spectrum is accomplished by means of integration instead of summation.

Finally, we wish to indicate that the Fourier transform pair in (4.1.29) and (4.1.30) can be expressed in terms of the radial frequency variable $\Omega = 2\pi F$. Since $dF = d\Omega/2\pi$, (4.1.29) and (4.1.30) become

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\Omega)e^{j\Omega t}\,d\Omega \tag{4.1.31}$$

$$X(\Omega) = \int_{-\infty}^{\infty} x(t)e^{-j\Omega t}\,dt \tag{4.1.32}$$

The set of conditions that guarantee the existence of the Fourier transform is the *Dirichlet conditions*, which may be expressed as:

1. The signal $x(t)$ has a finite number of finite discontinuities.
2. The signal $x(t)$ has a finite number of maxima and minima.
3. The signal $x(t)$ is absolutely integrable, that is,

$$\int_{-\infty}^{\infty} |x(t)|\,dt < \infty \tag{4.1.33}$$

The third condition follows easily from the definition of the Fourier transform, given in (4.1.30). Indeed,

$$|X(F)| = \left|\int_{-\infty}^{\infty} x(t)e^{-j2\pi Ft}\,dt\right| \le \int_{-\infty}^{\infty} |x(t)|\,dt$$

Hence $|X(F)| < \infty$ if (4.1.33) is satisfied.

A weaker condition for the existence of the Fourier transform is that $x(t)$ has finite energy; that is,

$$\int_{-\infty}^{\infty} |x(t)|^2\,dt < \infty \tag{4.1.34}$$

Note that if a signal $x(t)$ is absolutely integrable, it will also have finite energy. That is, if

$$\int_{-\infty}^{\infty} |x(t)|\,dt < \infty$$

then

$$E_x = \int_{-\infty}^{\infty} |x(t)|^2\,dt < \infty \tag{4.1.35}$$

However, the converse is not true. That is, a signal may have finite energy but may not be absolutely integrable. For example, the signal

$$x(t) = \frac{\sin 2\pi F_0 t}{\pi t} \tag{4.1.36}$$

is square integrable but is not absolutely integrable. This signal has the Fourier transform

$$X(F) = \begin{cases} 1, & |F| \le F_0 \\ 0, & |F| > F_0 \end{cases} \tag{4.1.37}$$

Since this signal violates (4.1.33), it is apparent that the Dirichlet conditions are sufficient but not necessary for the existence of the Fourier transform. In any case, nearly all finite energy signals have a Fourier transform, so that we need not worry about the pathological signals, which are seldom encountered in practice.

### 4.1.4 Energy Density Spectrum of Aperiodic Signals

Let $x(t)$ be any finite energy signal with Fourier transform $X(F)$. Its energy is

$$E_x = \int_{-\infty}^{\infty} |x(t)|^2 \, dt$$

which, in turn, may be expressed in terms of $X(F)$ as follows:

$$\begin{aligned} E_x &= \int_{-\infty}^{\infty} x(t)x^*(t) \, dt \\ &= \int_{-\infty}^{\infty} x(t) \, dt \left[ \int_{-\infty}^{\infty} X^*(F) e^{-j2\pi Ft} \, dF \right] \\ &= \int_{-\infty}^{\infty} X^*(F) \, dF \left[ \int_{-\infty}^{\infty} x(t) e^{-j2\pi Ft} \, dt \right] \\ &= \int_{-\infty}^{\infty} |X(F)|^2 \, dF \end{aligned}$$

Therefore, we conclude that

$$E_x = \int_{-\infty}^{\infty} |x(t)|^2 \, dt = \int_{-\infty}^{\infty} |X(F)|^2 \, dF \tag{4.1.38}$$

This is *Parseval's relation* for aperiodic, finite energy signals and expresses the principle of conservation of energy in the time and frequency domains.

The spectrum $X(F)$ of a signal is, in general, complex valued. Consequently, it is usually expressed in polar form as

$$X(F) = |X(F)| e^{j\Theta(F)}$$

where $|X(F)|$ is the magnitude spectrum and $\Theta(F)$ is the phase spectrum,

$$\Theta(F) = \measuredangle X(F)$$

On the other hand, the quantity

$$S_{xx}(F) = |X(F)|^2 \tag{4.1.39}$$

which is the integrand in (4.1.38), represents the distribution of energy in the signal as a function of frequency. Hence $S_{xx}(F)$ is called the *energy density spectrum* of $x(t)$. The integral of $S_{xx}(F)$ over all frequencies gives the total energy in the signal. Viewed in another way, the energy in the signal $x(t)$ over a band of frequencies $F_1 \le F \le F_1 + \Delta F$ is

$$\int_{F_1}^{F_1+\Delta F} S_{xx}(F)\, dF \ge 0$$

which implies that $S_{xx}(f) \ge 0$ for all $F$.

From (4.1.39) we observe that $S_{xx}(F)$ does not contain any phase information [i.e., $S_{xx}(F)$ is purely real and nonnegative]. Since the phase spectrum of $x(t)$ is not contained in $S_{xx}(F)$, it is impossible to reconstruct the signal given $S_{xx}(F)$.

Finally, as in the case of Fourier series, it is easily shown that if the signal $x(t)$ is real, then

$$|X(-F)| = |X(F)| \tag{4.1.40}$$

$$\measuredangle X(-F) = -\measuredangle X(F) \tag{4.1.41}$$

By combining (4.1.40) and (4.1.39), we obtain

$$S_{xx}(-F) = S_{xx}(F) \tag{4.1.42}$$

In other words, the energy density spectrum of a real signal has even symmetry.

### EXAMPLE 4.1.2

Determine the Fourier transform and the energy density spectrum of a rectangular pulse signal defined as

$$x(t) = \begin{cases} A, & |t| \le \tau/2 \\ 0, & |t| > \tau/2 \end{cases} \tag{4.1.43}$$

and illustrated in Fig 4.1.8(a).

**Solution.** Clearly, this signal is aperiodic and satisfies the Dirichlet conditions. Hence its Fourier transform exists. By applying (4.1.30), we find that

$$X(F) = \int_{-\tau/2}^{\tau/2} A e^{-j2\pi F t}\, dt = A\tau \frac{\sin \pi F \tau}{\pi F \tau} \tag{4.1.44}$$

We observe that $X(F)$ is real and hence it can be depicted graphically using only one diagram, as shown in Fig 4.1.8(b). Obviously, $X(F)$ has the shape of the $(\sin\phi)/\phi$ function shown in Fig 4.1.4. Hence the spectrum of the rectangular pulse is the envelope of the line spectrum (Fourier coefficients) of the periodic signal obtained by periodically repeating the pulse with period $T_p$ as in Fig 4.1.3. In other words, the Fourier coefficients $c_k$ in the corresponding periodic signal $x_p(t)$ are simply samples of $X(F)$ at frequencies $kF_0 = k/T_p$. Specifically,

$$c_k = \frac{1}{T_p} X(kF_0) = \frac{1}{T_p} X\left(\frac{k}{T_p}\right) \tag{4.1.45}$$

From (4.1.44) we note that the zero crossings of $X(F)$ occur at multiples of $1/\tau$. Furthermore, the width of the main lobe, which contains most of the signal energy, is equal to $2/\tau$. As the

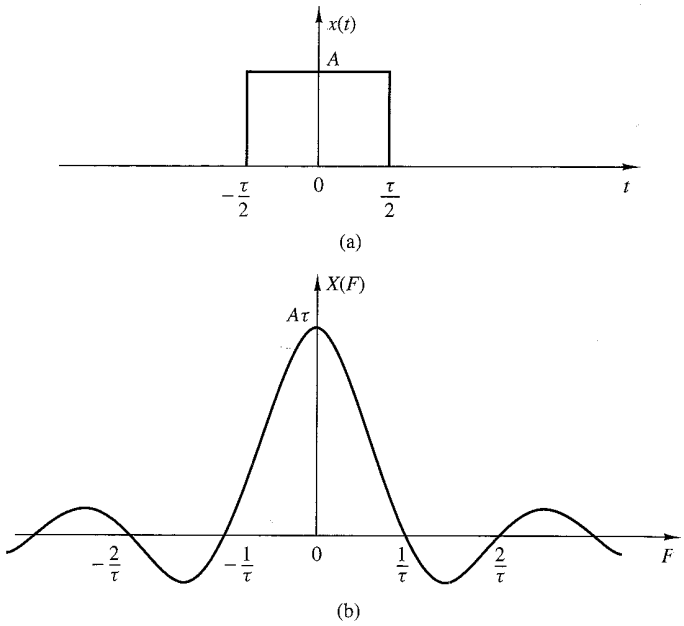



**Figure 4.1.8**
(a) Rectangular pulse and
(b) its Fourier transform.

pulse duration $\tau$ decreases (increases), the main lobe becomes broader (narrower) and more energy is moved to the higher (lower) frequencies, as illustrated in Fig 4.1.9. Thus as the signal

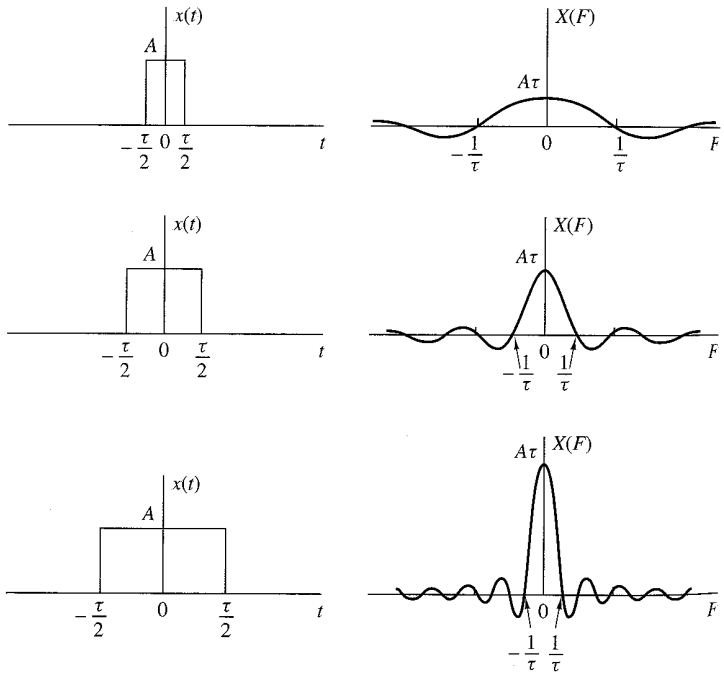



**Figure 4.1.9**
Fourier transform of a rectangular pulse for various width values.

pulse is expanded (compressed) in time, its transform is compressed (expanded) in frequency. This behavior, between the time function and its spectrum, is a type of uncertainty principle that appears in different forms in various branches of science and engineering.

Finally, the energy density spectrum of the rectangular pulse is

$$S_{xx}(F) = (A\tau)^2 \left( \frac{\sin \pi F \tau}{\pi F \tau} \right)^2 \tag{4.1.46}$$

## 4.2 Frequency Analysis of Discrete-Time Signals

In Section 4.1 we developed the Fourier series representation for continuous-time periodic (power) signals and the Fourier transform for finite energy aperiodic signals. In this section we repeat the development for the class of discrete-time signals.

As we have observed from the discussion of Section 4.1, the Fourier series representation of a continuous-time periodic signal can consist of an infinite number of frequency components, where the frequency spacing between two successive harmonically related frequencies is $1/T_p$, and where $T_p$ is the fundamental period. Since the frequency range for continuous-time signals extends from $-\infty$ to $\infty$, it is possible to have signals that contain an infinite number of frequency components. In contrast, the frequency range for discrete-time signals is unique over the interval $(-\pi, \pi)$ or $(0, 2\pi)$. A discrete-time signal of fundamental period $N$ can consist of frequency components separated by $2\pi/N$ radians or $f = 1/N$ cycles. Consequently, the Fourier series representation of the discrete-time periodic signal will contain at most $N$ frequency components. This is the basic difference between the Fourier series representations for continuous-time and discrete-time periodic signals.

### 4.2.1 The Fourier Series for Discrete-Time Periodic Signals

Suppose that we are given a periodic sequence $x(n)$ with period $N$, that is, $x(n) = x(n+N)$ for all $n$. The Fourier series representation for $x(n)$ consists of $N$ harmonically related exponential functions

$$e^{j2\pi kn/N}, \qquad k = 0, 1, \ldots, N-1$$

and is expressed as

$$x(n) = \sum_{k=0}^{N-1} c_k e^{j2\pi kn/N} \tag{4.2.1}$$

where the $\{c_k\}$ are the coefficients in the series representation.

To derive the expression for the Fourier coefficients, we use the following formula:

$$\sum_{n=0}^{N-1} e^{j2\pi kn/N} = \begin{cases} N, & k = 0, \pm N, \pm 2N, \ldots \\ 0, & \text{otherwise} \end{cases} \tag{4.2.2}$$

Note the similarity of (4.2.2) with the continuous-time counterpart in (4.1.3). The proof of (4.2.2) follows immediately from the application of the geometric summation formula

$$\sum_{n=0}^{N-1} a^n = \begin{cases} N, & a = 1 \\ \dfrac{1-a^N}{1-a}, & a \neq 1 \end{cases} \tag{4.2.3}$$

The expression for the Fourier coefficients $c_k$ can be obtained by multiplying both sides of (4.2.1) by the exponential $e^{-j2\pi ln/N}$ and summing the product from $n = 0$ to $n = N-1$. Thus

$$\sum_{n=0}^{N-1} x(n)e^{-j2\pi ln/N} = \sum_{n=0}^{N-1}\sum_{k=0}^{N-1} c_k e^{j2\pi(k-l)n/N} \tag{4.2.4}$$

If we perform the summation over $n$ first, in the right-hand side of (4.2.4), we obtain

$$\sum_{n=0}^{N-1} e^{j2\pi(k-l)n/N} = \begin{cases} N, & k-l = 0, \pm N, \pm 2N, \ldots \\ 0, & \text{otherwise} \end{cases} \tag{4.2.5}$$

where we have made use of (4.2.2). Therefore, the right-hand side of (4.2.4) reduces to $Nc_l$ and hence

$$c_l = \frac{1}{N}\sum_{n=0}^{N-1} x(n)e^{-j2\pi ln/N}, \qquad l = 0, 1, \ldots, N-1 \tag{4.2.6}$$

Thus we have the desired expression for the Fourier coefficients in terms of the signal $x(n)$.

The relationships (4.2.1) and (4.2.6) for the frequency analysis of discrete-time signals are summarized below.

Frequency Analysis of Discrete-Time Periodic Signals

| | | |
|---|---|---|
| Synthesis equation | $x(n) = \sum_{k=0}^{N-1} c_k e^{j2\pi kn/N}$ | (4.2.7) |
| Analysis equation | $c_k = \frac{1}{N}\sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N}$ | (4.2.8) |

Equation (4.2.7) is often called the *discrete-time Fourier series* (DTFS). The Fourier coefficients $\{c_k\}$, $k = 0, 1, \ldots, N-1$ provide the description of $x(n)$ in the frequency domain, in the sense that $c_k$ represents the amplitude and phase associated with the frequency component

$$s_k(n) = e^{j2\pi kn/N} = e^{j\omega_k n}$$

where $\omega_k = 2\pi k/N$.

We recall from Section 1.3.3 that the functions $s_k(n)$ are periodic with period $N$. Hence $s_k(n) = s_k(n+N)$. In view of this periodicity, it follows that the Fourier coefficients $c_k$, when viewed beyond the range $k = 0, 1, \ldots, N-1$, also satisfy a periodicity condition. Indeed, from (4.2.8), which holds for every value of $k$, we have

$$c_{k+N} = \frac{1}{N}\sum_{n=0}^{N-1} x(n)e^{-j2\pi(k+N)n/N} = \frac{1}{N}\sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N} = c_k \tag{4.2.9}$$

Therefore, the Fourier series coefficients $\{c_k\}$ form a periodic sequence when extended outside of the range $k = 0, 1, \ldots, N-1$. Hence

$$c_{k+N} = c_k$$

that is, $\{c_k\}$ is a periodic sequence with fundamental period $N$. *Thus the spectrum of a signal $x(n)$, which is periodic with period $N$, is a periodic sequence with period $N$.* Consequently, any $N$ consecutive samples of the signal or its spectrum provide a complete description of the signal in the time or frequency domains.

Although the Fourier coefficients form a periodic sequence, we will focus our attention on the single period with range $k = 0, 1, \ldots, N-1$. This is convenient, since in the frequency domain, this amounts to covering the fundamental range $0 \le \omega_k = 2\pi k/N < 2\pi$, for $0 \le k \le N-1$. In contrast, the frequency range $-\pi < \omega_k = 2\pi k/N \le \pi$ corresponds to $-N/2 < k \le N/2$, which creates an inconvenience when $N$ is odd. Clearly, if we use a sampling frequency $F_s$, the range $0 \le k \le N-1$ corresponds to the frequency range $0 \le F < F_s$.

### EXAMPLE 4.2.1

Determine the spectra of the signals

**(a)** $x(n) = \cos\sqrt{2}\pi n$

**(b)** $x(n) = \cos \pi n/3$

**(c)** $x(n)$ is periodic with period $N = 4$ and $x(n) = \{\underset{\uparrow}{1}, 1, 0, 0\}$

**Solution.**

**(a)** For $\omega_0 = \sqrt{2}\pi$, we have $f_0 = 1/\sqrt{2}$. Since $f_0$ is not a rational number, the signal is not periodic. Consequently, this signal cannot be expanded in a Fourier series. Nevertheless, the signal does possess a spectrum. Its spectral content consists of the single frequency component at $\omega = \omega_0 = \sqrt{2}\pi$.

**(b)** In this case $f_0 = \frac{1}{6}$ and hence $x(n)$ is periodic with fundamental period $N = 6$. From (4.2.8) we have

$$c_k = \frac{1}{6}\sum_{n=0}^{5} x(n)e^{-j2\pi kn/6}, \qquad k = 0, 1, \ldots, 5$$

However, $x(n)$ can be expressed as

$$x(n) = \cos\frac{2\pi n}{6} = \frac{1}{2}e^{j2\pi n/6} + \frac{1}{2}e^{-j2\pi n/6}$$

which is already in the form of the exponential Fourier series in (4.2.7). In comparing the two exponential terms in $x(n)$ with (4.2.7), it is apparent that $c_1 = \frac{1}{2}$. The second exponential in $x(n)$ corresponds to the term $k = -1$ in (4.2.7). However, this term can also be written as

$$e^{-j2\pi n/6} = e^{j2\pi(5-6)n/6} = e^{j2\pi(5n)/6}$$

which means that $c_{-1} = c_5$. But this is consistent with (4.2.9), and with our previous observation that the Fourier series coefficients form a periodic sequence of period $N$. Consequently, we conclude that

$$c_0 = c_2 = c_3 = c_4 = 0$$

$$c_1 = \frac{1}{2}, \qquad c_5 = \frac{1}{2}$$

**(c)** From (4.2.8), we have

$$c_k = \frac{1}{4}\sum_{n=0}^{3} x(n)e^{-j2\pi kn/4}, \qquad k = 0, 1, 2, 3$$

or

$$c_k = \frac{1}{4}(1 + e^{-j\pi k/2}), \qquad k = 0, 1, 2, 3$$

For $k = 0, 1, 2, 3$ we obtain

$$c_0 = \frac{1}{2}, \qquad c_1 = \frac{1}{4}(1 - j), \qquad c_2 = 0, \qquad c_3 = \frac{1}{4}(1 + j)$$

The magnitude and phase spectra are

$$|c_0| = \frac{1}{2}, \qquad |c_1| = \frac{\sqrt{2}}{4}, \qquad |c_2| = 0, \qquad |c_3| = \frac{\sqrt{2}}{4}$$

$$\measuredangle c_0 = 0, \qquad \measuredangle c_1 = -\frac{\pi}{4}, \qquad \measuredangle c_2 = \text{undefined}, \qquad \measuredangle c_3 = \frac{\pi}{4}$$

Figure 4.2.1 illustrates the spectral content of the signals in (b) and (c).

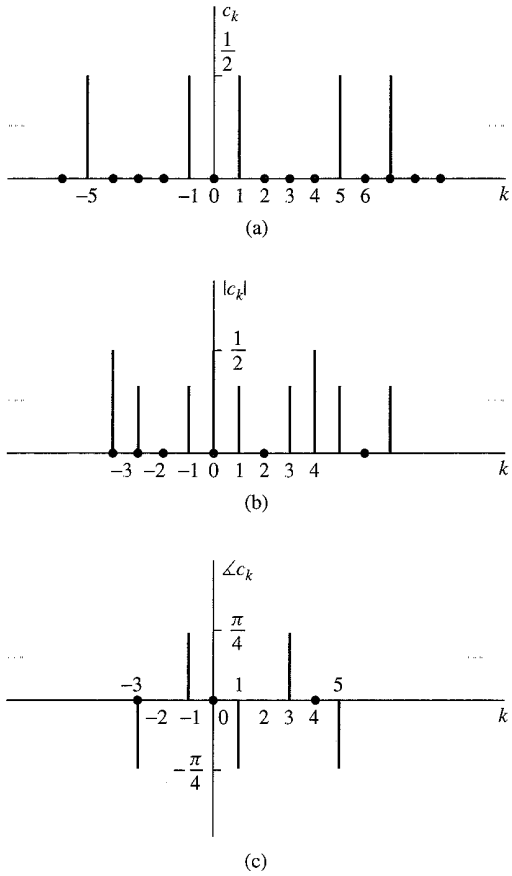



**Figure 4.2.1**
Spectra of the periodic signals discussed in Example 4.2.1 (b) and (c).

### 4.2.2 Power Density Spectrum of Periodic Signals

The average power of a discrete-time periodic signal with period $N$ was defined in (2.1.23) as

$$P_x = \frac{1}{N}\sum_{n=0}^{N-1} |x(n)|^2 \tag{4.2.10}$$

We shall now derive an expression for $P_x$ in terms of the Fourier coefficient $\{c_k\}$.

If we use the relation (4.2.7) in (4.2.10), we have

$$\begin{aligned} P_x &= \frac{1}{N}\sum_{n=0}^{N-1} x(n)x^*(n) \\ &= \frac{1}{N}\sum_{n=0}^{N-1} x(n)\left(\sum_{k=0}^{N-1} c_k^* e^{-j2\pi kn/N}\right) \end{aligned}$$

Now, we can interchange the order of the two summations and make use of (4.2.8), obtaining

$$\begin{aligned} P_x &= \sum_{k=0}^{N-1} c_k^* \left[ \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-j2\pi kn/N} \right] \\ &= \sum_{k=0}^{N-1} |c_k|^2 = \frac{1}{N} \sum_{n=0}^{N-1} |x(n)|^2 \end{aligned} \tag{4.2.11}$$

which is the desired expression for the average power in the periodic signal. In other words, the average power in the signal is the sum of the powers of the individual frequency components. We view (4.2.11) as a Parseval's relation for discrete-time periodic signals. The sequence $|c_k|^2$ for $k = 0, 1, \ldots, N-1$ is the distribution of power as a function of frequency and is called the *power density spectrum* of the periodic signal.

If we are interested in the energy of the sequence $x(n)$ over a single period, (4.2.11) implies that

$$E_N = \sum_{n=0}^{N-1} |x(n)|^2 = N \sum_{k=0}^{N-1} |c_k|^2 \tag{4.2.12}$$

which is consistent with our previous results for continuous-time periodic signals. If the signal $x(n)$ is real [i.e., $x^*(n) = x(n)$], then, proceeding as in Section 4.2.1, we can easily show that

$$c_k^* = c_{-k} \tag{4.2.13}$$

or equivalently,

$$|c_{-k}| = |c_k| \qquad \text{(even symmetry)} \tag{4.2.14}$$

$$-\measuredangle c_{-k} = \measuredangle c_k \qquad \text{(odd symmetry)} \tag{4.2.15}$$

These symmetry properties for the magnitude and phase spectra of a periodic signal, in conjunction with the periodicity property, have very important implications on the frequency range of discrete-time signals.

Indeed, by combining (4.2.9) with (4.2.14) and (4.2.15), we obtain

$$|c_k| = |c_{N-k}| \tag{4.2.16}$$

and

$$\measuredangle c_k = -\measuredangle c_{N-k} \tag{4.2.17}$$

More specifically, we have

$$\begin{array}{lll} |c_0| = |c_N|, & \measuredangle c_0 = -\measuredangle c_N = 0 & \\ |c_1| = |c_{N-1}|, & \measuredangle c_1 = -\measuredangle c_{N-1} & \\ |c_{N/2}| = |c_{N/2}|, & \measuredangle c_{N/2} = 0 & \text{if } N \text{ is even} \\ |c_{(N-1)/2}| = |c_{(N+1)/2}|, & \measuredangle c_{(N-1)/2} = -\measuredangle c_{(N+1)/2} & \text{if } N \text{ is odd} \end{array} \tag{4.2.18}$$

Thus, for a real signal, the spectrum $c_k$, $k = 0, 1, \ldots, N/2$ for $N$ even, or $k = 0, 1, \ldots, (N-1)/2$ for $N$ odd, completely specifies the signal in the frequency domain. Clearly, this is consistent with the fact that the highest relative frequency that can be represented by a discrete-time signal is equal to $\pi$. Indeed, if $0 \le \omega_k = 2\pi k/N \le \pi$, then $0 \le k \le N/2$.

By making use of these symmetry properties of the Fourier series coefficients of a real signal, the Fourier series in (4.2.7) can also be expressed in the alternative forms

$$x(n) = c_0 + 2\sum_{k=1}^{L} |c_k| \cos\left(\frac{2\pi}{N}kn + \theta_k\right) \tag{4.2.19}$$

$$= a_0 + \sum_{k=1}^{L}\left(a_k \cos\frac{2\pi}{N}kn - b_k \sin\frac{2\pi}{N}kn\right) \tag{4.2.20}$$

where $a_0 = c_0$, $a_k = 2|c_k|\cos\theta_k$, $b_k = 2|c_k|\sin\theta_k$, and $L = N/2$ if $N$ is even and $L = (N-1)/2$ if $N$ is odd.

Finally, we note that as in the case of continuous-time signals, the power density spectrum $|c_k|^2$ does not contain any phase information. Furthermore, the spectrum is discrete and periodic with a fundamental period equal to that of the signal itself.

### EXAMPLE 4.2.2 Periodic "Square-Wave" Signal

Determine the Fourier series coefficients and the power density spectrum of the periodic signal shown in Fig 4.2.2.

**Solution.** By applying the analysis equation (4.2.8) to the signal shown in Fig 4.2.2, we obtain

$$c_k = \frac{1}{N}\sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N} = \frac{1}{N}\sum_{n=0}^{L-1} Ae^{-j2\pi kn/N}, \qquad k = 0, 1, \ldots, N-1$$

which is a geometric summation. Now we can use (4.2.3) to simplify the summation above. Thus we obtain

$$c_k = \frac{A}{N}\sum_{n=0}^{L-1}(e^{-j2\pi k/N})^n = \begin{cases} \dfrac{AL}{N}, & k = 0 \\ \dfrac{A}{N}\dfrac{1-e^{-j2\pi kL/N}}{1-e^{-j2\pi k/N}}, & k = 1, 2, \ldots, N-1 \end{cases}$$

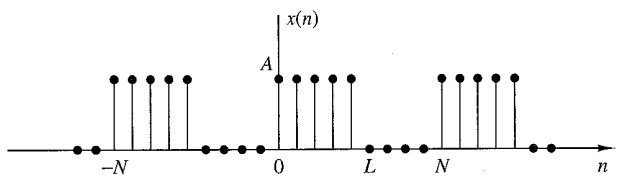



**Figure 4.2.2**
Discrete-time periodic square-wave signal.

The last expression can be simplified further if we note that

$$\frac{1-e^{-j2\pi kL/N}}{1-e^{-j2\pi k/N}} = \frac{e^{-j\pi kL/N}}{e^{-j\pi k/N}} \frac{e^{j\pi kL/N}-e^{-j\pi kL/N}}{e^{j\pi k/N}-e^{-j\pi k/N}}$$

$$= e^{-j\pi k(L-1)/N} \frac{\sin(\pi kL/N)}{\sin(\pi k/N)}$$

Therefore,

$$c_k = \begin{cases} \dfrac{AL}{N}, & k = 0, +N, \pm 2N, \ldots \\ \dfrac{A}{N} e^{-j\pi k(L-1)/N} \dfrac{\sin(\pi kL/N)}{\sin(\pi k/N)}, & \text{otherwise} \end{cases} \tag{4.2.21}$$

The power density spectrum of this periodic signal is

$$|c_k|^2 = \begin{cases} \left(\dfrac{AL}{N}\right)^2, & k = 0, +N, \pm 2N, \ldots \\ \left(\dfrac{A}{N}\right)^2 \left(\dfrac{\sin \pi kL/N}{\sin \pi k/N}\right)^2, & \text{otherwise} \end{cases} \tag{4.2.22}$$

Figure 4.2.3 illustrates the plots of $|c_k|^2$ for $L = 2$, $N = 10$ and 40, and $A = 1$.

### 4.2.3 The Fourier Transform of Discrete-Time Aperiodic Signals

Just as in the case of continuous-time aperiodic energy signals, the frequency analysis of discrete-time aperiodic finite-energy signals involves a Fourier transform of the time-domain signal. Consequently, the development in this section parallels, to a large extent, that given in Section 4.1.3.

The Fourier transform of a finite-energy discrete-time signal $x(n)$ is defined as

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n) e^{-j\omega n} \tag{4.2.23}$$

Physically, $X(\omega)$ represents the frequency content of the signal $x(n)$. In other words, $X(\omega)$ is a decomposition of $x(n)$ into its frequency components.

We observe two basic differences between the Fourier transform of a discrete-time finite-energy signal and the Fourier transform of a finite-energy analog signal. First, for continuous-time signals, the Fourier transform, and hence the spectrum of the signal, have a frequency range of $(-\infty, \infty)$. In contrast, the frequency range for a discrete-time signal is unique over the frequency interval of $(-\pi, \pi)$ or, equivalently, $(0, 2\pi)$. This property is reflected in the Fourier transform of the signal. Indeed, $X(\omega)$

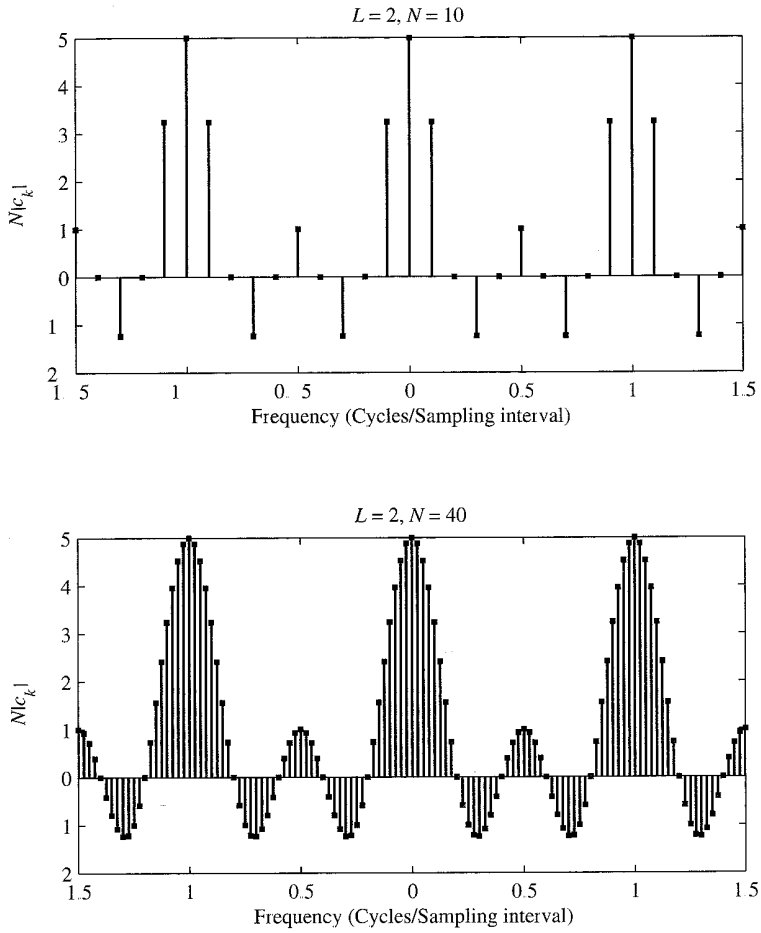



**Figure 4.2.3** Plot of the power density spectrum given by (4.2.22).

is periodic with period $2\pi$, that is,

$$\begin{aligned} X(\omega + 2\pi k) &= \sum_{n=-\infty}^{\infty} x(n)e^{-j(\omega+2\pi k)n} \\ &= \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n}e^{-j2\pi kn} \\ &= \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} = X(\omega) \end{aligned} \tag{4.2.24}$$

Hence $X(\omega)$ is periodic with period $2\pi$. But this property is just a consequence of the fact that the frequency range for any discrete-time signal is limited to $(-\pi, \pi)$ or

$(0, 2\pi)$, and any frequency outside this interval is equivalent to a frequency within the interval.

The second basic difference is also a consequence of the discrete-time nature of the signal. Since the signal is discrete in time, the Fourier transform of the signal involves a summation of terms instead of an integral, as in the case of continuous-time signals.

Since $X(\omega)$ is a periodic function of the frequency variable $\omega$, it has a Fourier series expansion, provided that the conditions for the existence of the Fourier series, described previously, are satisfied. In fact, from the definition of the Fourier transform $X(\omega)$ of the sequence $x(n)$, given by (4.2.23), we observe that $X(\omega)$ has the form of a Fourier series. The Fourier coefficients in this series expansion are the values of the sequence $x(n)$.

To demonstrate this point, let us evaluate the sequence $x(n)$ from $X(\omega)$. First, we multiply both sides (4.2.23) by $e^{j\omega m}$ and integrate over the interval $(-\pi, \pi)$. Thus we have

$$\int_{-\pi}^{\pi} X(\omega)e^{j\omega m}\, d\omega = \int_{-\pi}^{\pi} \left[ \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} \right] e^{j\omega m}\, d\omega \tag{4.2.25}$$

The integral on the right-hand side of (4.2.25) can be evaluated if we can interchange the order of summation and integration. This interchange can be made if the series

$$X_N(\omega) = \sum_{n=-N}^{N} x(n)e^{-j\omega n}$$

converges uniformly to $X(\omega)$ as $N \to \infty$. Uniform convergence means that, for every $\omega$, $X_N(\omega) \to X(\omega)$, as $N \to \infty$. The convergence of the Fourier transform is discussed in more detail in the following section. For the moment, let us assume that the series converges uniformly, so that we can interchange the order of summation and integration in (4.2.25). Then

$$\int_{-\pi}^{\pi} e^{j\omega(m-n)}\, d\omega = \begin{cases} 2\pi, & m = n \\ 0, & m \neq n \end{cases}$$

Consequently,

$$\sum_{n=-\infty}^{\infty} x(n) \int_{-\pi}^{\pi} e^{j\omega(m-n)}\, d\omega = \begin{cases} 2\pi x(m), & m = n \\ 0, & m \neq n \end{cases} \tag{4.2.26}$$

By combining (4.2.25) and (4.2.26), we obtain the desired result that

$$x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega)e^{j\omega n}\, d\omega \tag{4.2.27}$$

If we compare the integral in (4.2.27) with (4.1.9), we note that this is just the expression for the Fourier series coefficient for a function that is periodic with period

$2\pi$. The only difference between (4.1.9) and (4.2.27) is the sign on the exponent in the integrand, which is a consequence of our definition of the Fourier transform as given by (4.2.23). Therefore, the Fourier transform of the sequence $x(n)$, defined by (4.2.23), has the form of a Fourier series expansion.

In summary, the *Fourier transform pair for discrete-time signals* is as follows.

Frequency Analysis of Discrete-Time Aperiodic Signals

| | | |
|---|---|---|
| Synthesis equation (inverse transform) | $x(n) = \frac{1}{2\pi} \int_{2\pi} X(\omega) e^{j\omega n}\, d\omega$ | (4.2.28) |
| Analysis equation (direct transform) | $X(\omega) = \sum_{n=-\infty}^{\infty} x(n) e^{-j\omega n}$ | (4.2.29) |

### 4.2.4 Convergence of the Fourier Transform

In the derivation of the inverse transform given by (4.2.28), we assumed that the series

$$X_N(\omega) = \sum_{n=-N}^{N} x(n) e^{-j\omega n} \tag{4.2.30}$$

converges uniformly to $X(\omega)$, given in the integral of (4.2.25), as $N \to \infty$. By uniform convergence we mean that for each $\omega$,

$$\lim_{N\to\infty} \{\sup_\omega |X(\omega) - X_N(\omega)|\} = 0 \tag{4.2.31}$$

Uniform convergence is guaranteed if $x(n)$ is absolutely summable. Indeed, if

$$\sum_{n=-\infty}^{\infty} |x(n)| < \infty \tag{4.2.32}$$

then

$$|X(\omega)| = \left| \sum_{n=-\infty}^{\infty} x(n) e^{-j\omega n} \right| \le \sum_{n=-\infty}^{\infty} |x(n)| < \infty$$

Hence (4.2.32) is a sufficient condition for the existence of the discrete-time Fourier transform. We note that this is the discrete-time counterpart of the third Dirichlet condition for the Fourier transform of continuous-time signals. The first two conditions do not apply due to the discrete-time nature of $\{x(n)\}$.

Some sequences are not absolutely summable, but they are square summable. That is, they have finite energy

$$E_x = \sum_{n=-\infty}^{\infty} |x(n)|^2 < \infty \tag{4.2.33}$$

which is a weaker condition than (4.2.32). We would like to define the Fourier transform of finite-energy sequences, but we must relax the condition of uniform convergence. For such sequences we can impose a mean-square convergence condition:

$$\lim_{N\to\infty} \int_{-\pi}^{\pi} |X(\omega) - X_N(\omega)|^2 \, d\omega = 0 \tag{4.2.34}$$

Thus the energy in the error $X(\omega) - X_N(\omega)$ tends toward zero, but the error $|X(\omega) - X_N(\omega)|$ does not necessarily tend to zero. In this way we can include finite-energy signals in the class of signals for which the Fourier transform exists.

Let us consider an example from the class of finite-energy signals. Suppose that

$$X(\omega) = \begin{cases} 1, & |\omega| \le \omega_c \\ 0, & \omega_c < |\omega| \le \pi \end{cases} \tag{4.2.35}$$

The reader should remember that $X(\omega)$ is periodic with period $2\pi$. Hence (4.2.35) represents only one period of $X(\omega)$. The inverse transform of $X(\omega)$ results in the sequence

$$\begin{aligned} x(n) &= \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega) e^{j\omega n} \, d\omega \\ &= \frac{1}{2\pi} \int_{-\omega_c}^{\omega_c} e^{j\omega n} \, d\omega = \frac{\sin \omega_c n}{\pi n}, \qquad n \neq 0 \end{aligned}$$

For $n = 0$, we have

$$x(0) = \frac{1}{2\pi} \int^{\omega_c} -\omega_c \, d\omega = \frac{\omega_c}{\pi}$$

Hence

$$x(n) = \begin{cases} \dfrac{\omega_c}{\pi}, & n = 0 \\ \dfrac{\omega_c}{\pi} \dfrac{\sin \omega_c n}{\omega_c n}, & n \neq 0 \end{cases} \tag{4.2.36}$$

This transform pair is illustrated in Fig 4.2.4.

Sometimes, the sequence $\{x(n)\}$ in (4.2.36) is expressed as

$$x(n) = \frac{\sin \omega_c n}{\pi n}, \qquad -\infty < n < \infty \tag{4.2.37}$$

with the understanding that at $n = 0$, $x(n) = \omega_c/\pi$. We should emphasize, however, that $(\sin \omega_c n)/\pi n$ is not a continuous function, and hence L'Hospital's rule cannot be used to determine $x(0)$.

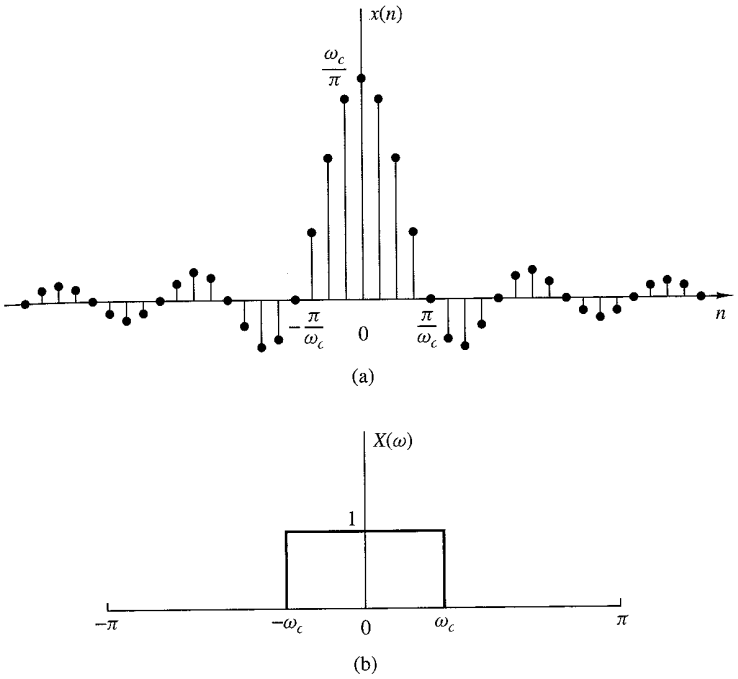



**Figure 4.2.4** Fourier transform pair in (4.2.35) and (4.2.36).

Now let us consider the determination of the Fourier transform of the sequence given by (4.2.37). The sequence $\{x(n)\}$ is not absolutely summable. Hence the infinite series

$$\sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} = \sum_{n=-\infty}^{\infty} \frac{\sin \omega_c n}{\pi n} e^{-j\omega n} \tag{4.2.38}$$

does not converge uniformly for all $\omega$. However, the sequence $\{x(n)\}$ has a finite energy $E_x = \omega_c/\pi$ as will be shown in Section 4.3. Hence the sum in (4.2.38) is guaranteed to converge to the $X(\omega)$ given by (4.2.35) in the mean-square sense.

To elaborate on this point, let us consider the finite sum

$$X_N(\omega) = \sum_{n=-N}^{N} \frac{\sin \omega_c n}{\pi n} e^{-j\omega n} \tag{4.2.39}$$

Figure 4.2.5 shows the function $X_N(\omega)$ for several values of $N$. We note that there is a significant oscillatory overshoot at $\omega = \omega_c$, independent of the value of $N$. As $N$ increases, the oscillations become more rapid, but the size of the ripple remains the same. One can show that as $N \to \infty$, the oscillations converge to the point of the discontinuity at $\omega = \omega_c$, but their amplitude does not go to zero. However, (4.2.34) is satisfied, and therefore $X_N(\omega)$ converges to $X(\omega)$ in the mean-square sense.

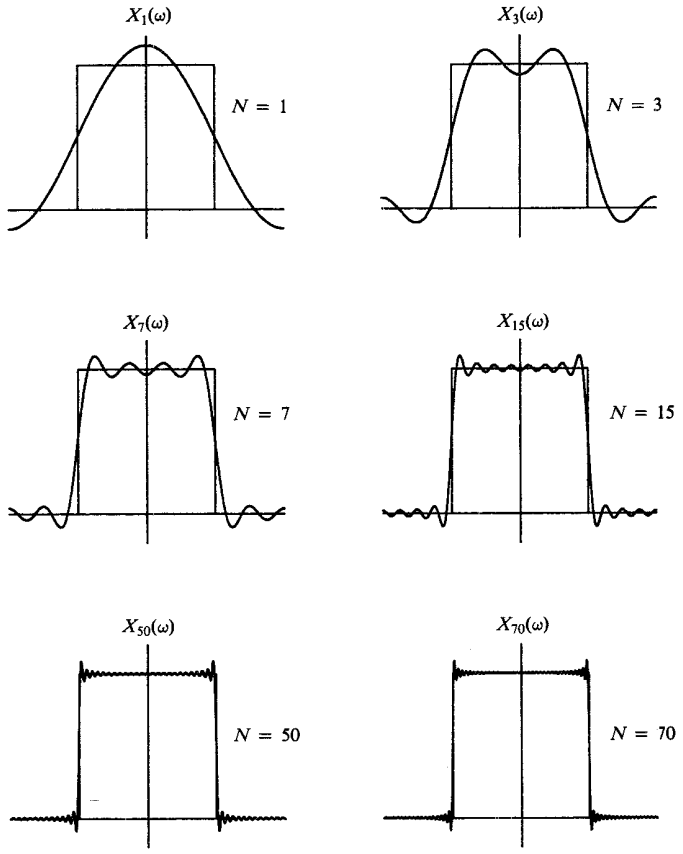



**Figure 4.2.5** Illustration of convergence of the Fourier transform and the Gibbs phenomenon at the point of discontinuity

The oscillatory behavior of the approximation $X_N(\omega)$ to the function $X(\omega)$ at a point of discontinuity of $X(\omega)$ is called the *Gibbs phenomenon*. A similar effect is observed in the truncation of the Fourier series of a continuous-time periodic signal, given by the synthesis equation (4.1.8). For example, the truncation of the Fourier series for the periodic square-wave signal in Example 4.1.1 gives rise to the same oscillatory behavior in the finite-sum approximation of $x(t)$. The Gibbs phenomenon will be encountered again in the design of practical, discrete-time FIR systems considered in Chapter 10.

### 4.2.5 Energy Density Spectrum of Aperiodic Signals

Recall that the energy of a discrete-time signal $x(n)$ is defined as

$$E_x = \sum_{n=-\infty}^{\infty} |x(n)|^2 \tag{4.2.40}$$

Let us now express the energy $E_x$ in terms of the spectral characteristic $X(\omega)$. First we have

$$E_x = \sum_{n=-\infty}^{\infty} x^*(n)x(n) = \sum_{n=-\infty}^{\infty} x(n)\left[\frac{1}{2\pi}\int_{-\pi}^{\pi} X^*(\omega)e^{-j\omega n}\,d\omega\right]$$

If we interchange the order of integration and summation in the equation above, we obtain

$$\begin{aligned} E_x &= \frac{1}{2\pi}\int_{-\pi}^{\pi} X^*(\omega)\left[\sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n}\right]d\omega \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} |X(\omega)|^2\,d\omega \end{aligned}$$

Therefore, the energy relation between $x(n)$ and $X(\omega)$ is

$$E_x = \sum_{n=-\infty}^{\infty} |x(n)|^2 = \frac{1}{2\pi}\int_{-\pi}^{\pi} |X(\omega)|^2\,d\omega \tag{4.2.41}$$

This is Parseval's relation for discrete-time aperiodic signals with finite energy.

The spectrum $X(\omega)$ is, in general, a complex-valued function of frequency. It may be expressed as

$$X(\omega) = |X(\omega)|e^{j\Theta(\omega)} \tag{4.2.42}$$

where

$$\Theta(\omega) = \measuredangle X(\omega)$$

is the phase spectrum and $|X(\omega)|$ is the magnitude spectrum.

As in the case of continuous-time signals, the quantity

$$S_{xx}(\omega) = |X(\omega)|^2 \tag{4.2.43}$$

represents the distribution of energy as a function of frequency, and it is called the *energy density spectrum* of $x(n)$. Clearly, $S_{xx}(\omega)$ does not contain any phase information.

Suppose now that the signal $x(n)$ is real. Then it easily follows that

$$X^*(\omega) = X(-\omega) \tag{4.2.44}$$

or equivalently,

$$|X(-\omega)| = |X(\omega)|, \qquad \text{(even symmetry)} \tag{4.2.45}$$

and

$$\measuredangle X(-\omega) = -\measuredangle X(\omega), \qquad \text{(odd symmetry)} \tag{4.2.46}$$

From (4.2.43) it also follows that

$$S_{xx}(-\omega) = S_{xx}(\omega), \qquad \text{(even symmetry)} \tag{4.2.47}$$

From these symmetry properties we conclude that the frequency range of real discrete-time signals can be limited further to the range $0 \le \omega \le \pi$ (i.e., one-half of the period). Indeed, if we know $X(\omega)$ in the range $0 \le \omega \le \pi$, we can determine it for the range $-\pi \le \omega < 0$ using the symmetry properties given above. As we have already observed, similar results hold for discrete-time periodic signals. Therefore, the frequency-domain description of a real discrete-time signal is completely specified by its spectrum in the frequency range $0 \le \omega \le \pi$.

Usually, we work with the fundamental interval $0 \le \omega \le \pi$ or $0 \le F \le F_s/2$, expressed in hertz. We sketch more than half a period only when required by the specific application.

### EXAMPLE 4.2.3

Determine and sketch the energy density spectrum $S_{xx}(\omega)$ of the signal

$$x(n) = a^n u(n), \qquad -1 < a < 1$$

**Solution.** Since $|a| < 1$, the sequence $x(n)$ is absolutely summable, as can be verified by applying the geometric summation formula,

$$\sum_{n=-\infty}^{\infty} |x(n)| = \sum_{n=0}^{\infty} |a|^n = \frac{1}{1-|a|} < \infty$$

Hence the Fourier transform of $x(n)$ exists and is obtained by applying (4.2.29). Thus

$$X(\omega) = \sum_{n=0}^{\infty} a^n e^{-j\omega n} = \sum_{n=0}^{\infty} (ae^{-j\omega})^n$$

Since $|ae^{-j\omega}| = |a| < 1$, use of the geometric summation formula again yields

$$X(\omega) = \frac{1}{1 - ae^{-j\omega}}$$

The energy density spectrum is given by

$$S_{xx}(\omega) = |X(\omega)|2 = X(\omega)X(\omega) = \frac{1}{(1-ae^{-j\omega})(1-ae^{j\omega})}$$

or, equivalently, as

$$S_{xx}(\omega) = \frac{1}{1 - 2a\cos\omega + a^2}$$

Note that $S_{xx}(-\omega) = S_{xx}(\omega)$ in accordance with (4.2.47).

Figure 4.2.6 shows the signal $x(n)$ and its corresponding spectrum for $a = 0.5$ and $a = -0.5$. Note that for $a = -0.5$ the signal has more rapid variations and as a result its spectrum has stronger high frequencies.

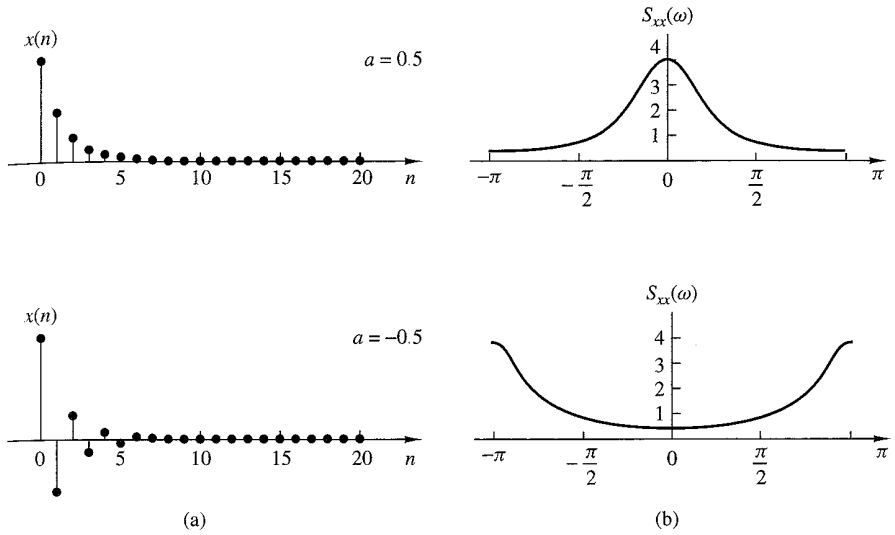



**Figure 4.2.6** (a) Sequence $x(n) = (\frac{1}{2})^n u(n)$ and $x(n) = (-\frac{1}{2})^n u(n)$; (b) their energy density spectra.

---

EXAMPLE 4.2.4

Determine the Fourier transform and the energy density spectrum of the sequence

$$x(n) = \begin{cases} A, & 0 \le n \le L-1 \\ 0, & \text{otherwise} \end{cases} \tag{4.2.48}$$

which is illustrated in Fig 4.2.7.

**Solution.** Before computing the Fourier transform, we observe that

$$\sum_{n=-\infty}^{\infty} |x(n)| = \sum_{n=0}^{L-1} |A| = L|A| < \infty$$

Hence $x(n)$ is absolutely summable and its Fourier transform exists. Furthermore, we note that $x(n)$ is a finite-energy signal with $E_x = |A|^2 L$.

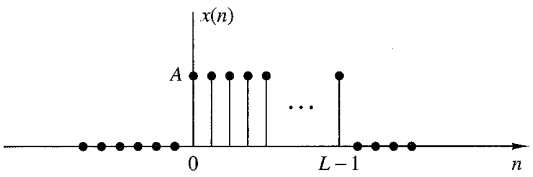



**Figure 4.2.7** Discrete-time rectangular pulse.

The Fourier transform of this signal is

$$\begin{aligned} X(\omega) &= \sum_{n=0}^{L-1} A e^{-j\omega n} \\ &= A \frac{1 - e^{-j\omega L}}{1 - e^{-j\omega}} \\ &= A e^{-j(\omega/2)(L-1)} \frac{\sin(\omega L/2)}{\sin(\omega/2)} \end{aligned} \tag{4.2.49}$$

For $\omega = 0$ the transform in (4.2.49) yields $X(0) = AL$, which is easily established by setting $\omega = 0$ in the defining equation for $X(\omega)$, or by using L'Hospital's rule in (4.2.49) to resolve the indeterminate form when $\omega = 0$.

The magnitude and phase spectra of $x(n)$ are

$$|X(\omega)| = \begin{cases} |A|L, & \omega = 0 \\ |A| \left| \dfrac{\sin(\omega L/2)}{\sin(\omega/2)} \right|, & \text{otherwise} \end{cases} \tag{4.2.50}$$

and

$$\measuredangle X(\omega) = \measuredangle A - \frac{\omega}{2}(L-1) + \measuredangle \frac{\sin(\omega L/2)}{\sin(\omega/2)} \tag{4.2.51}$$

where we should remember that the phase of a real quantity is zero if the quantity is positive and $\pi$ if it is negative.

The spectra $|X(\omega)|$ and $\measuredangle X(\omega)$ are shown in Fig 4.2.8 for the case $A = 1$ and $L = 5$. The energy density spectrum is simply the square of the expression given in (4.2.50).

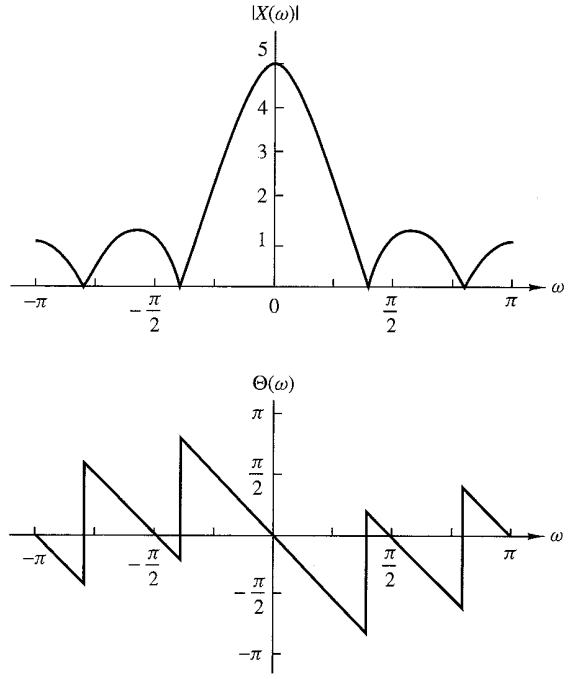



**Figure 4.2.8** Magnitude and phase of Fourier transform of the discrete-time rectangular pulse in Fig 4.2.7.

There is an interesting relationship between the Fourier transform of the constant amplitude pulse in Example 4.2.4 and the periodic rectangular wave considered in Example 4.2.2. If we evaluate the Fourier transform as given in (4.2.49) at a set of equally spaced (harmonically related) frequencies

$$\omega_k = \frac{2\pi}{N}k, \qquad k = 0, 1, \ldots, N-1$$

we obtain

$$X\left(\frac{2\pi}{N}k\right) = Ae^{-j(\pi/N)k(L-1)}\frac{\sin[(\pi/N)kL]}{\sin[(\pi/N)k]} \tag{4.2.52}$$

If we compare this result with the expression for the Fourier series coefficients given in (4.2.21) for the periodic rectangular wave, we find that

$$X\left(\frac{2\pi}{N}k\right) = Nc_k, \qquad k = 0, 1, \ldots, N-1 \tag{4.2.53}$$

To elaborate, we have established that the Fourier transform of the rectangular pulse, which is identical with a single period of the periodic rectangular pulse train, evaluated at the frequencies $\omega = 2\pi k/N$, $k = 0, 1, \ldots, N-1$, which are identical to the harmonically related frequency components used in the Fourier series representation of the periodic signal, is simply a multiple of the Fourier coefficients $\{c_k\}$ at the corresponding frequencies.

The relationship given in (4.2.53) for the Fourier transform of the rectangular pulse evaluated at $\omega = 2\pi k/N$, $k = 0, 1, \ldots, N-1$, and the Fourier coefficients of the corresponding periodic signal, is not only true for these two signals but, in fact, holds in general. This relationship is developed further in Chapter 7.

### 4.2.6 Relationship of the Fourier Transform to the $z$-Transform

The $z$-transform of a sequence $x(n)$ is defined as

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n}, \qquad \text{ROC: } r_2 < |z| < r_1 \tag{4.2.54}$$

where $r_2 < |z| < r_1$ is the region of convergence of $X(z)$. Let us express the complex variable $z$ in polar form as

$$z = re^{j\omega} \tag{4.2.55}$$

where $r = |z|$ and $\omega = \measuredangle z$. Then, within the region of convergence of $X(z)$, we can substitute $z = re^{j\omega}$ into (4.2.54). This yields

$$X(z)|_{z=re^{j\omega}} = \sum_{n=-\infty}^{\infty} [x(n)r^{-n}]e^{-j\omega n} \tag{4.2.56}$$

From the relationship in (4.2.56) we note that $X(z)$ can be interpreted as the Fourier transform of the signal sequence $x(n)r^{-n}$. The weighting factor $r^{-n}$ is growing with $n$ if $r < 1$ and decaying if $r > 1$. Alternatively, if $X(z)$ converges for $|z| = 1$, then

$$X(z)|_{z=e^{j\omega}} \equiv X(\omega) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} \tag{4.2.57}$$

Therefore, the Fourier transform can be viewed as the $z$-transform of the sequence evaluated on the unit circle. If $X(z)$ does not converge in the region $|z| = 1$ [i.e., if the unit circle is not contained in the region of convergence of $X(z)$], the Fourier transform $X(\omega)$ does not exist. Figure 4.2.9 illustrates the relationship between $X(z)$ and $X(\omega)$ for the rectangular sequence in Example 4.2.4, where $A = 1$ and $L = 10$.

We should note that the existence of the $z$-transform requires that the sequence $\{x(n)r^{-n}\}$ be absolutely summable for some value of $r$, that is,

$$\sum_{n=-\infty}^{\infty} |x(n)r^{-n}| < \infty \tag{4.2.58}$$

Hence if (4.2.58) converges only for values of $r > r_0 > 1$, the $z$-transform exists, but the Fourier transform does not exist. This is the case, for example, for causal sequences of the form $x(n) = a^n u(n)$, where $|a| > 1$.

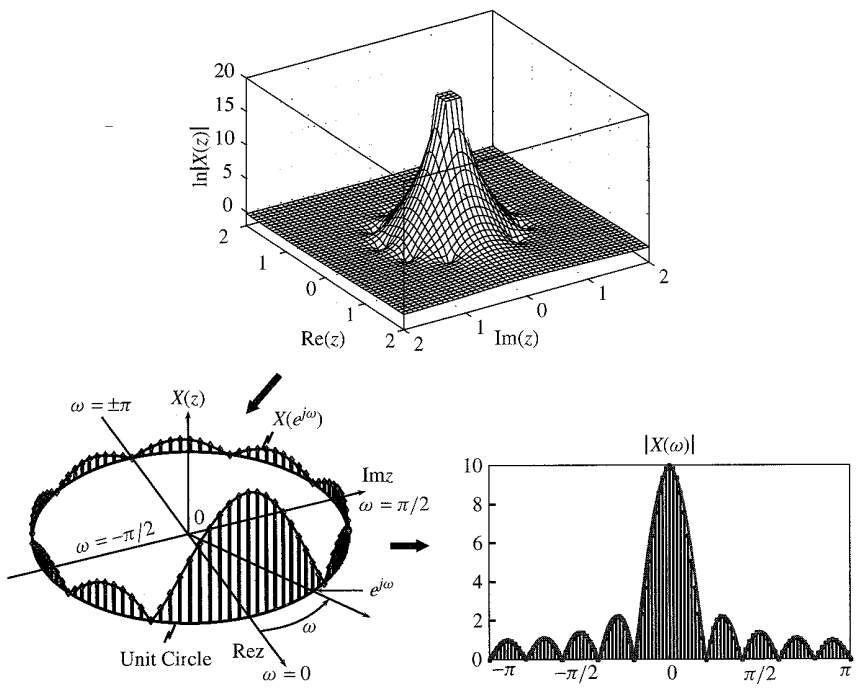



**Figure 4.2.9** relationship between $X(z)$ and $X(\omega)$ for the sequence in Example 4.2.4, with $A = 1$ and $L = 10$

There are sequences, however, that do not satisfy the requirement in (4.2.58), for example, the sequence

$$x(n) = \frac{\sin \omega_c n}{\pi n}, \qquad -\infty < n < \infty \tag{4.2.59}$$

This sequence does not have a $z$-transform. Since it has a finite energy, its Fourier transform converges in the mean-square sense to the discontinuous function $X(\omega)$, defined as

$$X(\omega) = \begin{cases} 1, & |\omega| < \omega_c \\ 0, & \omega_c < |\omega| \leq \pi \end{cases} \tag{4.2.60}$$

In conclusion, the existence of the $z$-transform requires that (4.2.58) be satisfied for some region in the $z$-plane. If this region contains the unit circle, the Fourier transform $X(\omega)$ exists. However, the existence of the Fourier transform, which is defined for finite energy signals, does not necessarily ensure the existence of the $z$-transform.

### 4.2.7 The Cepstrum

Let us consider a sequence $\{x(n)\}$ having a $z$-transform $X(z)$. We assume that $\{x(n)\}$ is a stable sequence so that $X(z)$ converges on the unit circle. The *complex cepstrum* of the sequence $\{x(n)\}$ is defined as the sequence $\{c_x(n)\}$, which is the inverse $z$-transform of $C_x(z)$, where

$$C_x(z) = \ln X(z) \tag{4.2.61}$$

The complex cepstrum exists if $C_x(z)$ converges in the annular region $r_1 < |z| < r_2$, where $0 < r_1 < 1$ and $r_2 > 1$. Within this region of convergence, $C_x(z)$ can be represented by the Laurent series

$$C_x(z) = \ln X(z) = \sum_{n=-\infty}^{\infty} c_x(n) z^{-n} \tag{4.2.62}$$

where

$$c_x(n) = \frac{1}{2\pi j} \int_C \ln X(z) z^{n-1}\, dz \tag{4.2.63}$$

$C$ is a closed contour about the origin and lies within the region of convergence. Clearly, if $C_x(z)$ can be represented as in (4.2.62), the complex cepstrum sequence $\{c_x(n)\}$ is stable. Furthermore, if the complex cepstrum exists, $C_x(z)$ converges on the unit circle and hence we have

$$C_x(\omega) = \ln X(\omega) = \sum_{n=-\infty}^{\infty} c_x(n) e^{-j\omega n} \tag{4.2.64}$$

where $\{c_x(n)\}$ is the sequence obtained from the inverse Fourier transform of $\ln X(\omega)$, that is,

$$c_x(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \ln X(\omega) e^{j\omega n}\, d\omega \tag{4.2.65}$$

If we express $X(\omega)$ in terms of its magnitude and phase, say

$$X(\omega) = |X(\omega)|e^{j\theta(\omega)} \tag{4.2.66}$$

then

$$\ln X(\omega) = \ln |X(\omega)| + j\theta(\omega) \tag{4.2.67}$$

By substituting (4.2.67) into (4.2.65), we obtain the complex cepstrum in the form

$$c_x(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} [\ln |X(\omega)| + j\theta(\omega)]e^{j\omega n}\, d\omega \tag{4.2.68}$$

We can separate the inverse Fourier transform in (4.2.68) into the inverse Fourier transforms of $\ln |X(\omega)|$ and $\theta(\omega)$:

$$c_m(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \ln |X(\omega)|e^{j\omega n}\, d\omega \tag{4.2.69}$$

$$c_\theta(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \theta(\omega)e^{j\omega n}\, d\omega \tag{4.2.70}$$

In some applications, such as speech signal processing, only the component $c_m(n)$ is computed. In such a case the phase of $X(\omega)$ is ignored. Therefore, the sequence $\{x(n)\}$ cannot be recovered from $\{c_m(n)\}$. That is, the transformation from $\{x(n)\}$ to $\{c_m(n)\}$ is not invertible.

In speech signal processing, the (real) cepstrum has been used to separate and thus to estimate the spectral content of the speech from the pitch frequency of the speech. The complex cepstrum is used in practice to separate signals that are convolved. The process of separating two convolved signals is called *deconvolution* and the use of the complex cepstrum to perform the separation is called *homomorphic deconvolution*. This topic is discussed in Section 5.5.4.

### 4.2.8 The Fourier Transform of Signals with Poles on the Unit Circle

As was shown in Section 4.2.6, the Fourier transform of a sequence $x(n)$ can be determined by evaluating its $z$-transform $X(z)$ on the unit circle, provided that the unit circle lies within the region of convergence of $X(z)$. Otherwise, the Fourier transform does not exist.

There are some aperiodic sequences that are neither absolutely summable nor square summable. Hence their Fourier transforms do not exist. One such sequence is the unit step sequence, which has the $z$-transform

$$X(z) = \frac{1}{1 - z^{-1}}$$

Another such sequence is the causal sinusoidal signal sequence $x(n) = (\cos \omega_0 n)u(n)$. This sequence has the $z$-transform

$$X(z) = \frac{1 - z^{-1}\cos\omega_0}{1 - 2z^{-1}\cos\omega_0 + z^{-2}}$$

Note that both of these sequences have poles on the unit circle.

For sequences such as these two examples, it is sometimes useful to extend the Fourier transform representation. This can be accomplished, in a mathematically rigorous way, by allowing the Fourier transform to contain impulses at certain frequencies corresponding to the location of the poles of $X(z)$ that lie on the unit circle. The impulses are functions of the continuous frequency variable $\omega$ and have infinite amplitude, zero width, and unit area. An impulse can be viewed as the limiting form of a rectangular pulse of height $1/a$ and width $a$, in the limit as $a \to 0$. Thus, by allowing impulses in the spectrum of a signal, it is possible to extend the Fourier transform representation to some signal sequences that are neither absolutely summable nor square summable.

The following example illustrates the extension of the Fourier transform representation for three sequences.

### EXAMPLE 4.2.5

Determine the Fourier transform of the following signals.

**(a)** $x_1(n) = u(n)$

**(b)** $x_2(n) = (-1)^n u(n)$

**(c)** $x_3(n) = (\cos \omega_0 n) u(n)$

by evaluating their $z$-transforms on the unit circle.

**Solution.**

**(a)** From Table 3.3 we find that

$$X_1(z) = \frac{1}{1 - z^{-1}} = \frac{z}{z - 1}, \qquad \text{ROC: } |z| > 1$$

$X_1(z)$ has a pole, $p_1 = 1$, on the unit circle, but converges for $|z| > 1$.

If we evaluate $X_1(z)$ on the unit circle, except at $z = 1$, we obtain

$$X_1(\omega) = \frac{e^{j\omega/2}}{2j \sin(\omega/2)} = \frac{1}{2 \sin(\omega/2)} e^{j(\omega - \pi/2)}, \qquad \omega \neq 2\pi k, \quad k = 0, 1, \ldots$$

At $\omega = 0$ and multiples of $2\pi$, $X_1(\omega)$ contains impulses of area $\pi$.

Hence the presence of a pole at $z = 1$ (i.e., at $\omega = 0$) creates a problem only when we want to compute $|X_1(\omega)|$ at $\omega = 0$, because $|X_1(\omega)| \to \infty$ as $\omega \to 0$. For any other value of $\omega$, $X_1(\omega)$ is finite (i.e., well behaved). Although at first glance one might expect the signal to have zero-frequency components at all frequencies except at $\omega = 0$, this is not the case. This happens because the signal $x_1(n)$ is not a constant for all $-\infty < n < \infty$. Instead, it is *turned on* at $n = 0$. This abrupt jump creates all frequency components existing in the range $0 < \omega \leq \pi$. Generally, all signals which start at a finite time have nonzero-frequency components everywhere in the frequency axis from zero up to the folding frequency.

**(b)** From Table 3.3 we find that the $z$-transform of $a^n u(n)$ with $a = -1$ reduces to

$$X_2(z) = \frac{1}{1 + z^{-1}} = \frac{z}{z + 1}, \qquad \text{ROC: } |z| > 1$$

which has a pole at $z = -1 = e^{j\pi}$. The Fourier transform evaluated at frequencies other than $\omega = \pi$ and multiples of $2\pi$ is

$$X_2(\omega) = \frac{e^{j\omega/2}}{2\cos(\omega/2)}, \qquad \omega \neq 2\pi(k + \frac{1}{2}), \quad k = 0, 1, \ldots$$

In this case the impulse occurs at $\omega = \pi + 2\pi k$.

Hence the magnitude is

$$|X_2(\omega)| = \frac{1}{2|\cos(\omega/2)|}, \qquad \omega \neq 2\pi k + \pi, \quad k = 0, 1, \ldots$$

and the phase is

$$\measuredangle X_2(\omega) = \begin{cases} \frac{\omega}{2}, & \text{if } \cos\frac{\omega}{2} \geq 0 \\ \frac{\omega}{2} + \pi, & \text{if } \cos\frac{\omega}{2} < 0 \end{cases}$$

Note that due to the presence of the pole at $a = -1$ (i.e., at frequency $\omega = \pi$), the magnitude of the Fourier transform becomes infinite. Now $|X(\omega)| \to \infty$ as $\omega \to \pi$. We observe that $(-1)^n u(n) = (\cos \pi n)u(n)$, which is the fastest possible oscillating signal in discrete time.

**(c)** From the discussion above, it follows that $X_3(\omega)$ is infinite at the frequency component $\omega = \omega_0$. Indeed, from Table 3.3, we find that

$$x_3(n) = (\cos \omega_0 n)u(n) \overset{z}{\longleftrightarrow} X_3(z) = \frac{1 - z^{-1}\cos\omega_0}{1 - 2z^{-1}\cos\omega_0 + z^{-2}}, \qquad \text{ROC: } |z| > 1$$

The Fourier transform is

$$X_3(\omega) = \frac{1 - e^{-j\omega}\cos\omega_0}{(1 - e^{-j(\omega-\omega_0)})(1 - e^{j(\omega+\omega_0)})}, \qquad \omega \neq \pm\omega_0 + 2\pi k, \quad k = 0, 1, \ldots$$

The magnitude of $X_3(\omega)$ is given by

$$|X_3(\omega)| = \frac{|1 - e^{-j\omega}\cos\omega_0|}{|1 - e^{-j(\omega-\omega_0)}||1 - e^{-j(\omega+\omega_0)}|}, \qquad \omega \neq \pm\omega_0 + 2\pi k, \quad k = 0, 1, \ldots$$

Now if $\omega = -\omega_0$ or $\omega = \omega_0$, $|X_3(\omega)|$ becomes infinite. For all other frequencies, the Fourier transform is well behaved.

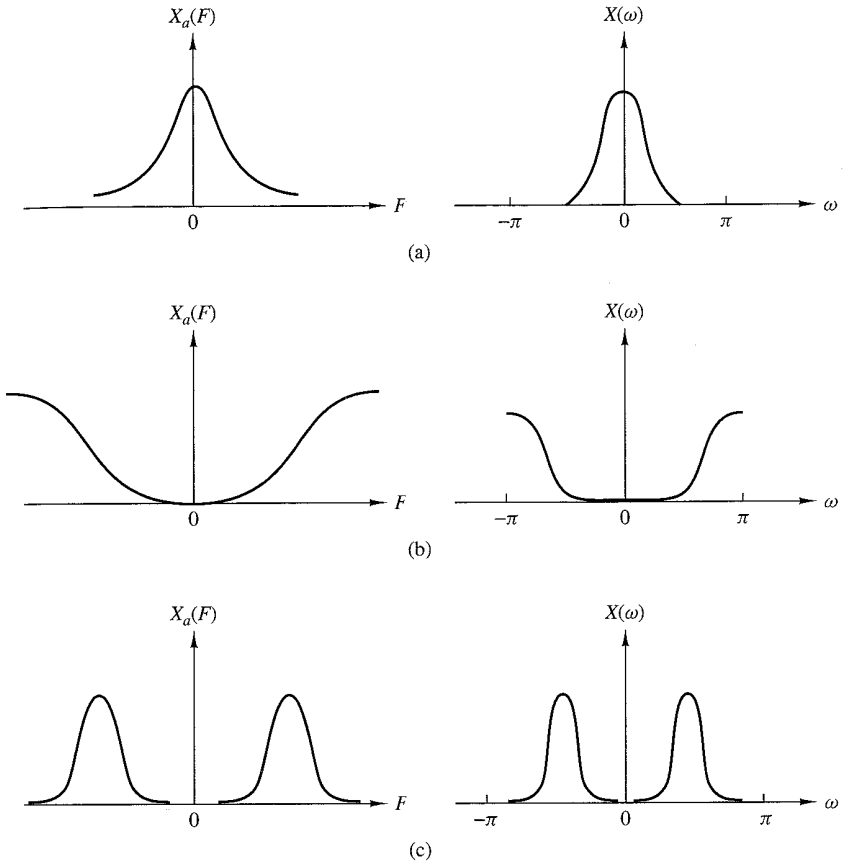



**Figure 4.2.10** (a) Low-frequency, (b) high-frequency, and (c) medium-frequency signals.

### 4.2.9 Frequency-Domain Classification of Signals: The Concept of Bandwidth

Just as we have classified signals according to their time-domain characteristics, it is also desirable to classify signals according to their frequency-domain characteristics. It is common practice to classify signals in rather broad terms according to their frequency content.

In particular, if a power signal (or energy signal) has its power density spectrum (or its energy density spectrum) concentrated about zero frequency, such a signal is called a *low-frequency signal*. Figure 4.2.10(a) illustrates the spectral characteristics of such a signal. On the other hand, if the signal power density spectrum (or the energy density spectrum) is concentrated at high frequencies, the signal is called a *high-frequency signal*. Such a signal spectrum is illustrated in Fig 4.2.10(b). A signal having a power density spectrum (or an energy density spectrum) concentrated somewhere in the broad frequency range between low frequencies and high frequencies

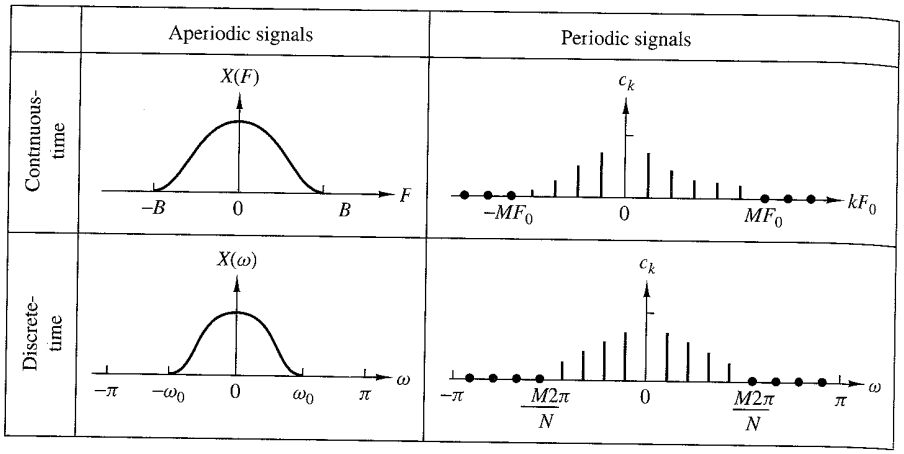



**Figure 4.2.11** Some examples of bandlimited signals.

is called a *medium-frequency signal* or a *bandpass signal*. Figure 4.2.10(c) illustrates such a signal spectrum.

In addition to this relatively broad frequency-domain classification of signals, it is often desirable to express quantitatively the range of frequencies over which the power or energy density spectrum is concentrated. This quantitative measure is called the *bandwidth* of a signal. For example, suppose that a continuous-time signal has 95% of its power (or energy) density spectrum concentrated in the frequency range $F_1 \leq F \leq F_2$. Then the 95% bandwidth of the signal is $F_2 - F_1$. In a similar manner, we may define the 75% or 90% or 99% bandwidth of the signal.

In the case of a bandpass signal, the term *narrowband* is used to describe the signal if its bandwidth $F_2 - F_1$ is much smaller (say, by a factor of 10 or more) than the median frequency $(F_2 + F_1)/2$. Otherwise, the signal is called *wideband*.

We shall say that a signal is *bandlimited* if its spectrum is zero outside the frequency range $|F| \geq B$. For example, a continuous-time finite-energy signal $x(t)$ is bandlimited if its Fourier transform $X(F) = 0$ for $|F| > B$. A discrete-time finite-energy signal $x(n)$ is said to be *(periodically) bandlimited* if

$$|X(\omega)| = 0, \qquad \text{for } \omega_0 < |\omega| < \pi$$

Similarly, a periodic continuous-time signal $x_p(t)$ is periodically bandlimited if its Fourier coefficients $c_k = 0$ for $|k| > M$, where $M$ is some positive integer. A periodic discrete-time signal with fundamental period $N$ is periodically bandlimited if the Fourier coefficients $c_k = 0$ for $k_0 < |k| < N$. Figure 4.2.11 illustrates the four types of bandlimited signals.

By exploiting the duality between the frequency domain and the time domain, we can provide similar means for characterizing signals in the time domain. In particular, a signal $x(t)$ will be called *time-limited* if

$$x(t) = 0, \qquad |t| > \tau$$

If the signal is periodic with period $T_p$, it will be called *periodically time-limited* if

$$x_p(t) = 0, \qquad \tau < |t| < T_p/2$$

If we have a discrete-time signal $x(n)$ of finite duration, that is,

$$x(n) = 0, \qquad |n| > N$$

it is also called time-limited. When the signal is periodic with fundamental period $N$, it is said to be periodically time-limited if

$$x(n) = 0, \qquad n_0 < |n| < N$$

We state, without proof, that no *signal can be time-limited and bandlimited simultaneously*. Furthermore, a reciprocal relationship exists between the time duration and the frequency duration of a signal. To elaborate, if we have a short-duration rectangular pulse in the time domain, its spectrum has a width that is inversely proportional to the duration of the time-domain pulse. The narrower the pulse becomes in the time domain, the larger the bandwidth of the signal becomes. Consequently, the product of the time duration and the bandwidth of a signal cannot be made arbitrarily small. A short-duration signal has a large bandwidth and a small bandwidth signal has a long duration. Thus, for any signal, the time–bandwidth product is fixed and cannot be made arbitrarily small.

Finally, we note that we have discussed frequency analysis methods for periodic and aperiodic signals with finite energy. However, there is a family of deterministic aperiodic signals with finite power. These signals consist of a linear superposition of complex exponentials with nonharmonically related frequencies, that is,

$$x(n) = \sum_{k=1}^{M} A_k e^{j\omega_k n}$$

where $\omega_1, \omega_2, \ldots, \omega_M$ are nonharmonically related. These signals have discrete spectra but the distances among the lines are nonharmonically related. Signals with discrete nonharmonic spectra are sometimes called quasi-periodic.

### 4.2.10 The Frequency Ranges of Some Natural Signals

The frequency analysis tools that we have developed in this chapter are usually applied to a variety of signals that are encountered in practice (e.g., seismic, biological, and electromagnetic signals). In general, the frequency analysis is performed for the purpose of extracting information from the observed signal. For example, in the case of biological signals, such as an ECG signal, the analytical tools are used to extract information relevant for diagnostic purposes. In the case of seismic signals, we may be interested in detecting the presence of a nuclear explosion or in determining the characteristics and location of an earthquake. An electromagnetic signal, such as a radar signal reflected from an airplane, contains information on the position of the

plane and its radial velocity. These parameters can be estimated from observation of the received radar signal.

In processing any signal for the purpose of measuring parameters or extracting other types of information, one must know approximately the range of frequencies contained by the signal. For reference, Tables 4.1, 4.2, and 4.3 give approximate limits in the frequency domain for biological, seismic, and electromagnetic signals.

## 4.3 Frequency-Domain and Time-Domain Signal Properties

In the previous sections of the chapter we have introduced several methods for the frequency analysis of signals. Several methods were necessary to accommodate the different types of signals. To summarize, the following frequency analysis tools have been introduced:

1. The Fourier series for continuous-time periodic signals.
2. The Fourier transform for continuous-time aperiodic signals.
3. The Fourier series for discrete-time periodic signals.
4. The Fourier transform for discrete-time aperiodic signals.

**TABLE 4.1** Frequency Ranges of Some Biological Signals

| Type of Signal | Frequency Range (Hz) |
|---|---|
| Electroretinogram[a] | 0–20 |
| Electronystagmogram[b] | 0–20 |
| Pneumogram[c] | 0–40 |
| Electrocardiogram (ECG) | 0–100 |
| Electroencephalogram (EEG) | 0–100 |
| Electromyogram[d] | 10–200 |
| Sphygmomanogram[e] | 0–200 |
| Speech | 100–4000 |

[a] A graphic recording of retina characteristics.
[b] A graphic recording of involuntary movement of the eyes.
[c] A graphic recording of respiratory activity.
[d] A graphic recording of muscular action, such as muscular contraction.
[e] A recording of blood pressure.

**TABLE 4.2** Frequency Ranges of Some Seismic Signals

| Type of Signal | Frequency Range (Hz) |
|---|---|
| Wind noise | 100–1000 |
| Seismic exploration signals | 10–100 |
| Earthquake and nuclear explosion signals | 0.01–10 |
| Seismic noise | 0.1–1 |

**TABLE 4.3** Frequency Ranges of Electromagnetic Signals

| Type of Signal | Wavelength (m) | Frequency Range (Hz) |
|---|---|---|
| Radio broadcast | $10^4$–$10^2$ | $3 \times 10^4$–$3 \times 10^6$ |
| Shortwave radio signals | $10^2$–$10^{-2}$ | $3 \times 10^6$–$3 \times 10^{10}$ |
| Radar, satellite communications, space communications, common-carrier microwave | 1–$10^{-2}$ | $3 \times 10^8$–$3 \times 10^{10}$ |
| Infrared | $10^{-3}$–$10^{-6}$ | $3 \times 10^{11}$–$3 \times 10^{14}$ |
| Visible light | $3.9 \times 10^{-7}$–$8.1 \times 10^{-7}$ | $3.7 \times 10^{14}$–$7.7 \times 10^{14}$ |
| Ultraviolet | $10^{-7}$–$10^{-8}$ | $3 \times 10^{15}$–$3 \times 10^{16}$ |
| Gamma rays and X rays | $10^{-9}$–$10^{-10}$ | $3 \times 10^{17}$–$3 \times 10^{18}$ |

Figure 4.3.1 summarizes the analysis and synthesis formulas for these types of signals.

As we have already indicated several times, there are two time-domain characteristics that determine the type of signal spectrum we obtain. These are whether the time variable is continuous or discrete, and whether the signal is periodic or aperiodic. Let us briefly summarize the results of the previous sections.

**Continuous-time signals have aperiodic spectra** A close inspection of the Fourier series and Fourier transform analysis formulas for continuous-time signals does not reveal any kind of periodicity in the spectral domain. This lack of periodicity is a consequence of the fact that the complex exponential $\exp(j2\pi Ft)$ is a function of the continuous variable t, and hence it is not periodic in $F$. Thus the frequency range of continuous-time signals extends from $F = 0$ to $F = \infty$.

**Discrete-time signals have periodic spectra** Indeed, both the Fourier series and the Fourier transform for discrete-time signals are periodic with period $\omega = 2\pi$. As a result of this periodicity, the frequency range of discrete-time signals is finite and extends from $\omega = -\pi$ to $\omega = \pi$ radians, where $\omega = \pi$ corresponds to the highest possible rate of oscillation.

**Periodic signals have discrete spectra** As we have observed, periodic signals are described by means of Fourier series. The Fourier series coefficients provide the "lines" that constitute the discrete spectrum. The line spacing $\Delta F$ or $\Delta f$ is equal to the inverse of the period $T_p$ or $N$, respectively, in the time domain. That is, $\Delta F = 1/T_p$ for continuous-time periodic signals and $\Delta f = 1/N$ for discrete-time signals.

**Aperiodic finite energy signals have continuous spectra** This property is a direct consequence of the fact that both $X(F)$ and $X(\omega)$ are functions of $\exp(j2\pi Ft)$ and $\exp(j\omega n)$, respectively, which are continuous functions of the variables $F$ and $\omega$. The continuity in frequency is necessary to break the harmony and thus create aperiodic signals.

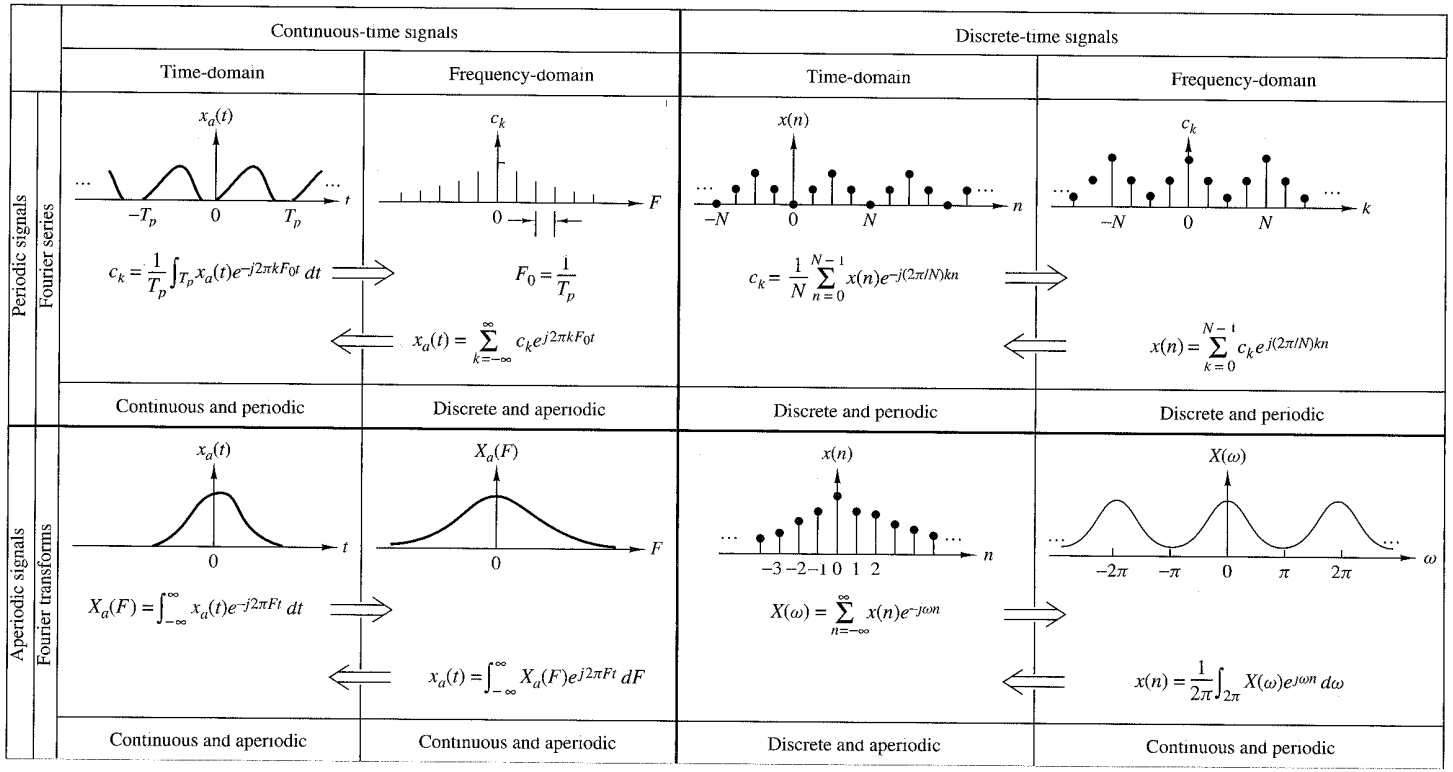



**Figure 4.3.1** Summary of analysis and synthesis formulas.

In summary, we can conclude that *periodicity with "period" $\alpha$ in one domain automatically implies discretization with "spacing" of $1/\alpha$ in the other domain, and vice versa.*

If we keep in mind that "period" in the frequency domain means the frequency range, "spacing" in the time domain is the sampling period $T$, line spacing in the frequency domain is $\Delta F$, then $\alpha = T_p$ implies that $1/\alpha = 1/T_p = \Delta F$, $\alpha = N$ implies that $\Delta f = 1/N$, and $\alpha = F_s$ implies that $T = 1/F_s$.

These time-frequency dualities are apparent from observation of Fig 4.3.1. We stress, however, that the illustrations used in this figure do not correspond to any actual transform pairs. Thus any comparison among them should be avoided.

A careful inspection of Fig 4.3.1 also reveals some mathematical symmetries and dualities among the several frequency analysis relationships. In particular, we observe that there are dualities between the following analysis and synthesis equations:

1. The analysis and synthesis equations of the continuous-time Fourier transform.
2. The analysis and synthesis equations of the discrete-time Fourier series.
3. The analysis equation of the continuous-time Fourier series and the synthesis equation of the discrete-time Fourier transform.
4. The analysis equation of the discrete-time Fourier transform and the synthesis equation of the continuous-time Fourier series.

Note that all dual relations differ only in the sign of the exponent of the corresponding complex exponential. It is interesting to note that this change in sign can be thought of either as a folding of the signal or a folding of the spectrum, since

$$e^{-j2\pi Ft} = e^{j2\pi(-F)t} = e^{j2\pi F(-t)}$$

If we turn our attention now to the spectral density of signals, we recall that we have used the term *energy density spectrum* for characterizing finite-energy aperiodic signals and the term *power density spectrum* for periodic signals. This terminology is consistent with the fact that periodic signals are power signals and aperiodic signals with finite energy are energy signals.

## 4.4 Properties of the Fourier Transform for Discrete-Time Signals

The Fourier transform for aperiodic finite-energy discrete-time signals described in the preceding section possesses a number of properties that are very useful in reducing the complexity of frequency analysis problems in many practical applications. In this section we develop the important properties of the Fourier transform. Similar properties hold for the Fourier transform of aperiodic finite-energy continuous-time signals.

For convenience, we adopt the notation

$$X(\omega) \equiv F\{x(n)\} = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} \tag{4.4.1}$$

for the direct transform (analysis equation) and

$$x(n) \equiv F^{-1}\{X(\omega)\} = \frac{1}{2\pi}\int_{2\pi} X(\omega)e^{j\omega n}\, d\omega \tag{4.4.2}$$

for the inverse transform (synthesis equation). We also refer to $x(n)$ and $X(\omega)$ as a *Fourier transform pair* and denote this relationship with the notation

$$x(n) \overset{F}{\longleftrightarrow} X(\omega) \tag{4.4.3}$$

Recall that $X(\omega)$ is periodic with period $2\pi$. Consequently, any interval of length $2\pi$ is sufficient for the specification of the spectrum. Usually, we plot the spectrum in the fundamental interval $[-\pi, \pi]$. We emphasize that all the spectral information contained in the fundamental interval is necessary for the complete description or characterization of the signal. For this reason, the range of integration in (4.4.2) is always $2\pi$, independent of the specific characteristics of the signal within the fundamental interval.

### 4.4.1 Symmetry Properties of the Fourier Transform

When a signal satisfies some symmetry properties in the time domain, these properties impose some symmetry conditions on its Fourier transform. Exploitation of any symmetry characteristics leads to simpler formulas for both the direct and inverse Fourier transform. A discussion of various symmetry properties and the implications of these properties in the frequency domain is given here.

Suppose that both the signal $x(n)$ and its transform $X(\omega)$ are complex-valued functions. Then they can be expressed in rectangular form as

$$x(n) = x_R(n) + jx_I(n) \tag{4.4.4}$$

$$X(\omega) = X_R(\omega) + jX_I(\omega) \tag{4.4.5}$$

By substituting (4.4.4) and $e^{-j\omega} = \cos\omega - j\sin\omega$ into (4.4.1) and separating the real and imaginary parts, we obtain

$$X_R(\omega) = \sum_{n=-\infty}^{\infty} [x_R(n)\cos\omega n + x_I(n)\sin\omega n] \tag{4.4.6}$$

$$X_I(\omega) = -\sum_{n=-\infty}^{\infty} [x_R(n)\sin\omega n - x_I(n)\cos\omega n] \tag{4.4.7}$$

In a similar manner, by substituting (4.4.5) and $e^{j\omega} = \cos\omega + j\sin\omega$ into (4.4.2), we obtain

$$x_R(n) = \frac{1}{2\pi}\int_{2\pi} [X_R(\omega)\cos\omega n - X_I(\omega)\sin\omega n]\, d\omega \tag{4.4.8}$$

$$x_I(n) = \frac{1}{2\pi}\int_{2\pi} [X_R(\omega)\sin\omega n + X_I(\omega)\cos\omega n]\, d\omega \tag{4.4.9}$$

Now, let us investigate some special cases.

**Real signals.** If $x(n)$ is real, then $x_R(n) = x(n)$ and $x_I(n) = 0$. Hence (4.4.6) and (4.4.7) reduce to

$$X_R(\omega) = \sum_{n=-\infty}^{\infty} x(n)\cos\omega n \tag{4.4.10}$$

and

$$X_I(\omega) = -\sum_{n=-\infty}^{\infty} x(n)\sin\omega n \tag{4.4.11}$$

Since $\cos(-\omega n) = \cos\omega n$ and $\sin(-\omega n) = -\sin\omega n$, it follows from (4.4.10) and (4.4.11) that

$$X_R(-\omega) = X_R(\omega)\ , \qquad \text{(even)} \tag{4.4.12}$$

$$X_I(-\omega) = -X_I(\omega), \qquad \text{(odd)} \tag{4.4.13}$$

If we combine (4.4.12) and (4.4.13) into a single equation, we have

$$X^*(\omega) = X(-\omega) \tag{4.4.14}$$

In this case we say that the spectrum of a real signal has *Hermitian symmetry*.

With the aid of Fig 4.4.1, we observe that the magnitude and phase spectra for real signals are

$$|X(\omega)| = \sqrt{X_R^2(\omega) + X_I^2(\omega)} \tag{4.4.15}$$

$$\measuredangle X|\omega| = \tan^{-1}\frac{X_I(\omega)}{X_R(\omega)} \tag{4.4.16}$$

As a consequence of (4.4.12) and (4.4.13), the magnitude and phase spectra also possess the symmetry properties

$$|X(\omega)| = |X(-\omega)|\ , \qquad \text{(even)} \tag{4.4.17}$$

$$\measuredangle X(-\omega) = -\measuredangle X(\omega), \qquad \text{(odd)} \tag{4.4.18}$$

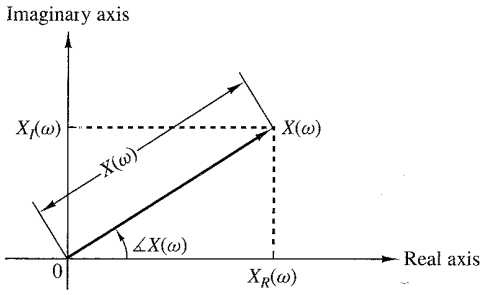



**Figure 4.4.1**
Magnitude and phase functions.

In the case of the inverse transform of a real-valued signal [i.e., $x(n) = x_R(n)$], (4.4.8) implies that

$$x(n) = \frac{1}{2\pi} \int_{2\pi} [X_R(\omega) \cos \omega n - X_I(\omega) \sin \omega n] \, d\omega \tag{4.4.19}$$

Since both products $X_R(\omega) \cos \omega n$ and $X_I(\omega) \sin \omega n$ are even functions of $\omega$, we have

$$x(n) = \frac{1}{\pi} \int_0^{\pi} [X_R(\omega) \cos \omega n - X_I(\omega) \sin \omega n] \, d\omega \tag{4.4.20}$$

**Real and even signals.** If $x(n)$ is real and even [i.e., $x(-n) = x(n)$], then $x(n) \cos \omega n$ is even and $x(n) \sin \omega n$ is odd. Hence, from (4.4.10), (4.4.11), and (4.4.20) we obtain

$$X_R(\omega) = x(0) + 2 \sum_{n=1}^{\infty} x(n) \cos \omega n, \qquad \text{(even)} \tag{4.4.21}$$

$$X_I(\omega) = 0 \tag{4.4.22}$$

$$x(n) = \frac{1}{\pi} \int_0^{\pi} X_R(\omega) \cos \omega n \, d\omega \tag{4.4.23}$$

Thus real and even signals possess real-valued spectra, which, in addition, are even functions of the frequency variable $\omega$.

**Real and odd signals.** If $x(n)$ is real and odd [i.e., $x(-n) = -x(n)$], then $x(n) \cos \omega n$ is odd and $x(n) \sin \omega n$ is even. Consequently, (4.4.10), (4.4.11) and (4.4.20) imply that

$$X_R(\omega) = 0 \tag{4.4.24}$$

$$X_I(\omega) = -2 \sum_{n=1}^{\infty} x(n) \sin \omega n, \qquad \text{(odd)} \tag{4.4.25}$$

$$x(n) = -\frac{1}{\pi} \int_0^{\pi} X_I(\omega) \sin \omega n \, d\omega \tag{4.4.26}$$

Thus real-valued odd signals possess purely imaginary-valued spectral characteristics, which, in addition, are odd functions of the frequency variable $\omega$.

**Purely imaginary signals.** In this case $x_R(n) = 0$ and $x(n) = jx_I(n)$. Thus (4.4.6), (4.4.7), and (4.4.9) reduce to

$$X_R(\omega) = \sum_{n=-\infty}^{\infty} x_I(n) \sin \omega n, \qquad \text{(odd)} \tag{4.4.27}$$

$$X_I(\omega) = \sum_{n=-\infty}^{\infty} x_I(n) \cos \omega n, \qquad \text{(even)} \tag{4.4.28}$$

$$x_I(n) = \frac{1}{\pi} \int_0^{\pi} [X_R(\omega) \sin \omega n + X_I(\omega) \cos \omega n] \, d\omega \tag{4.4.29}$$

If $x_I(n)$ is odd [i.e., $x_I(-n) = -x_I(n)$], then

$$X_R(\omega) = 2 \sum_{n=1}^{\infty} x_I(n) \sin \omega n, \qquad \text{(odd)} \tag{4.4.30}$$

$$X_I(\omega) = 0 \tag{4.4.31}$$

$$x_I(n) = \frac{1}{\pi} \int_0^{\pi} X_R(\omega) \sin \omega n \, d\omega \tag{4.4.32}$$

Similarly, if $x_I(n)$ is even [i.e., $x_I(-n) = x_I(n)$], we have

$$X_R(\omega) = 0 \tag{4.4.33}$$

$$X_I(\omega) = x_I(0) + 2 \sum_{n=1}^{\infty} x_I(n) \cos \omega n, \qquad \text{(even)} \tag{4.4.34}$$

$$x_I(n) = \frac{1}{\pi} \int_0^{\pi} X_I(\omega) \cos \omega n \, d\omega \tag{4.4.35}$$

An arbitrary, possibly complex-valued signal $x(n)$ can be decomposed as

$$x(n) = x_R(n) + jx_I(n) = x_R^e(n) + x_R^o(n) + j[x_I^e(n) + x_I^o(n)] = x_e(n) + x_o(n) \tag{4.4.36}$$

where, by definition,

$$x_e(n) = x_R^e(n) + jx_I^e(n) = \frac{1}{2}[x(n) + x^*(-n)]$$

$$x_o(n) = x_R^o(n) + jx_I^o(n) = \frac{1}{2}[x(n) - x^*(-n)]$$

The superscripts $e$ and $o$ denote the even and odd signal components, respectively. We note that $x_e(n) = x_e(-n)$ and $x_o(-n) = -x_o(n)$. From (4.4.36) and the Fourier transform properties established above, we obtain the following relationships:

$$\begin{array}{c} x(n) = [x_R^e(n) + jx_I^e(n)] + [x_R^o(n) + jx_I^o(n)] = x_e(n) + x_o(n) \\ \updownarrow \qquad\quad \updownarrow \qquad\qquad \nwarrow\!\!\!\!\!\!\nearrow \qquad\quad \updownarrow \qquad\quad \updownarrow \\ X(\omega) = [X_R^e(\omega) + jX_I^e(\omega)] + [X_R^o(\omega) - jX_I^o(\omega)] = X_e(\omega) + X_o(\omega) \end{array} \tag{4.4.37}$$

These symmetry properties of the Fourier transform are summarized in Table 4.4 and in Fig 4.4.2. They are often used to simplify Fourier transform calculations in practice.

**TABLE 4.4** Symmetry Properties of the Discrete-Time Fourier Transform

| Sequence | DTFT |
|---|---|
| $x(n)$ | $X(\omega)$ |
| $x^*(n)$ | $X^*(-\omega)$ |
| $x^*(-n)$ | $X^*(\omega)$ |
| $x_R(n)$ | $X_e(\omega) = \frac{1}{2}[X(\omega) + X^*(-\omega)]$ |
| $jx_I(n)$ | $X_o(\omega) = \frac{1}{2}[X(\omega) - X^*(-\omega)]$ |
| $x_e(n) = \frac{1}{2}[x(n) + x^*(-n)]$ | $X_R(\omega)$ |
| $x_o(n) = \frac{1}{2}[x(n) - x^*(-n)]$ | $jX_I(\omega)$ |
| Real Signals | |
| | $X(\omega) = X^*(-\omega)$ |
| Any real signal | $X_R(\omega) = X_R(-\omega)$ |
| $x(n)$ | $X_I(\omega) = -X_I(-\omega)$ |
| | $\lvert X(\omega)\rvert = \lvert X(-\omega)\rvert$ |
| | $\measuredangle X(\omega) = -\measuredangle X(-\omega)$ |
| $x_e(n) = \frac{1}{2}[x(n) + x(-n)]$ | $X_R(\omega)$ |
| (real and even) | (real and even) |
| $x_o(n) = \frac{1}{2}[x(n) - x(-n)]$ | $jX_I(\omega)$ |
| (real and odd) | (imaginary and odd) |

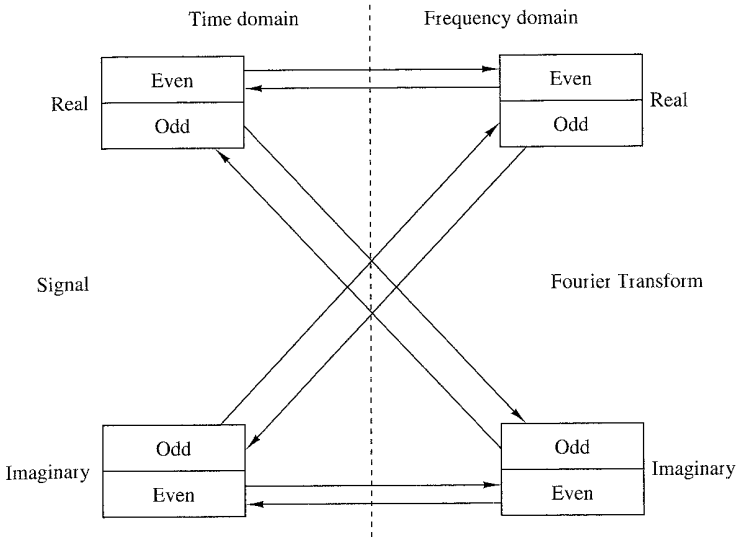



**Figure 4.4.2** Summary of symmetry properties for the Fourier transform.

EXAMPLE 4.4.1

Determine and sketch $X_R(\omega)$, $X_I(\omega)$, $|X(\omega)|$, and $\measuredangle X(\omega)$ for the Fourier transform

$$X(\omega) = \frac{1}{1 - ae^{-j\omega}}, \qquad -1 < a < 1 \tag{4.4.38}$$

**Solution.** By multiplying both the numerator and denominator of (4.4.38) by the complex conjugate of the denominator, we obtain

$$X(\omega) = \frac{1 - ae^{j\omega}}{(1 - ae^{-j\omega})(1 - ae^{j\omega})} = \frac{1 - a\cos\omega - ja\sin\omega}{1 - 2a\cos\omega + a^2}$$

This expression can be subdivided into real and imaginary parts. Thus we obtain

$$X_R(\omega) = \frac{1 - a\cos\omega}{1 - 2a\cos\omega + a^2}$$

$$X_I(\omega) = -\frac{a\sin\omega}{1 - 2a\cos\omega + a^2}$$

Substitution of the last two equations into (4.4.15) and (4.4.16) yields the magnitude and phase spectra as

$$|X(\omega)| = \frac{1}{\sqrt{1 - 2a\cos\omega + a^2}} \tag{4.4.39}$$

and

$$\measuredangle X(\omega) = -\tan^{-1}\frac{a\sin\omega}{1 - a\cos\omega} \tag{4.4.40}$$

Figures 4.4.3 and 4.4.4 show the graphical representation of these spectra for $a = 0.8$. The reader can easily verify that as expected, all symmetry properties for the spectra of real signals apply to this case.

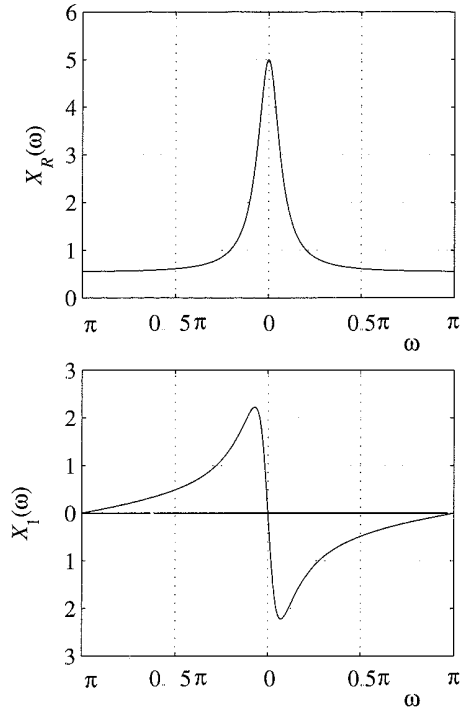



**Figure 4.4.3**
Graph of $X_R(\omega)$ and $X_I(\omega)$ for the transform in Example 4.4.1.

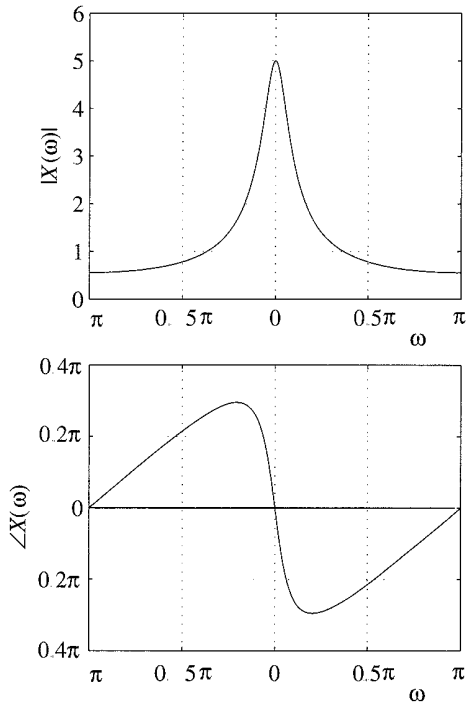



**Figure 4.4.4**
Magnitude and phase spectra of the transform in Example 4.4.1.

EXAMPLE 4.4.2

Determine the Fourier transform of the signal

$$x(n) = \begin{cases} A, & -M \le n \le M \\ 0, & \text{elsewhere} \end{cases} \tag{4.4.41}$$

**Solution.** Clearly, $x(-n) = x(n)$. Thus $x(n)$ is a real and even signal. From (4.4.21) we obtain

$$X(\omega) = X_R(\omega) = A\left(1 + 2\sum_{n=1}^{M} \cos \omega n\right)$$

If we use the identity given in Problem 4.13, we obtain the simpler form

$$X(\omega) = A\frac{\sin(M + \frac{1}{2})\omega}{\sin(\omega/2)}$$

Since $X(\omega)$ is real, the magnitude and phase spectra are given by

$$|X(\omega)| = \left| A\frac{\sin(M + \frac{1}{2})\omega}{\sin(\omega/2)} \right| \tag{4.4.42}$$

and

$$\measuredangle X(\omega) = \begin{cases} 0, & \text{if } X(\omega) > 0 \\ \pi, & \text{if } X(\omega) < 0 \end{cases} \tag{4.4.43}$$

Figure 4.4.5 shows the graphs for $X(\omega)$.

---

### 4.4.2 Fourier Transform Theorems and Properties

In this section we introduce several Fourier transform theorems and illustrate their use in practice by examples.

**Linearity.** If

$$x_1(n) \overset{F}{\longleftrightarrow} X_1(\omega)$$

and

$$x_2(n) \overset{F}{\longleftrightarrow} X_2(\omega)$$

then

$$a_1x_1(n) + a_2x_2(n) \overset{F}{\longleftrightarrow} a_1X_1(\omega) + a_2X_2(\omega) \tag{4.4.44}$$

Simply stated, the Fourier transformation, viewed as an operation on a signal $x(n)$, is a linear transformation. Thus the Fourier transform of a linear combination of two or more signals is equal to the same linear combination of the Fourier transforms of the individual signals. This property is easily proved by using (4.4.1). The linearity property makes the Fourier transform suitable for the study of linear systems.

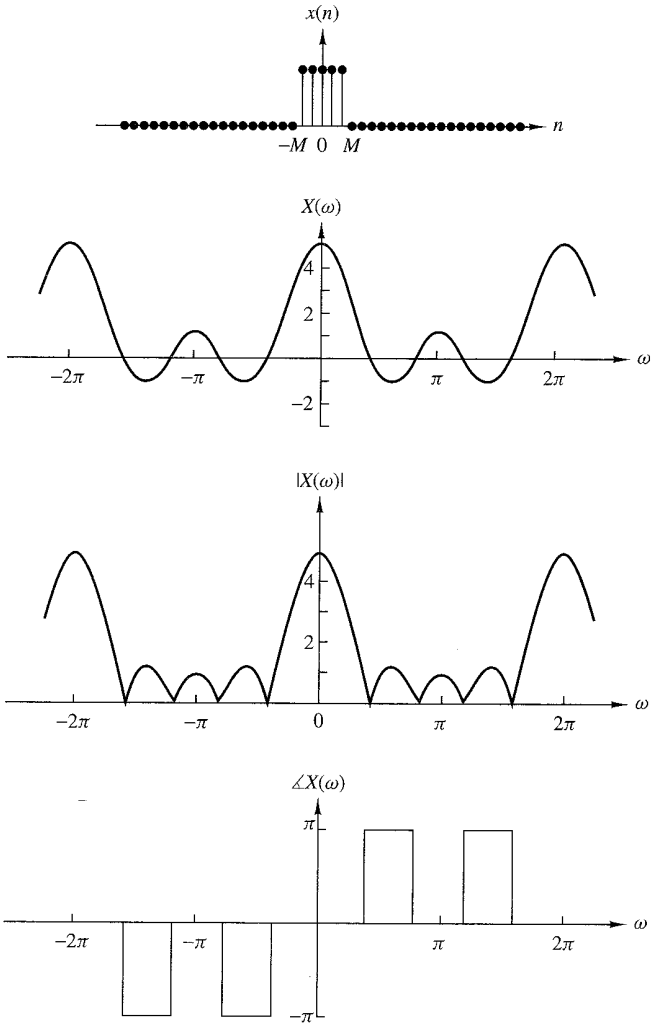



**Figure 4.4.5** Spectral characteristics of rectangular pulse in Example 4.4.2.

## EXAMPLE 4.4.3

Determine the Fourier transform of the signal

$$x(n) = a^{|n|}, \qquad -1 < a < 1 \tag{4.4.45}$$

**Solution.** First, we observe that $x(n)$ can be expressed as

$$x(n) = x_1(n) + x_2(n)$$

where

$$x_1(n) = \begin{cases} a^n, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

and

$$x_2(n) = \begin{cases} a^{-n}, & n < 0 \\ 0, & n \geq 0 \end{cases}$$

Beginning with the definition of the Fourier transform in (4.4.1), we have

$$X_1(\omega) = \sum_{n=-\infty}^{\infty} x_1(n)e^{-j\omega n} = \sum_{n=0}^{\infty} a^n e^{-j\omega n} = \sum_{n=0}^{\infty} (ae^{-j\omega})^n$$

The summation is a geometric series that converges to

$$X_1(\omega) = \frac{1}{1 - ae^{-j\omega}}$$

provided that

$$|ae^{-j\omega}| = |a| \cdot |e^{-j\omega}| = |a| < 1$$

which is a condition that is satisfied in this problem. Similarly, the Fourier transform of $x_2(n)$ is

$$\begin{aligned} X_2(\omega) &= \sum_{n=-\infty}^{\infty} x_2(n)e^{-j\omega n} = \sum_{n=-\infty}^{-1} a^{-n}e^{-j\omega n} \\ &= \sum_{n=-\infty}^{-1} (ae^{j\omega})^{-n} = \sum_{k=1}^{\infty} (ae^{j\omega})^k \\ &= \frac{ae^{j\omega}}{1 - ae^{j\omega}} \end{aligned}$$

By combining these two transforms, we obtain the Fourier transform of $x(n)$ in the form

$$\begin{aligned} X(\omega) &= X_1(\omega) + X_2(\omega) \\ &= \frac{1 - a^2}{1 - 2a\cos\omega + a^2} \end{aligned} \tag{4.4.46}$$

Figure 4.4.6 illustrates $x(n)$ and $X(\omega)$ for the case in which $a = 0.8$.

---

**Time shifting.** If

$$x(n) \overset{F}{\longleftrightarrow} X(\omega) \tag{4.4.47}$$

then

$$x(n-k) \overset{F}{\longleftrightarrow} e^{-j\omega k}X(\omega)$$

The proof of this property follows immediately from the Fourier transform of $x(n-k)$ by making a change in the summation index. Thus

$$\begin{aligned} F\{x(n-k)\} &= X(\omega)e^{-j\omega k} \\ &= |X(\omega)|e^{j[\measuredangle X(\omega) - \omega k]} \end{aligned}$$

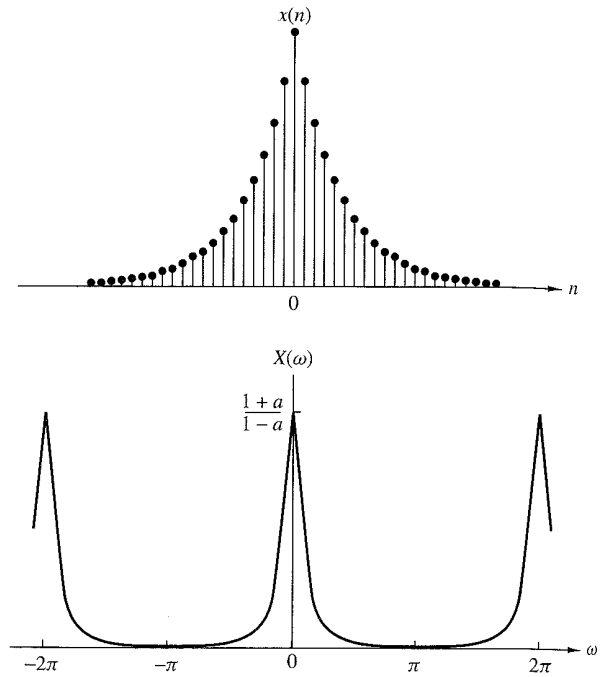



**Figure 4.4.6**
Sequence $x(n)$ and its Fourier transform in Example 4.4.3 with $a = 0.8$.

This relation means that if a signal is shifted in the time domain by $k$ samples, its magnitude spectrum remains unchanged. However, the phase spectrum is changed by an amount $-\omega k$. This result can easily be explained if we recall that the frequency content of a signal depends only on its shape. From a mathematical point of view, we can say that shifting by $k$ in the time domain is equivalent to multiplying the spectrum by $e^{-j\omega k}$ in the frequency domain.

**Time reversal.** If

$$x(n) \overset{F}{\longleftrightarrow} X(\omega)$$

then

$$x(-n) \overset{F}{\longleftrightarrow} X(-\omega) \tag{4.4.48}$$

This property can be established by performing the Fourier transformation of $x(-n)$ and making a simple change in the summation index. Thus

$$F\{x(-n)\} = \sum_{l=-\infty}^{\infty} x(l)e^{j\omega l} = X(-\omega)$$

If $x(n)$ is real, then from (4.4.17) and (4.4.18) we obtain

$$\begin{aligned} F\{x(-n)\} &= X(-\omega) = |X(-\omega)|e^{j\measuredangle X(-\omega)} \\ &= |X(\omega)|e^{-j\measuredangle X(\omega)} \end{aligned}$$

This means that if a signal is folded about the origin in time, its magnitude spectrum remains unchanged, and the phase spectrum undergoes a change in sign (phase reversal).

**Convolution theorem.** If

$$x_1(n) \overset{F}{\longleftrightarrow} X_1(\omega)$$

and

$$x_2(n) \overset{F}{\longleftrightarrow} X_2(\omega)$$

then

$$x(n) = x_1(n) * x_2(n) \overset{F}{\longleftrightarrow} X(\omega) = X_1(\omega)X_2(\omega) \tag{4.4.49}$$

To prove (4.4.49), we recall the convolution formula

$$x(n) = x_1(n) * x_2(n) = \sum_{k=-\infty}^{\infty} x_1(k)x_2(n-k)$$

By multiplying both sides of this equation by the exponential $\exp(-j\omega n)$ and summing over all $n$, we obtain

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} = \sum_{n=-\infty}^{\infty}\left[\sum_{k=-\infty}^{\infty} x_1(k)x_2(n-k)\right]e^{-j\omega n}$$

After interchanging the order of the summations and making a simple change in the summation index, the right-hand side of this equation reduces to the product $X_1(\omega)X_2(\omega)$. Thus (4.4.49) is established.

The convolution theorem is one of the most powerful tools in linear systems analysis. That is, if we convolve two signals in the time domain, then this is equivalent to multiplying their spectra in the frequency domain. In later chapters we will see that the convolution theorem provides an important computational tool for many digital signal processing applications.

### EXAMPLE 4.4.4

By use of (4.4.49), determine the convolution of the sequences

$$x_1(n) = x_2(n) = \{1, \underset{\uparrow}{1}, 1\}$$

**Solution.** By using (4.4.21), we obtain

$$X_1(\omega) = X_2(\omega) = 1 + 2\cos\omega$$

Then

$$\begin{aligned} X(\omega) = X_1(\omega)X_2(\omega) &= (1 + 2\cos\omega)^2 \\ &= 3 + 4\cos\omega + 2\cos 2\omega \\ &= 3 + 2(e^{j\omega} + e^{-j\omega}) + (e^{j2\omega} + e^{-j2\omega}) \end{aligned}$$

Hence the convolution of $x_1(n)$ with $x_2(n)$ is

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

Figure 4.4.7 illustrates the foregoing relationships

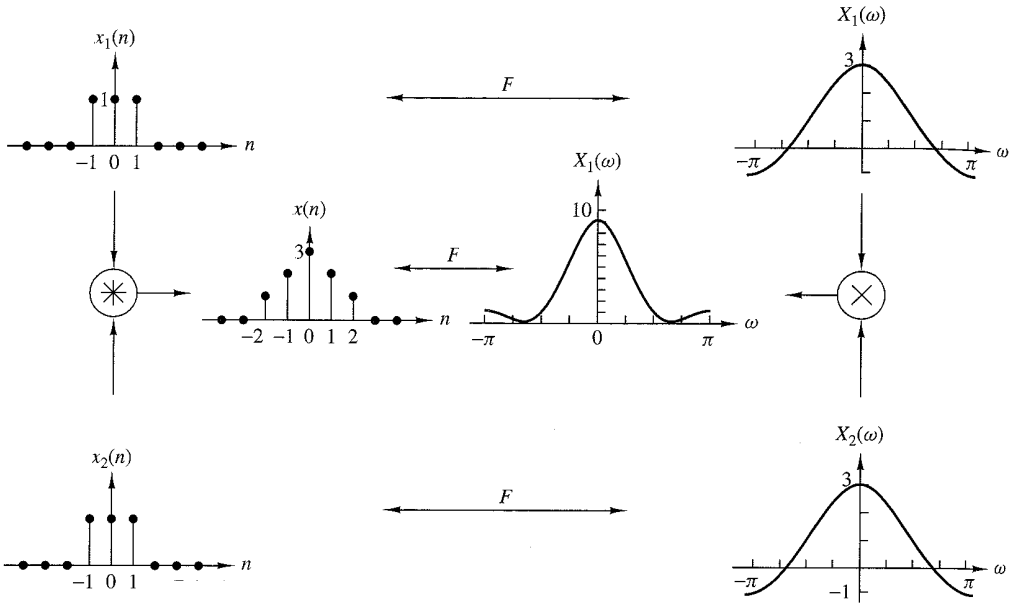



**Figure 4.4.7** Graphical representation of the convolution property.

---

**The correlation theorem.** If

$$x_1(n) \overset{F}{\longleftrightarrow} X_1(\omega)$$

and

$$x_2(n) \overset{F}{\longleftrightarrow} X_2(\omega)$$

then

$$r_{x_1x_2}(m) \overset{F}{\longleftrightarrow} S_{x_1x_2}(\omega) = X_1(\omega)X_2(-\omega) \tag{4.4.50}$$

The proof of (4.4.50) is similar to the proof of (4.4.49). In this case, we have

$$r_{x_1x_2}(n) = \sum_{k=-\infty}^{\infty} x_1(k)x_2(k-n)$$

By multiplying both sides of this equation by the exponential $\exp(-j\omega n)$ and summing over all $n$, we obtain

$$S_{x_1x_2}(\omega) = \sum_{n=-\infty}^{\infty} r_{x1x2}(n)e^{-j\omega n} = \sum_{n=-\infty}^{\infty}\left[\sum_{k=-\infty}^{\infty} x_1(k)x_2(k-n)\right]e^{-j\omega n}$$

Finally, we interchange the order of the summations and make a change in the summation index. Thus we find that the right-hand side of the equation above reduces to $X_1(\omega)X_2(-\omega)$. The function $S_{x_1x_2}(\omega)$ is called the *cross-energy density spectrum* of the signals $x_1(n)$ and $x_2(n)$.

**The Wiener–Khintchine theorem.** Let $x(n)$ be a real signal. Then

$$r_{xx}(l) \overset{F}{\longleftrightarrow} S_{xx}(\omega) \tag{4.4.51}$$

That is, the energy spectral density of an energy signal is the Fourier transform of its autocorrelation sequence. This is a special case of (4.4.50).

This is a very important result. It means that the autocorrelation sequence of a signal and its energy spectral density contain the same information about the signal. Since neither of these contains any phase information, it is impossible to uniquely reconstruct the signal from the autocorrelation function or the energy density spectrum.

EXAMPLE 4.4.5

Determine the energy density spectrum of the signal

$$x(n) = a^n u(n), \qquad -1 < a < 1$$

**Solution.** From Example 2.6.2 we found that the autocorrelation function for this signal is

$$r_{xx}(l) = \frac{1}{1-a2}a^{|l|}, \qquad \infty < l < \infty$$

By using the result in (4.4.46) for the Fourier transform of $a^{|l|}$, derived in Example 4.4.3, we have

$$F\{r_{xx}(l)\} = \frac{1}{1-a^2}F\{a^{|l|}\} = \frac{1}{1-2a\cos\omega + a^2}$$

Thus, according to the Wiener–Khintchine theorem,

$$S_{xx}(\omega) = \frac{1}{1-2a\cos\omega + a^2}$$

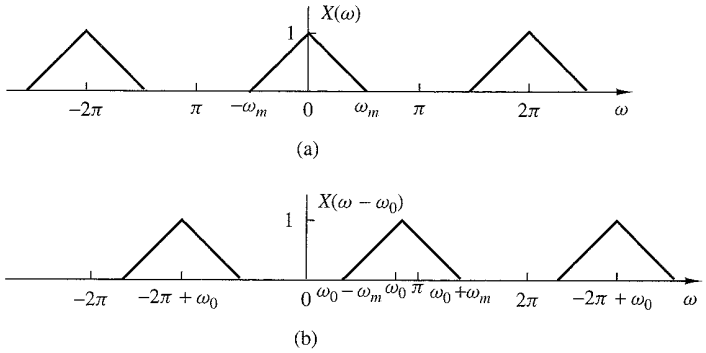



**Figure 4.4.8** Illustration of the frequency-shifting property of the Fourier transform ($\omega_0 \leq 2\pi - \omega_m$).

**Frequency shifting.** If

$$x(n) \overset{F}{\longleftrightarrow} X(\omega)$$

then

$$e^{j\omega_0 n} x(n) \overset{F}{\longleftrightarrow} X(\omega - \omega_0) \tag{4.4.52}$$

This property is easily proved by direct substitution into the analysis equation (4.4.1). According to this property, multiplication of a sequence $x(n)$ by $e^{j\omega_0 n}$ is equivalent to a frequency translation of the spectrum $X(\omega)$ by $\omega_0$. This frequency translation is illustrated in Fig 4.4.8. Since the spectrum $X(\omega)$ is periodic, the shift $\omega_0$ applies to the spectrum of the signal in every period.

**The modulation theorem.** If

$$x(n) \overset{F}{\longleftrightarrow} X(\omega)$$

then

$$x(n) \cos \omega_0 n \overset{F}{\longleftrightarrow} \frac{1}{2}[X(\omega + \omega_0) + X(\omega - \omega_0)] \tag{4.4.53}$$

To prove the modulation theorem, we first express the signal $\cos \omega_0 n$ as

$$\cos \omega_0 n = \frac{1}{2}(e^{j\omega_0 n} + e^{-j\omega_0 n})$$

Upon multiplying $x(n)$ by these two exponentials and using the frequency-shifting property described in the preceding section, we obtain the desired result in (4.4.53).

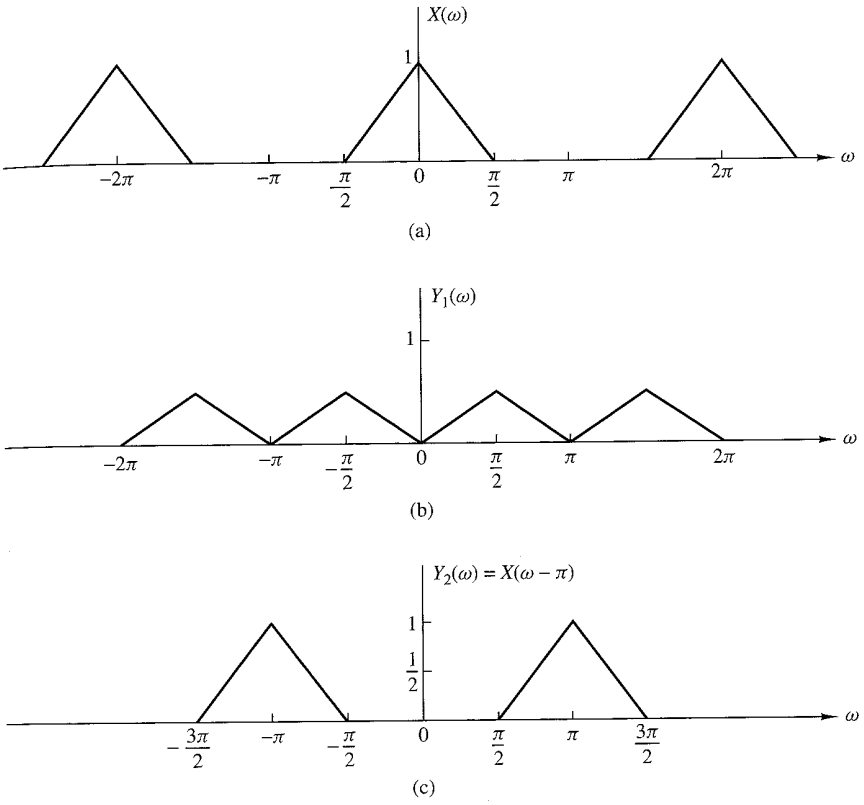



**Figure 4.4.9** Graphical representation of the modulation theorem.

Although the property given in (4.4.52) can also be viewed as (complex) modulation, in practice we prefer to use (4.4.53) because the signal $x(n)\cos\omega_0 n$ is real. Clearly, in this case the symmetry properties (4.4.12) and (4.4.13) are preserved.

The modulation theorem is illustrated in Fig 4.4.9, which contains a plot of the spectra of the signals $x(n)$, $y_1(n) = x(n)\cos 0.5\pi n$ and $y_2(n) = x(n)\cos \pi n$.

**Parseval's theorem.** If

$$x_1(n) \overset{F}{\longleftrightarrow} X_1(\omega)$$

and

$$x_2(n) \overset{F}{\longleftrightarrow} X_2(\omega)$$

then

$$\sum_{n=-\infty}^{\infty} x_1(n)x_2^*(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} X_1(\omega)X_2^*(\omega)\,d\omega \tag{4.4.54}$$

To prove this theorem, we use (4.4.1) to eliminate $X_1(\omega)$ on the right-hand side of (4.4.54). Thus we have

$$\frac{1}{2\pi}\int_{2\pi}\left[\sum_{n=-\infty}^{\infty} x_1(n)e^{-j\omega n}\right] X_2^*(\omega)\, d\omega$$

$$= \sum_{n=-\infty}^{\infty} x_1(n)\frac{1}{2\pi}\int_{2\pi} X_2^*(\omega)e^{-j\omega n}\, d\omega = \sum_{n=-\infty}^{\infty} x_1(n)x_2^*(n)$$

In the special case where $x_2(n) = x_1(n) = x(n)$, Parseval's relation (4.4.54) reduces to

$$\sum_{n=-\infty}^{\infty} |x(n)|^2 = \frac{1}{2\pi}\int_{2\pi} |X(\omega)|^2\, d\omega \tag{4.4.55}$$

We observe that the left-hand side of (4.4.55) is simply the energy $E_x$ of the signal $x(n)$. It is also equal to the autocorrelation of $x(n)$, $r_{xx}(l)$, evaluated at $l = 0$. The integrand in the right-hand side of (4.4.55) is equal to the energy density spectrum, so the integral over the interval $-\pi \le \omega \le \pi$ yields the total signal energy. Therefore, we conclude that

$$E_x = r_{xx}(0) = \sum_{n=-\infty}^{\infty} |x(n)|^2 = \frac{1}{2\pi}\int_{2\pi} |X(\omega)|^2\, d\omega = \frac{1}{2\pi}\int_{-\pi}^{\pi} S_{xx}(\omega)\, d\omega \tag{4.4.56}$$

**Multiplication of two sequences (Windowing theorem).** If

$$x_1(n) \overset{F}{\longleftrightarrow} X_1(\omega)$$

and

$$x_2(n) \overset{F}{\longleftrightarrow} X_2(\omega)$$

then

$$x_3(n) \equiv x_1(n)x_2(n) \overset{F}{\longleftrightarrow} X_3(\omega) = \frac{1}{2\pi}\int_{-\pi}^{\pi} X_1(\lambda)X_2(\omega-\lambda)d\lambda \tag{4.4.57}$$

The integral on the right-hand side of (4.4.57) represents the convolution of the Fourier transforms $X_1(\omega)$ and $X_2(\omega)$. This relation is the dual of the time-domain convolution. In other words, the multiplication of two time-domain sequences is equivalent to the convolution of their Fourier transforms. On the other hand, the convolution of two time-domain sequences is equivalent to the multiplication of their Fourier transforms.

To prove (4.4.57) we begin with the Fourier transform of $x_3(n) = x_1(n)x_2(n)$ and use the formula for the inverse transform, namely,

$$x_1(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} X_1(\lambda)e^{j\lambda n}d\lambda$$

Thus, we have

$$\begin{aligned}
X_3(\omega) &= \sum_{n=-\infty}^{\infty} x_3(n)e^{-j\omega n} = \sum_{n=-\infty}^{\infty} x_1(n)x_2(n)e^{-j\omega n} \\
&= \sum_{n=-\infty}^{\infty}\left[\frac{1}{2\pi}\int_{-\pi}^{\pi} X_1(\lambda)e^{j\lambda n}d\lambda\right]x_2(n)e^{-j\omega n} \\
&= \frac{1}{2\pi}\int_{-\pi}^{\pi} X_1(\lambda)d\lambda\left[\sum_{n=-\infty}^{\infty} x_2(n)e^{-j(\omega-\lambda)n}\right] \\
&= \frac{1}{2\pi}\int_{-\pi}^{\pi} X_1(\lambda)X_2(\omega-\lambda)d\lambda
\end{aligned}$$

The convolution integral in (4.4.57) is known as the *periodic convolution* of $X_1(\omega)$ and $X_2(\omega)$ because it is the convolution of two periodic functions having the same period. We note that the limits of integration extend over a single period. Furthermore, we note that due to the periodicity of the Fourier transform for discrete-time signals, there is no "perfect" duality between the time and frequency domains with respect to the convolution operation, as in the case of continuous-time signals. Indeed, convolution in the time domain (aperiodic summation) is equivalent to multiplication of continuous periodic Fourier transforms. However, multiplication of aperiodic sequences is equivalent to periodic convolution of their Fourier transforms.

The Fourier transform pair in (4.4.57) will prove useful in our treatment of FIR filter design based on the window technique.

**Differentiation in the frequency domain.** If

$$x(n) \overset{F}{\longleftrightarrow} X(\omega)$$

then

$$nx(n) \overset{F}{\longleftrightarrow} j\frac{dX(\omega)}{d\omega} \tag{4.4.58}$$

To prove this property, we use the definition of the Fourier transform in (4.4.1) and differentiate the series term by term with respect to $\omega$. Thus we obtain

$$\begin{aligned}\frac{dX(\omega)}{d\omega} &= \frac{d}{d\omega}\left[\sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n}\right] \\ &= \sum_{n=-\infty}^{\infty} x(n)\frac{d}{d\omega}e^{-j\omega n} \\ &= -j\sum_{n=-\infty}^{\infty} nx(n)e^{-j\omega n}\end{aligned}$$

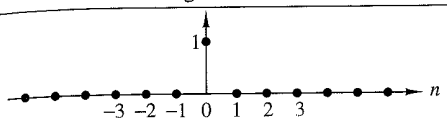
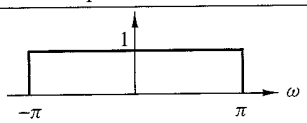
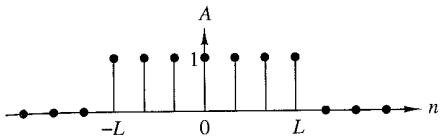
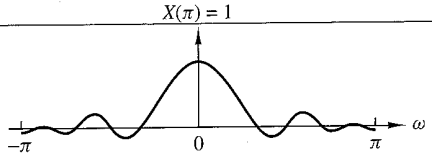
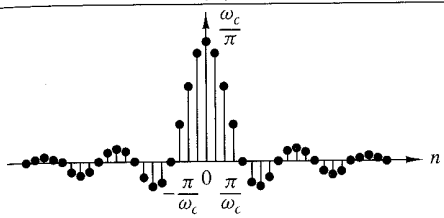
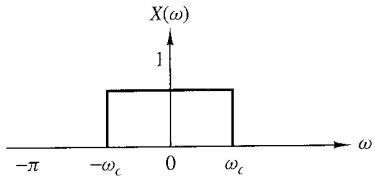
Now we multiply both sides of the equation by $j$ to obtain the desired result in (4.4.58).

The properties derived in this section are summarized in Table 4.5, which serves as a convenient reference. Table 4.6 illustrates some useful Fourier transform pairs that will be encountered in later chapters.

**TABLE 4.5** Properties of the Fourier Transform for Discrete-Time Signals

| Property | Time Domain | Frequency Domain |
|---|---|---|
| Notation | $x(n)$ | $X(\omega)$ |
| | $x_1(n)$ | $X_1(\omega)$ |
| | $x_2(n)$ | $X_2(\omega)$ |
| Linearity | $a_1x_1(n) + a_2x_2(n)$ | $a_1X_1(\omega) + a_2X_2(\omega)$ |
| Time shifting | $x(n-k)$ | $e^{-j\omega k}X(\omega)$ |
| Time reversal | $x(-n)$ | $X(-\omega)$ |
| Convolution | $x_1(n) * x_2(n)$ | $X_1(\omega)X_2(\omega)$ |
| Correlation | $r_{x_1x_2}(l) = x_1(l) * x_2(-l)$ | $S_{x_1x_2}(\omega) = X_1(\omega)X_2(-\omega)$ $= X_1(\omega)X_2^*(\omega)$ [if $x_2(n)$ is real] |
| Wiener–Khintchine theorem | $r_{xx}(l)$ | $S_{xx}(\omega)$ |
| Frequency shifting | $e^{j\omega_0 n}x(n)$ | $X(\omega-\omega_0)$ |
| Modulation | $x(n)\cos\omega_0 n$ | $\frac{1}{2}X(\omega+\omega_0) + \frac{1}{2}X(\omega-\omega_0)$ |
| Multiplication | $x_1(n)x_2(n)$ | $\frac{1}{2\pi}\int_{-\pi}^{\pi} X_1(\lambda)X_2(\omega-\lambda)d\lambda$ |
| Differentiation in the frequency domain | $nx(n)$ | $j\frac{dX(\omega)}{d\omega}$ |
| Conjugation | $x^*(n)$ | $X^*(-\omega)$ |
| Parseval's theorem | $\sum_{n=-\infty}^{\infty} x_1(n)x_2^*(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} X_1(\omega)X_2^*(\omega)d\omega$ | |

**TABLE 4.6** Some Useful Fourier Transform Pairs for Discrete-Time Aperiodic Signals

| Signal $x(n)$ | Spectrum $X(\omega)$ |
|---|---|
|   $x(n) = \delta(n)$ |   $X(\pi) = 1$ |
|   $x(n) = \begin{cases} A, & \lvert n\rvert \le L \\ 0, & \lvert n\rvert > L \end{cases}$ |   $X(\omega) = A\,\dfrac{\sin\left(L+\frac{1}{2}\right)\omega}{\sin\frac{\omega}{2}}$ |
|   $x(n) = \begin{cases} \dfrac{\omega_c}{\pi}, & n = 0 \\ \dfrac{\sin \omega_c n}{\pi n}, & n \ne 0 \end{cases}$ |   $X(\omega) = \begin{cases} 1, & \lvert\omega\rvert < \omega_c \\ 0, & \omega_c \le \lvert\omega\rvert \le \pi \end{cases}$ |
| $x(n) = \begin{cases} a^n, & n \ge 0 \\ 0, & n > 0 \end{cases}$ | $X(\omega) = \frac{1}{1-ae^{-j\omega}}$ |

## 4.5 Summary and References

The Fourier series and the Fourier transform are the mathematical tools for analyzing the characteristics of signals in the frequency domain. The Fourier series is appropriate for representing a periodic signal as a weighted sum of harmonically related sinusoidal components, where the weighting coefficients represent the strengths of each of the harmonics, and the magnitude squared of each weighting coefficient represents the power of the corresponding harmonic. As we have indicated, the Fourier series is one of many possible orthogonal series expansions for a periodic signal. Its importance stems from the characteristic behavior of LTI systems, as we shall see in Chapter 5.

The Fourier transform is appropriate for representing the spectral characteristics of aperiodic signals with finite energy. The important properties of the Fourier transform were also presented in this chapter.

There are many excellent texts on Fourier series and Fourier transforms. For reference, we include the texts by Bracewell (1978), Davis (1963), Dym and McKean (1972), and Papoulis (1962).

## Problems

**4.1** Consider the full-wave rectified sinusoid in Fig. P4.1.

**(a)** Determine its spectrum $X_a(F)$.

**(b)** Compute the power of the signal.

**(c)** Plot the power spectral density.

**(d)** Check the validity of Parseval's relation for this signal.

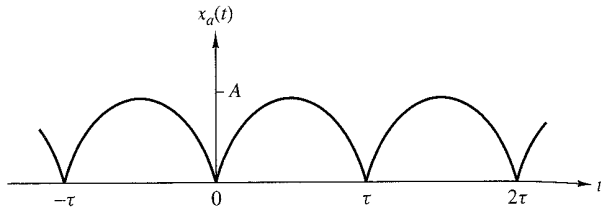



**Figure P4.1**

**4.2** Compute and sketch the magnitude and phase spectra for the following signals ($a > 0$).

**(a)** $x_a(t) = \begin{cases} Ae^{-at}, & t \geq 0 \\ 0, & t < 0 \end{cases}$

**(b)** $x_a(t) = Ae^{-a|t|}$

**4.3** Consider the signal

$$x(t) = \begin{cases} 1 - |t|/\tau, & |t| \leq \tau \\ 0, & \text{elsewhere} \end{cases}$$

**(a)** Determine and sketch its magnitude and phase spectra, $|X_a(F)|$ and $\measuredangle X_a(F)$, respectively.

**(b)** Create a periodic signal $x_p(t)$ with fundamental period $T_p \geq 2\tau$, so that $x(t) = x_p(t)$ for $|t| < T_p/2$. What are the Fourier coefficients $c_k$ for the signal $x_p(t)$?

**(c)** Using the results in parts (a) and (b), show that $c_k = (1/T_p)X_a(k/T_p)$.

**4.4** Consider the following periodic signal:

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

**(a)** Sketch the signal $x(n)$ and its magnitude and phase spectra.

**(b)** Using the results in part (a), verify Parseval's relation by computing the power in the time and frequency domains.

**4.5** Consider the signal

$$x(n) = 2 + 2\cos\frac{\pi n}{4} + \cos\frac{\pi n}{2} + \frac{1}{2}\cos\frac{3\pi n}{4}$$

**(a)** Determine and sketch its power density spectrum.

**(b)** Evaluate the power of the signal.

**4.6** Determine and sketch the magnitude and phase spectra of the following periodic signals.

**(a)** $x(n) = 4\sin\frac{\pi(n-2)}{3}$

**(b)** $x(n) = \cos\frac{2\pi}{3}n + \sin\frac{2\pi}{5}n$

**(c)** $x(n) = \cos\frac{2\pi}{3}n \sin\frac{2\pi}{5}n$

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

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

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

**(g)** $x(n) = 1, -\infty < n < \infty$

**(h)** $x(n) = (-1)^n, -\infty < n < \infty$

**4.7** Determine the periodic signals $x(n)$, with fundamental period $N = 8$, if their Fourier coefficients are given by:

**(a)** $c_k = \cos\frac{k\pi}{4} + \sin\frac{3k\pi}{4}$

**(b)** $c_k = \begin{cases} \sin\frac{k\pi}{3}, & 0 \le k \le 6 \\ 0, & k = 7 \end{cases}$

**(c)** $\{c_k\} = \{\ldots, 0, \frac{1}{4}, \frac{1}{2}, 1, \underset{\uparrow}{2}, 1, \frac{1}{2}, \frac{1}{4}, 0 \ldots\}$

**4.8** Two DT signals, $s_k(n)$ and $s_l(n)$, are said to be orthogonal over an interval $[N_1, N_2]$ if

$$\sum_{n=N_1}^{N_2} s_k(n)s_l^*(n) = \begin{cases} A_k, & k = l \\ 0, & k \neq l \end{cases}$$

If $A_k = 1$, the signals are called orthonormal.

**(a)** Prove the relation

$$\sum_{n=0}^{N-1} e^{j2\pi kn/N} = \begin{cases} N, & k = 0, \pm N, \pm 2N, \ldots \\ 0, & \text{otherwise} \end{cases}$$

**(b)** Illustrate the validity of the relation in part (a) by plotting for every value of $k = 1, 2, \ldots, 6$, the signals $s_k(n) = e^{j(2\pi/6)kn}$, $n = 0, 1, \ldots, 5$. [*Note*: For a given $k$, $n$ the signal $s_k(n)$ can be represented as a vector in the complex plane.]

**(c)** Show that the harmonically related signals

$$s_k(n) = e^{j(2\pi/N)kn}$$

are orthogonal over any interval of length $N$.

**4.9** Compute the Fourier transform of the following signals.

**(a)** $x(n) = u(n) - u(n-6)$

**(b)** $x(n) = 2^n u(-n)$

**(c)** $x(n) = (\frac{1}{4})^n u(n+4)$

**(d)** $x(n) = (\alpha^n \sin \omega_0 n) u(n), \qquad |\alpha| < 1$

**(e)** $x(n) = |\alpha|^n \sin \omega_0 n, \qquad |\alpha| < 1$

**(f)** $x(n) = \begin{cases} 2 - (\frac{1}{2})n, & |n| \le 4 \\ 0, & \text{elsewhere} \end{cases}$

**(g)** $x(n) = \{-2, -1, \underset{\uparrow}{0}, 1, 2\}$

**(h)** $x(n) = \begin{cases} A(2M + 1 - |n|), & |n| \le M \\ 0, & |n| > M \end{cases}$

Sketch the magnitude and phase spectra for parts (a), (f), and (g).

**4.10** Determine the signals having the following Fourier transforms.

**(a)** $X(\omega) = \begin{cases} 0, & 0 \le |\omega| \le \omega_0 \\ 1, & \omega_0 < |\omega| \le \pi \end{cases}$

**(b)** $X(\omega) = \cos^2 \omega$

**(c)** $X(\omega) = \begin{cases} 1, & \omega_0 - \delta\omega/2 \le |\omega| \le \omega_0 + \delta\omega/2 \\ 0, & \text{elsewhere} \end{cases}$

**(d)** The signal shown in Fig. P4.10.

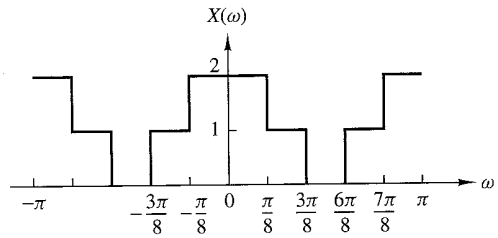



**Figure P4.10**

**4.11** Consider the signal

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

with Fourier transform $X(\omega) = X_R(\omega) + j(X_I(\omega))$. Determine and sketch the signal $y(n)$ with Fourier transform

$$Y(\omega) = X_I(\omega) + X_R(\omega)e^{j2\omega}$$

**4.12** Determine the signal $x(n)$ if its Fourier transform is as given in Fig. P4.12.

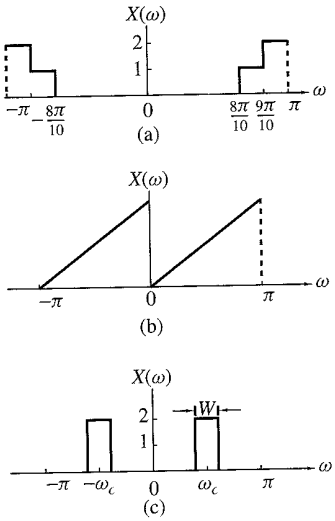



**Figure P4.12**

**4.13** In Example 4.4.2, the Fourier transform of the signal

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

was shown to be

$$X(\omega) = 1 + 2\sum_{n=1}^{M} \cos \omega n$$

Show that the Fourier transform of

$$x_1(n) = \begin{cases} 1, & 0 \le n \le M \\ 0, & \text{otherwise} \end{cases}$$

and

$$x_2(n) = \begin{cases} 1, & -M \le n \le -1 \\ 0, & \text{otherwise} \end{cases}$$

are, respectively,

$$X_1(\omega) = \frac{1 - e^{-j\omega(M+1)}}{1 - e^{-j\omega}}$$

$$X_2(\omega) = \frac{e^{j\omega} - e^{j\omega(M+1)}}{1 - e^{j\omega}}$$

Thus prove that

$$X(\omega) = X_1(\omega) + X_2(\omega)$$
$$= \frac{\sin(M + \frac{1}{2})\omega}{\sin(\omega/2)}$$

and therefore,

$$1 + 2\sum_{n=1}^{M} \cos \omega n = \frac{\sin(M + \frac{1}{2})\omega}{\sin(\omega/2)}$$

**4.14** Consider the signal

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

with Fourier transform $X(\omega)$. Compute the following quantities, without explicitly computing $X(\omega)$:

**(a)** $X(0)$

**(b)** $\angle X(\omega)$

**(c)** $\int_{-\pi}^{\pi} X(\omega)\, d\omega$

**(d)** $X(\pi)$

**(e)** $\int_{-\pi}^{\pi} |X(\omega)|^2\, d\omega$

**4.15** The center of gravity of a signal $x(n)$ is defined as

$$c = \frac{\sum_{n=-\infty}^{\infty} n x(n)}{\sum_{n=-\infty}^{\infty} x(n)}$$

and provides a measure of the "time delay" of the signal.

**(a)** Express $c$ in terms of $X(\omega)$.

**(b)** Compute $c$ for the signal $x(n)$ whose Fourier transform is shown in Fig. P4.15.

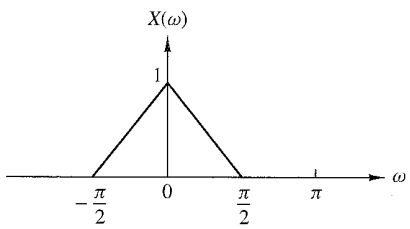



**Figure P4.15**

**4.16** Consider the Fourier transform pair

$$a^n u(n) \overset{F}{\longleftrightarrow} \frac{1}{1 - ae^{-j\omega}}, \qquad |a| < 1$$

Use the differentiation in frequency theorem and induction to show that

$$x(n) = \frac{(n + l - 1)!}{n!(l - 1)!} a^n u(n) \overset{F}{\longleftrightarrow} X(\omega) = \frac{1}{(1 - ae^{-j\omega})^l}$$

**4.17** Let $x(n)$ be an arbitrary signal, not necessarily real valued, with Fourier transform $X(\omega)$. Express the Fourier transforms of the following signals in terms of $X(\omega)$.

**(a)** $x^*(n)$

**(b)** $x^*(-n)$

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

**(d)** $y(n) = \sum_{k=-\infty}^{n} x(k)$

**(e)** $y(n) = x(2n)$

**(f)** $y(n) = \begin{cases} x(n/2), & n \text{ even} \\ 0, & n \text{ odd} \end{cases}$

**4.18** Determine and sketch the Fourier transforms $X_1(\omega)$, $X_2(\omega)$, and $X_3(\omega)$ of the following signals.

**(a)** $x_1(n) = \{1, 1, \underset{\uparrow}{1}, 1, 1\}$

**(b)** $x_2(n) = \{1, 0, 1, 0, \underset{\uparrow}{1}, 0, 1, 0, 1\}$

**(c)** $x_3(n) = \{1, 0, 0, 1, 0, 0, \underset{\uparrow}{1}, 0, 0, 1, 0, 0, 1\}$

**(d)** Is there any relation between $X_1(\omega)$, $X_2(\omega)$, and $X_3(\omega)$? What is its physical meaning?

**(e)** Show that if

$$x_k(n) = \begin{cases} x\left(\dfrac{n}{k}\right), & \text{if } n/k \text{ integer} \\ 0, & \text{otherwise} \end{cases}$$

then

$$X_k(\omega) = X(k\omega)$$

**4.19** Let $x(n)$ be a signal with Fourier transform as shown in Fig. P4.19. Determine and sketch the Fourier transforms of the following signals.

**(a)** $x_1(n) = x(n)\cos(\pi n/4)$

**(b)** $x_2(n) = x(n)\sin(\pi n/2)$

**(c)** $x_3(n) = x(n)\cos(\pi n/2)$

**(d)** $x_4(n) = x(n)\cos \pi n$

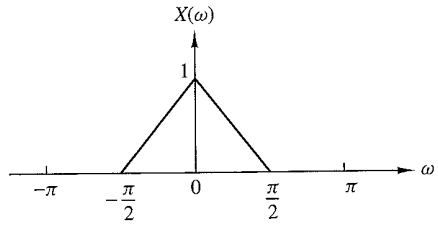



**Figure P4.19**

Note that these signal sequences are obtained by *amplitude modulation* of a carrier $\cos \omega_c n$ or $\sin \omega_c n$ by the sequence $x(n)$.

**4.20** Consider an aperiodic signal $x(n)$ with Fourier transform $X(\omega)$. Show that the Fourier series coefficients $C_k^y$ of the periodic signal

$$y(n) = \sum_{l=-\infty}^{\infty} x(n - lN)$$

are given by

$$C_k^y = \frac{1}{N} X\left(\frac{2\pi}{N} k\right), \qquad k = 0, 1, \ldots, N-1$$

**4.21** Prove that

$$X_N(\omega) = \sum_{n=-N}^{N} \frac{\sin \omega_c n}{\pi n} e^{-j\omega n}$$

may be expressed as

$$X_N(\omega) = \frac{1}{2\pi} \int^{\omega_c} -\omega_c \frac{\sin[(2N+1)(\omega - \theta/2)]}{sin[(\omega - \theta)/2]} d\theta$$

**4.22** A signal $x(n)$ has the following Fourier transform:

$$X(\omega) = \frac{1}{1 - ae^{-j\omega}}$$

Determine the Fourier transforms of the following signals:

**(a)** $x(2n+1)$

**(b)** $e^{\pi n/2} x(n+2)$

**(c)** $x(-2n)$

**(d)** $x(n)\cos(0.3\pi n)$

**(e)** $x(n) * x(n-1)$

**(f)** $x(n) * x(-n)$

**4.23** From a discrete-time signal $x(n)$ with Fourier transform $X(\omega)$, shown in Fig. P4.23, determine and sketch the Fourier transform of the following signals:

**(a)** $y_1(n) = \begin{cases} x(n), & n \text{ even} \\ 0, & n \text{ odd} \end{cases}$

**(b)** $y_2(n) = x(2n)$

**(c)** $y_3(n) = \begin{cases} x(n/2), & n \text{ even} \\ 0, & n \text{ odd} \end{cases}$

Note that $y_1(n) = x(n)s(n)$, where $s(n) = \{\ldots 0, 1, 0, \underset{\uparrow}{1}, 0, 1, 0, 1, \ldots\}$

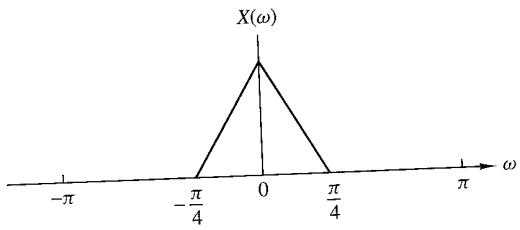



**Figure P4.23**

ElCoM
Electrical | Computer | Mechatronics
ELCOM-HU.COM

CHAPTER 5

# Frequency-Domain Analysis of LTI Systems

In this chapter we treat the characterization of linear time-invariant systems in the frequency domain. The basic excitation signals in this development are the complex exponentials and sinusoidal functions. We will observe that an LTI system performs a discrimination or filtering on the various frequency components at its input. This observation leads us to characterize and classify some simple LTI systems according to the type of filtering they perform on any input signal. The design of these simple filters is described and some applications are given.

We also develop frequency-domain relationships between the spectra of the input and output sequences of an LTI system. The final section of this chapter is focused on the application of LTI systems for performing inverse filtering and deconvolution.

## 5.1 Frequency-Domain Characteristics of Linear Time-Invariant Systems

In this section we develop the characterization of linear time-invariant systems in the frequency domain. The basic excitation signals in this development are the complex exponentials and sinusoidal functions. The characteristics of the system are described by a function of the frequency variable $\omega$ called the frequency response, which is the Fourier transform of the impulse response $h(n)$ of the system.

The frequency response function completely characterizes a linear time-invariant system in the frequency domain. This allows us to determine the steady-state response of the system to any arbitrary weighted linear combination of sinusoids or complex exponentials. Since periodic sequences, in particular, lend themselves to a Fourier series decomposition as a weighted sum of harmonically related complex

exponentials, it becomes a simple matter to determine the response of a linear time-invariant system to this class of signals. This methodology is also applied to aperiodic signals since such signals can be viewed as a superposition of infinitesimal size complex exponentials.

### 5.1.1 Response to Complex Exponential and Sinusoidal Signals: The Frequency Response Function

In Chapter 2, it was demonstrated that the response of any relaxed linear time-invariant system to an arbitrary input signal $x(n)$ is given by the convolution sum formula

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k) \tag{5.1.1}$$

In this input–output relationship, the system is characterized in the time domain by its unit sample response $\{h(n), -\infty < n < \infty\}$.

To develop a frequency-domain characterization of the system, let us excite the system with the complex exponential

$$x(n) = Ae^{j\omega n}, \qquad -\infty < n < \infty \tag{5.1.2}$$

where $A$ is the amplitude and $\omega$ is any arbitrary frequency confined to the frequency interval $[-\pi, \pi]$. By substituting (5.1.2) into (5.1.1), we obtain the response

$$\begin{aligned} y(n) &= \sum_{k=-\infty}^{\infty} h(k)[Ae^{j\omega(n-k)}] \\ &= A\left[\sum_{k=-\infty}^{\infty} h(k)e^{-j\omega k}\right]e^{j\omega n} \end{aligned} \tag{5.1.3}$$

We observe that the term in brackets in (5.1.3) is a function of the frequency variable $\omega$. In fact, this term is the Fourier transform of the unit sample response $h(k)$ of the system. Hence we denote this function as

$$H(\omega) = \sum_{k=-\infty}^{\infty} h(k)e^{-j\omega k} \tag{5.1.4}$$

Clearly, the function $H(\omega)$ exists if the system is BIBO stable, that is, if

$$\sum_{n=-\infty}^{\infty} |h(n)| < \infty$$

With the definition in (5.1.4), the response of the system to the complex exponential given in (5.1.2) is

$$y(n) = AH(\omega)e^{j\omega n} \tag{5.1.5}$$

We note that the response is also in the form of a complex exponential with the same frequency as the input, but altered by the multiplicative factor $H(\omega)$.

As a result of this characteristic behavior, the exponential signal in (5.1.2) is called an *eigenfunction* of the system. In other words, an eigenfunction of a system is an input signal that produces an output that differs from the input by a constant multiplicative factor. The multiplicative factor is called an *eigenvalue* of the system. In this case, a complex exponential signal of the form (5.1.2) is an eigenfunction of a linear time-invariant system, and $H(\omega)$ evaluated at the frequency of the input signal is the corresponding eigenvalue.

### EXAMPLE 5.1.1

Determine the output sequence of the system with impulse response

$$h(n) = (\frac{1}{2})^n u(n) \tag{5.1.6}$$

when the input is the complex exponential sequence

$$x(n) = Ae^{j\pi n/2}, \qquad -\infty < n < \infty$$

**Solution.** First we evaluate the Fourier transform of the impulse response $h(n)$, and then we use (5.1.5) to determine $y(n)$. From Example 4.2.3 we recall that

$$H(\omega) = \sum_{n=-\infty}^{\infty} h(n)e^{-j\omega n} = \frac{1}{1 - \frac{1}{2}e^{-j\omega}} \tag{5.1.7}$$

At $\omega = \pi/2$, (5.1.7) yields

$$H\left(\frac{\pi}{2}\right) = \frac{1}{1 + j\frac{1}{2}} = \frac{2}{\sqrt{5}}e^{-j26.6^\circ}$$

and therefore the output is

$$\begin{aligned} y(n) &= A\left(\frac{2}{\sqrt{5}}e^{-j26.6^\circ}\right)e^{j\pi n/2} \\ y(n) &= \frac{2}{\sqrt{5}}Ae^{j(\pi n/2 - 26.6^\circ)}, \qquad -\infty < n < \infty \end{aligned} \tag{5.1.8}$$

---

This example clearly illustrates that the only effect of the system on the input signal is to scale the amplitude by $2/\sqrt{5}$ and shift the phase by $-26.6^\circ$. Thus the output is also a complex exponential of frequency $\pi/2$, amplitude $2A/\sqrt{5}$, and phase $-26.6^\circ$.

If we alter the frequency of the input signal, the effect of the system on the input also changes and hence the output changes. In particular, if the input sequence is a complex exponential of frequency $\pi$, that is,

$$x(n) = Ae^{j\pi n}, \qquad -\infty < n < \infty \tag{5.1.9}$$

then, at $\omega = \pi$,

$$H(\pi) = \frac{1}{1 - \frac{1}{2}e^{-j\pi}} = \frac{1}{\frac{3}{2}} = \frac{2}{3}$$

and the output of the system is

$$y(n) = \frac{2}{3}Ae^{j\pi n}, \qquad -\infty < n < \infty \tag{5.1.10}$$

We note that $H(\pi)$ is purely real [i.e., the phase associated with $H(\omega)$ is zero at $\omega = \pi$]. Hence, the input is scaled in amplitude by the factor $H(\pi) = \frac{2}{3}$, but the phase shift is zero.

In general, $H(\omega)$ is a complex-valued function of the frequency variable $\omega$. Hence it can be expressed in polar form as

$$H(\omega) = |H(\omega)|e^{j\Theta(\omega)} \tag{5.1.11}$$

where $|H(\omega)|$ is the magnitude of $H(\omega)$ and

$$\Theta(\omega) = \measuredangle H(\omega)$$

which is the phase shift imparted on the input signal by the system at the frequency $\omega$.

Since $H(\omega)$ is the Fourier transform of $\{h(k)\}$, it follows that $H(\omega)$ is a periodic function with period $2\pi$. Furthermore, we can view (5.1.4) as the exponential Fourier series expansion for $H(\omega)$, with $h(k)$ as the Fourier series coefficients. Consequently, the unit impulse $h(k)$ is related to $H(\omega)$ through the integral expression

$$h(k) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H(\omega)e^{j\omega k}d\omega \tag{5.1.12}$$

For a linear time-invariant system with a real-valued impulse response, the magnitude and phase functions possess symmetry properties which are developed as follows. From the definition of $H(\omega)$, we have

$$\begin{aligned} H(\omega) &= \sum_{k=-\infty}^{\infty} h(k)e^{-j\omega k} \\ &= \sum_{k=-\infty}^{\infty} h(k)\cos\omega k - j\sum_{k=-\infty}^{\infty} h(k)\sin\omega k \\ &= H_R(\omega) + jH_I(\omega) \\ &= \sqrt{H_R^2(\omega) + H_I^2(\omega)}\,e^{j\tan^{-1}[H_I(\omega)/H_R(\omega)]} \end{aligned} \tag{5.1.13}$$

where $H_R(\omega)$ and $H_I(\omega)$ denote the real and imaginary components of $H(\omega)$, defined as

$$H_R(\omega) = \sum_{k=-\infty}^{\infty} h(k)\cos\omega k$$
$$H_I(\omega) = -\sum_{k=-\infty}^{\infty} h(k)\sin\omega k \tag{5.1.14}$$

It is clear from (5.1.12) that the magnitude and phase of $H(\omega)$, expressed in terms of $H_R(\omega)$ and $H_I(\omega)$, are

$$|H(\omega)| = \sqrt{H_R^2(\omega) + H_I^2(\omega)}$$
$$\Theta(\omega) = \tan^{-1}\frac{H_I(\omega)}{H_R(\omega)} \tag{5.1.15}$$

We note that $H_R(\omega) = H_R(-\omega)$ and $H_I(\omega) = -H_I(-\omega)$, so that $H_R(\omega)$ is an even function of $\omega$ and $H_I(\omega)$ is an odd function of $\omega$. As a consequence, it follows that $|H(\omega)|$ is an even function of $\omega$ and $\Theta(\omega)$ is an odd function of $\omega$. Hence, if we know $|H(\omega)|$ and $\Theta(\omega)$ for $0 \le \omega \le \pi$, we also know these functions for $-\pi \le \omega \le 0$.

### EXAMPLE 5.1.2 Moving Average Filter

Determine the magnitude and phase of $H(\omega)$ for the three-point moving average (MA) system

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

and plot these two functions for $0 \le \omega \le \pi$.

**Solution.** Since

$$h(n) = \{\tfrac{1}{3}, \underset{\uparrow}{\tfrac{1}{3}}, \tfrac{1}{3}\}$$

it follows that

$$H(\omega) = \frac{1}{3}(e^{j\omega} + 1 + e^{-j\omega}) = \frac{1}{3}(1 + 2\cos\omega)$$

Hence

$$|H(\omega)| = \frac{1}{3}|1 + 2\cos\omega| \tag{5.1.16}$$
$$\Theta(\omega) = \begin{cases} 0, & 0 \le \omega \le 2\pi/3 \\ \pi, & 2\pi/3 \le \omega < \pi \end{cases}$$

Figure 5.1.1 illustrates the graphs of the magnitude and phase of $H(\omega)$. As indicated previously, $|H(\omega)|$ is an even function of frequency and $\Theta(\omega)$ is an odd function of frequency. It is apparent from the frequency response characteristic $H(\omega)$ that this moving average filter smooths the input data, as we would expect from the input–output equation.

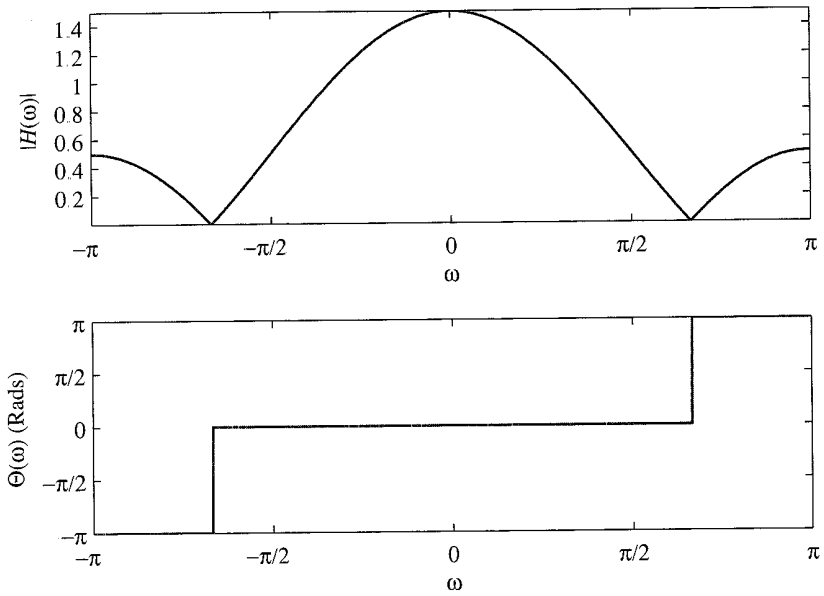



**Figure 5.1.1** Magnitude and phase responses for the MA system in Example 5.1.2.

---

The symmetry properties satisfied by the magnitude and phase functions of $H(\omega)$, and the fact that a sinusoid can be expressed as a sum or difference of two complex-conjugate exponential functions, imply that the response of a linear time-invariant system to a sinusoid is similar in form to the response when the input is a complex exponential. Indeed, if the input is

$$x_1(n) = Ae^{j\omega n}$$

the output is

$$y_1(n) = A|H(\omega)|e^{j\Theta(\omega)}e^{j\omega n}$$

On the other hand, if the input is

$$x_2(n) = Ae^{-j\omega n}$$

the response of the system is

$$\begin{aligned} y_2(n) &= A|H(-\omega)|e^{j\Theta(-\omega)}e^{-j\omega n} \\ &= A|H(\omega)|e^{-j\Theta(\omega)}e^{-j\omega n} \end{aligned}$$

where, in the last expression, we have made use of the symmetry properties $|H(\omega)| = |H(-\omega)|$ and $\Theta(\omega) = -\Theta(-\omega)$. Now, by applying the superposition property of the linear time-invariant system, we find that the response of the system to the input

$$x(n) = \frac{1}{2}[x_1(n) + x_2(n)] = A\cos\omega n$$

is

$$\begin{aligned} y(n) &= \frac{1}{2}[y_1(n) + y_2(n)] \\ y(n) &= A|H(\omega)|\cos[\omega n + \Theta(\omega)] \end{aligned} \tag{5.1.17}$$

Similarly, if the input is

$$x(n) = \frac{1}{j2}[x_1(n) - x_2(n)] = A\sin\omega n$$

the response of the system is

$$\begin{aligned} y(n) &= \frac{1}{j2}[y_1(n) - y_2(n)] \\ y(n) &= A|H(\omega)|\sin[\omega n + \Theta(\omega)] \end{aligned} \tag{5.1.18}$$

It is apparent from this discussion that $H(\omega)$, or equivalently, $|H(\omega)|$ and $\Theta(\omega)$, completely characterize the effect of the system on a sinusoidal input signal of any arbitrary frequency. Indeed, we note that $|H(\omega)|$ determines the amplification ($|H(\omega)| > 1$) or attenuation ($|H(\omega)| < 1$) imparted by the system on the input sinusoid. The phase $\Theta(\omega)$ determines the amount of phase shift imparted by the system on the input sinusoid. Consequently, by knowing $H(\omega)$, we are able to determine the response of the system to any sinusoidal input signal. Since $H(\omega)$ specifies the response of the system in the frequency domain, it is called the *frequency response* of the system. Correspondingly, $|H(\omega)|$ is called the *magnitude response* and $\Theta(\omega)$ is called the *phase response* of the system.

If the input to the system consists of more than one sinusoid, the superposition property of the linear system can be used to determine the response. The following examples illustrate the use of the superposition property.

### EXAMPLE 5.1.3

Determine the response of the system in Example 5.1.1 to the input signal

$$x(n) = 10 - 5\sin\frac{\pi}{2}n + 20\cos\pi n, \qquad -\infty < n < \infty$$

**Solution.** The frequency response of the system is given in (5.1.7) as

$$H(\omega) = \frac{1}{1 - \frac{1}{2}e^{-j\omega}}$$

The first term in the input signal is a fixed signal component corresponding to $\omega = 0$. Thus

$$H(0) = \frac{1}{1 - \frac{1}{2}} = 2$$

The second term in $x(n)$ has a frequency $\pi/2$. At this frequency the frequency response of the system is

$$H\left(\frac{\pi}{2}\right) = \frac{2}{\sqrt{5}} e^{-j26.6^\circ}$$

Finally, the third term in $x(n)$ has a frequency $\omega = \pi$. At this frequency

$$H(\pi) = \frac{2}{3}$$

Hence the response of the system to $x(n)$ is

$$y(n) = 20 - \frac{10}{\sqrt{5}} \sin\left(\frac{\pi}{2}n - 26.6^\circ\right) + \frac{40}{3} \cos \pi n, \qquad -\infty < n < \infty$$

---

## EXAMPLE 5.1.4

A linear time-invariant system is described by the following difference equation:

$$y(n) = ay(n-1) + bx(n), \qquad 0 < a < 1$$

**(a)** Determine the magnitude and phase of the frequency response $H(\omega)$ of the system.

**(b)** Choose the parameter $b$ so that the maximum value of $|H(\omega)|$ is unity, and sketch $|H(\omega)|$ and $\measuredangle H(\omega)$ for $a = 0.9$.

**(c)** Determine the output of the system to the input signal

$$x(n) = 5 + 12 \sin \frac{\pi}{2} n - 20 \cos\left(\pi n + \frac{\pi}{4}\right)$$

**Solution.** The impulse response of the system is

$$h(n) = ba^n u(n)$$

Since $|a| < 1$, the system is BIBO stable and hence $H(\omega)$ exists.

**(a)** The frequency response is

$$H(\omega) = \sum_{n=-\infty}^{\infty} h(n) e^{-j\omega n}$$

$$= \frac{b}{1 - ae^{-j\omega}}$$

Since

$$1 - ae^{-j\omega} = (1 - a\cos\omega) + ja\sin\omega$$

it follows that

$$|1 - ae^{-j\omega}| = \sqrt{(1 - a\cos\omega)^2 + (a\sin\omega)^2}$$

$$= \sqrt{1 + a^2 - 2a\cos\omega}$$

and

$$\measuredangle(1 - ae^{-j\omega}) = \tan^{-1}\frac{a\sin\omega}{1 - a\cos\omega}$$

Therefore,

$$|H(\omega)| = \frac{|b|}{\sqrt{1 + a^2 - 2a\cos\omega}}$$

$$\measuredangle H(\omega) = \Theta(\omega) = \measuredangle b - \tan^{-1}\frac{a\sin\omega}{1 - a\cos\omega}$$

**(b)** Since the parameter $a$ is positive, the denominator of $|H(\omega)|$ attains a minimum at $\omega = 0$. Therefore, $|H(\omega)|$ attains its maximum value at $\omega = 0$. At this frequency we have

$$|H(0)| = \frac{|b|}{1 - a} = 1$$

which implies that $b = \pm(1 - a)$. We choose $b = 1 - a$, so that

$$|H(\omega)| = \frac{1 - a}{\sqrt{1 + a^2 - 2a\cos\omega}}$$

and

$$\Theta(\omega) = -\tan^{-1}\frac{a\sin\omega}{1 - a\cos\omega}$$

The frequency response plots for $|H(\omega)|$ and $\Theta(\omega)$ are illustrated in Fig. 5.1.2. We observe that this system attenuates high-frequency signals.

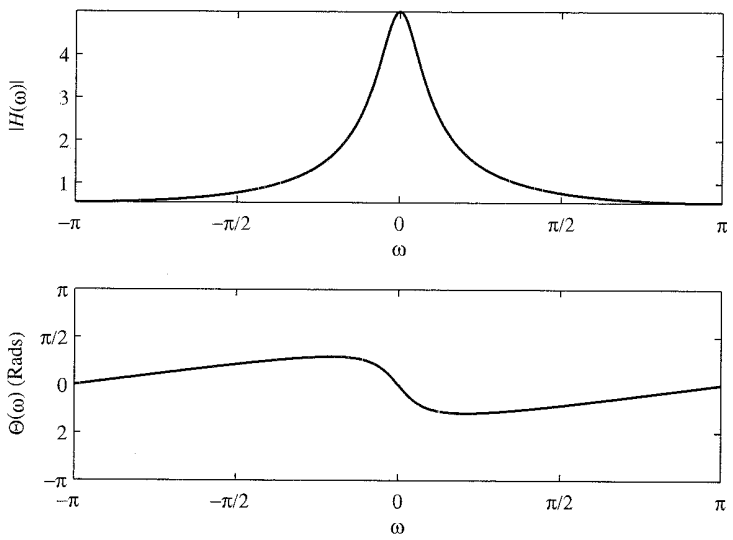



**Figure 5.1.2** Magnitude and phase responses for the system in Example 5.1.4 with $a = 0.9$.

**(c)** The input signal consists of components at frequencies $\omega = 0$, $\pi/2$, and $\pi$. For $\omega = 0$, $|H(0)| = 1$ and $\Theta(0) = 0$. For $\omega = \pi/2$,

$$\left|H\left(\frac{\pi}{2}\right)\right| = \frac{1-a}{\sqrt{1+a^2}} = \frac{0.1}{\sqrt{1.81}} = 0.074$$

$$\Theta\left(\frac{\pi}{2}\right) = -\tan^{-1} a = -42^\circ$$

For $\omega = \pi$,

$$|H(\pi)| = \frac{1-a}{1+a} = \frac{0.1}{1.9} = 0.053$$

$$\Theta(\pi) = 0$$

Therefore, the output of the system is

$$\begin{aligned} y(n) &= 5|H(0)| + 12\left|H\left(\frac{\pi}{2}\right)\right| \sin\left[\frac{\pi}{2}n + \Theta\left(\frac{\pi}{2}\right)\right] \\ &\quad - 20|H(\pi)| \cos\left[\pi n + \frac{\pi}{4} + \Theta(\pi)\right] \\ &= 5 + 0.888 \sin\left(\frac{\pi}{2}n - 42^\circ\right) - 1.06 \cos\left(\pi n + \frac{\pi}{4}\right), \qquad -\infty < n < \infty \end{aligned}$$

---

In the most general case, if the input to the system consists of an arbitrary linear combination of sinusoids of the form

$$x(n) = \sum_{i=1}^{L} A_i \cos(\omega_i n + \phi_i), \qquad -\infty < n < \infty$$

where $\{A_i\}$ and $\{\phi_i\}$ are the amplitudes and phases of the corresponding sinusoidal components, then the response of the system is simply

$$y(n) = \sum_{i=1}^{L} A_i |H(\omega_i)| \cos[\omega_i n + \phi_i + \Theta(\omega_i)] \tag{5.1.19}$$

where $|H(\omega_i)|$ and $\Theta(\omega_i)$ are the magnitude and phase, respectively, imparted by the system to the individual frequency components of the input signal.

It is clear that depending on the frequency response $H(\omega)$ of the system, input sinusoids of different frequencies will be affected differently by the system. For example, some sinusoids may be completely suppressed by the system if $H(\omega) = 0$ at the frequencies of these sinusoids. Other sinusoids may receive no attenuation (or perhaps, some amplification) by the system. In effect, we can view the linear time-invariant system functioning as a *filter* to sinusoids of different frequencies, passing some of the frequency components to the output and suppressing or preventing other frequency components from reaching the output. In fact, as discussed in Chapter 10, the basic digital filter design problem involves determining the parameters of a linear time-invariant system to achieve a desired frequency response $H(\omega)$.

### 5.1.2 Steady-State and Transient Response to Sinusoidal Input Signals

In the discussion in the preceding section, we determined the response of a linear time-invariant system to exponential and sinusoidal input signals applied to the system at $n = -\infty$. We usually call such signals eternal exponentials or eternal sinusoids, because they were applied at $n = -\infty$. In such a case, the response that we observe at the output of the system is the steady-state response. There is no transient response in this case.

On the other hand, if the exponential or sinusoidal signal is applied at some finite time instant, say at $n = 0$, the response of the system consists of two terms, the transient response and the steady-state response. To demonstrate this behavior, let us consider, as an example, the system described by the first-order difference equation

$$y(n) = ay(n-1) + x(n) \tag{5.1.20}$$

This system was considered in Section 2.4.2. Its response to any input $x(n)$ applied at $n = 0$ is given by (2.4.8) as

$$y(n) = a^{n+1}y(-1) + \sum_{k=0}^{n} a^k x(n-k), \qquad n \geq 0 \tag{5.1.21}$$

where $y(-1)$ is the initial condition.

Now, let us assume that the input to the system is the complex exponential

$$x(n) = Ae^{j\omega n}, \qquad n \geq 0 \tag{5.1.22}$$

applied at $n = 0$. When we substitute (5.1.22) into (5.1.21), we obtain

$$\begin{aligned} y(n) &= a^{n+1}y(-1) + A\sum_{k=0}^{n} a^k e^{j\omega(n-k)} \\ &= a^{n+1}y(-1) + A\left[\sum_{k=0}^{n}(ae^{-j\omega})^k\right]e^{j\omega n} \\ &= a^{n+1}y(-1) + A\frac{1-a^{n+1}e^{-j\omega(n+1)}}{1-ae^{-j\omega}}e^{j\omega n}, \qquad n \geq 0 \\ &= a^{n+1}y(-1) - \frac{Aa^{n+1}e^{-j\omega(n+1)}}{1-ae^{-j\omega}}e^{j\omega n} + \frac{A}{1-ae^{-j\omega}}e^{j\omega n}, \qquad n \geq 0 \end{aligned} \tag{5.1.23}$$

We recall that the system in (5.1.20) is BIBO stable if $|a| < 1$. In this case the two terms involving $a^{n+1}$ in (5.1.23) decay toward zero as $n$ approaches infinity. Consequently, we are left with the steady-state response

$$\begin{aligned} y_{ss}(n) &= \lim_{n\to\infty} y(n) = \frac{A}{1-ae^{-j\omega}}e^{j\omega n} \\ &= AH(\omega)e^{j\omega n} \end{aligned} \tag{5.1.24}$$

The first two terms in (5.1.23) constitute the transient response of the system, that is,

$$y_{\text{tr}}(n) = a^{n+1}y(-1) - \frac{Aa^{n+1}e^{-j\omega(n+1)}}{1 - ae^{-j\omega}}e^{j\omega n}, \qquad n \geq 0 \tag{5.1.25}$$

which decay toward zero as $n$ approaches infinity. The first term in the transient response is the zero-input response of the system and the second term is the transient produced by the exponential input signal.

In general, all linear time-invariant BIBO systems behave in a similar fashion when excited by a complex exponential, or by a sinusoid at $n = 0$ or at some other finite time instant. That is, the transient response decays toward zero as $n \to \infty$, leaving only the steady-state response that we determined in the preceding section. In many practical applications, the transient response of the system is unimportant, and therefore it is usually ignored in dealing with the response of the system to sinusoidal inputs.

### 5.1.3 Steady-State Response to Periodic Input Signals

Suppose that the input to a stable linear time-invariant system is a periodic signal $x(n)$ with fundamental period $N$. Since such a signal exists from $-\infty < n < \infty$, the total response of the system at any time instant $n$ is simply equal to the steady-state response.

To determine the response $y(n)$ of the system, we make use of the Fourier series representation of the periodic signal, which is

$$x(n) = \sum_{k=0}^{N-1} c_k e^{j2\pi kn/N}, \qquad k = 0, 1, \ldots, N-1 \tag{5.1.26}$$

where the $\{c_k\}$ are the Fourier series coefficients. Now the response of the system to the complex exponential signal

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

is

$$y_k(n) = c_k H\left(\frac{2\pi}{N}k\right) e^{j2\pi kn/N}, \qquad k = 0, 1, \ldots, N-1 \tag{5.1.27}$$

where

$$H\left(\frac{2\pi k}{N}\right) = H(\omega)|_{\omega=2\pi k/N}, \qquad k = 0, 1, \ldots, N-1$$

By using the superposition principle for linear systems, we obtain the response of the system to the periodic signal $x(n)$ in (5.1.26) as

$$y(n) = \sum_{k=0}^{N-1} c_k H\left(\frac{2\pi k}{N}\right) e^{j2\pi kn/N}, \qquad -\infty < n < \infty \tag{5.1.28}$$

This result implies that the response of the system to the periodic input signal $x(n)$ is also periodic with the same period $N$. The Fourier series coefficients for $y(n)$ are

$$d_k \equiv c_k H\left(\frac{2\pi k}{N}\right), \qquad k = 0, 1, \ldots, N-1 \tag{5.1.29}$$

Hence, the linear system can change the shape of the periodic input signal by scaling the amplitude and shifting the phase of the Fourier series components, but it does not affect the period of the periodic input signal.

### 5.1.4 Response to Aperiodic Input Signals

The convolution theorem, given in (4.4.49), provides the desired frequency-domain relationship for determining the output of an LTI system to an aperiodic finite-energy signal. If $\{x(n)\}$ denotes the input sequence, $\{y(n)\}$ denotes the output sequence, and $\{h(n)\}$ denotes the unit sample response of the system, then from the convolution theorem, we have

$$Y(\omega) = H(\omega)X(\omega) \tag{5.1.30}$$

where $Y(\omega)$, $X(\omega)$, and $H(\omega)$ are the corresponding Fourier transforms of $\{y(n)\}$, $\{x(n)\}$, and $\{h(n)\}$, respectively. From this relationship we observe that the spectrum of the output signal is equal to the spectrum of the input signal multiplied by the frequency response of the system.

If we express $Y(\omega)$, $H(\omega)$, and $X(\omega)$ in polar form, the magnitude and phase of the output signal can be expressed as

$$|Y(\omega)| = |H(\omega)||X(\omega)| \tag{5.1.31}$$

$$\measuredangle Y(\omega) = \measuredangle X(\omega) + \measuredangle H(\omega) \tag{5.1.32}$$

where $|H(\omega)|$ and $\measuredangle H(\omega)$ are the magnitude and phase responses of the system.

By its very nature, a finite-energy aperiodic signal contains a continuum of frequency components. The linear time-invariant system, through its frequency response function, attenuates some frequency components of the input signal and amplifies other frequency components. Thus the system acts as a *filter* to the input signal. Observation of the graph of $|H(\omega)|$ shows which frequency components are amplified and which are attenuated. On the other hand, the angle of $H(\omega)$ determines the phase shift imparted in the continuum of frequency components of the input signal as a function of frequency. If the input signal spectrum is changed by the system in an undesirable way, we say that the system has caused *magnitude and phase distortion.*

We also observe that *the output of a linear time-invariant system cannot contain frequency components that are not contained in the input signal.* It takes either a linear time-variant system or a nonlinear system to create frequency components that are not necessarily contained in the input signal.

Figure 5.1.3 illustrates the time-domain and frequency-domain relationships that can be used in the analysis of BIBO-stable LTI systems. We observe that in time-domain analysis, we deal with the convolution of the input signal with the impulse

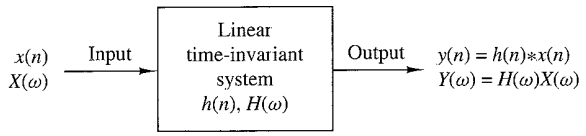



**Figure 5.1.3**
Time- and frequency-domain input–output relationships in LTI systems.

response of the system to obtain the output sequence of the system. On the other hand, in frequency-domain analysis, we deal with the input signal spectrum $X(\omega)$ and the frequency response $H(\omega)$ of the system, which are related through multiplication, to yield the spectrum of the signal at the output of the system.

We can use the relation in (5.1.30) to determine the spectrum $Y(\omega)$ of the output signal. Then the output sequence $\{y(n)\}$ can be determined from the inverse Fourier transform

$$y(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} Y(\omega)e^{j\omega n}d\omega \tag{5.1.33}$$

However, this method is seldom used. Instead, the $z$-transform introduced in Chapter 3 is a simpler method for solving the problem of determining the output sequence $\{y(n)\}$.

Let us return to the basic input–output relation in (5.1.30) and compute the squared magnitude of both sides. Thus we obtain

$$\begin{aligned} |Y(\omega)|^2 &= |H(\omega)|^2|X(\omega)|^2 \\ S_{yy}(\omega) &= |H(\omega)|^2 S_{xx}(\omega) \end{aligned} \tag{5.1.34}$$

where $S_{xx}(\omega)$ and $S_{yy}(\omega)$ are the energy density spectra of the input and output signals, respectively. By integrating (5.1.34) over the frequency range $(-\pi, \pi)$, we obtain the energy of the output signal as

$$\begin{aligned} E_y &= \frac{1}{2\pi}\int_{-\pi}^{\pi} S_{yy}(\omega)d\omega \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} |H(\omega)|^2 S_{xx}(\omega)d\omega \end{aligned} \tag{5.1.35}$$

**EXAMPLE 5.1.5**

A linear time-invariant system is characterized by its impulse response

$$h(n) = (\frac{1}{2})^n u(n)$$

Determine the spectrum and the energy density spectrum of the output signal when the system is excited by the signal

$$x(n) = (\frac{1}{4})^n u(n)$$

**Solution.** The frequency response function of the system

$$H(\omega) = \sum_{n=0}^{\infty} (\frac{1}{2})^n e^{-j\omega n}$$

$$= \frac{1}{1 - \frac{1}{2}e^{-j\omega}}$$

Similarly, the input sequence $\{x(n)\}$ has a Fourier transform

$$X(\omega) = \frac{1}{1 - \frac{1}{4}e^{-j\omega}}$$

Hence the spectrum of the signal at the output of the system is

$$Y(\omega) = H(\omega)X(\omega)$$

$$= \frac{1}{(1 - \frac{1}{2}e^{-j\omega})(1 - \frac{1}{4}e^{-j\omega})}$$

The corresponding energy density spectrum is

$$S_{yy}(\omega) = |Y(\omega)|^2 = |H(\omega)|^2 |X(\omega)|^2$$

$$= \frac{1}{(\frac{5}{4} - \cos\omega)(\frac{17}{16} - \frac{1}{2}\cos\omega)}$$

---

## 5.2 Frequency Response of LTI Systems

In this section we focus on determining the frequency response of LTI systems that have rational system functions. Recall that this class of LTI systems is described in the time domain by constant-coefficient difference equations.

### 5.2.1 Frequency Response of a System with a Rational System Function

From the discussion in Section 4.2.6 we know that if the system function $H(z)$ converges on the unit circle, we can obtain the frequency response of the system by evaluating $H(z)$ on the unit circle. Thus

$$H(\omega) = H(z)|_{z=e^{j\omega}} = \sum_{n=-\infty}^{\infty} h(n)e^{-j\omega n} \tag{5.2.1}$$

In the case where $H(z)$ is a rational function of the form $H(z) = B(z)/A(z)$, we have

$$H(\omega) = \frac{B(\omega)}{A(\omega)} = \frac{\sum_{k=0}^{M} b_k e^{-j\omega k}}{1 + \sum_{k=1}^{N} a_k e^{-j\omega k}} \tag{5.2.2}$$

$$= b_0 \frac{\prod_{k=1}^{M}(1 - z_k e^{-j\omega})}{\prod_{k=1}^{N}(1 - p_k e^{-j\omega})} \tag{5.2.3}$$

where the $\{a_k\}$ and $\{b_k\}$ are real, but $\{z_k\}$ and $\{p_k\}$ may be complex valued.

It is sometimes desirable to express the magnitude squared of $H(\omega)$ in terms of $H(z)$. First, we note that

$$|H(\omega)|^2 = H(\omega)H^*(\omega)$$

For the rational system function given by (5.2.3), we have

$$H^*(\omega) = b_0 \frac{\prod_{k=1}^{M}(1 - z_k^* e^{j\omega})}{\prod_{k=1}^{N}(1 - p_k^* e^{j\omega})} \tag{5.2.4}$$

It follows that $H^*(\omega)$ is obtained by evaluating $H^*(1/z^*)$ on the unit circle, where for a rational system function,

$$H^*(1/z^*) = b_0 \frac{\prod_{k=1}^{M}(1 - z_k^* z)}{\prod_{k=1}^{N}(1 - p_k^* z)} \tag{5.2.5}$$

However, when $\{h(n)\}$ is real or, equivalently, the coefficients $\{a_k\}$ and $\{b_k\}$ are real, complex-valued poles and zeros occur in complex-conjugate pairs. In this case, $H^*(1/z^*) = H(z^{-1})$. Consequently, $H^*(\omega) = H(-\omega)$, and

$$|H(\omega)|^2 = H(\omega)H^*(\omega) = H(\omega)H(-\omega) = H(z)H(z^{-1})|_{z=e^{j\omega}} \tag{5.2.6}$$

According to the correlation theorem for the $z$-transform (see Table 3.2), the function $H(z)H(z^{-1})$ is the $z$-transform of the autocorrelation sequence $\{r_{hh}(m)\}$

of the unit sample response $\{h(n)\}$. Then it follows from the Wiener–Khintchine theorem that $|H(\omega)|^2$ is the Fourier transform of $\{r_{hh}(m)\}$.

Similarly, if $H(z) = B(z)/A(z)$, the transforms $D(z) = B(z)B(z^{-1})$ and $C(z) = A(z)A(z^{-1})$ are the $z$-transforms of the autocorrelation sequences $\{c_l\}$ and $\{d_l\}$, where

$$c_l = \sum_{k=0}^{N-|l|} a_k a_{k+l}, \qquad -N \le l \le N \tag{5.2.7}$$

$$d_l = \sum_{k=0}^{M-|l|} b_k b_{k+l}, \qquad -M \le l \le M \tag{5.2.8}$$

Since the system parameters $\{a_k\}$ and $\{b_k\}$ are real valued, it follows that $c_l = c_{-l}$ and $d_l = d_{-l}$. By using this symmetry property, $|H(\omega)|^2$ may be expressed as

$$|H(\omega)|^2 = \frac{d_0 + 2\sum_{k=1}^{M} d_k \cos k\omega}{c_0 + 2\sum_{k=1}^{N} c_k \cos k\omega} \tag{5.2.9}$$

Finally, we note that $\cos k\omega$ can be expressed as a polynomial function of $\cos\omega$. That is,

$$\cos k\omega = \sum_{m=0}^{k} \beta_m (\cos\omega)^m \tag{5.2.10}$$

where $\{\beta_m\}$ are the coefficients in the expansion. Consequently, the numerator and denominator of $|H(\omega)|^2$ can be viewed as polynomial functions of $\cos\omega$. The following example illustrates the foregoing relationships.

### EXAMPLE 5.2.1

Determine $|H(\omega)|^2$ for the system

$$y(n) = -0.1y(n-1) + 0.2y(n-2) + x(n) + x(n-1)$$

**Solution.** The system function is

$$H(z) = \frac{1 + z^{-1}}{1 + 0.1z^{-1} - 0.2z^{-2}}$$

and its ROC is $|z| > 0.5$. Hence $H(\omega)$ exists. Now

$$\begin{aligned} H(z)H(z^{-1}) &= \frac{1 + z^{-1}}{1 + 0.1z^{-1} - 0.2z^{-2}} \cdot \frac{1 + z}{1 + 0.1z - 0.2z^2} \\ &= \frac{2 + z + z^{-1}}{1.05 + 0.08(z + z^{-1}) - 0.2(z^{-2} + z^{-2})} \end{aligned}$$

By evaluating $H(z)H(z^{-1})$ on the unit circle, we obtain

$$|H(\omega)|^2 = \frac{2 + 2\cos\omega}{1.05 + 0.16\cos\omega - 0.4\cos 2\omega}$$

However, $\cos 2\omega = 2\cos^2\omega - 1$. Consequently, $|H(\omega)|^2$ may be expressed as

$$|H(\omega)|^2 = \frac{2(1 + \cos\omega)}{1.45 + 0.16\cos\omega - 0.8\cos^2\omega}$$

---

We note that given $H(z)$, it is straightforward to determine $H(z^{-1})$ and then $|H(\omega)|^2$. However, the inverse problem of determining $H(z)$, given $|H(\omega)|^2$ or the corresponding impulse response $\{h(n)\}$, is not straightforward. Since $|H(\omega)|^2$ does not contain the phase information in $H(\omega)$, it is not possible to uniquely determine $H(z)$.

To elaborate on the point, let us assume that the $N$ poles and $M$ zeros of $H(z)$ are $\{p_k\}$ and $\{z_k\}$, respectively. The corresponding poles and zeros of $H(z^{-1})$ are $\{1/p_k\}$ and $\{1/z_k\}$, respectively. Given $|H(\omega)|^2$ or, equivalently, $H(z)H(z^{-1})$, we can determine different system functions $H(z)$ by assigning to $H(z)$, a pole $p_k$ or its reciprocal $1/p_k$, and a zero $z_k$ or its reciprocal $1/z_k$. For example, if $N = 2$ and $M = 1$, the poles and zeros of $H(z)H(z^{-1})$ are $\{p_1, p_2, 1/p_1, 1/p_2\}$ and $\{z_1, 1/z_1\}$. If $p_1$ and $p_2$ are real, the possible poles for $H(z)$ are $\{p_1, p_2\}$, $\{1/p_1, 1/p_2\}$, $\{p_1, 1/p_2\}$, and $\{p_2, 1/p_1\}$ and the possible zeros are $\{z_1\}$ or $\{1/z_1\}$. Therefore, there are eight possible choices of system functions, all of which result in the same $|H(\omega)|^2$. Even if we restrict the poles of $H(z)$ to be inside the unit circle, there are still two different choices for $H(z)$, depending on whether we pick the zero $\{z_1\}$ or $\{1/z_1\}$. Therefore, we cannot determine $H(z)$ uniquely given only the magnitude response $|H(\omega)|$.

### 5.2.2 Computation of the Frequency Response Function

In evaluating the magnitude response and the phase response as functions of frequency, it is convenient to express $H(\omega)$ in terms of its poles and zeros. Hence we write $H(\omega)$ in factored form as

$$H(\omega) = b_0 \frac{\prod_{k=1}^{M}(1 - z_k e^{-j\omega k})}{\prod_{k=1}^{N}(1 - p_k e^{-j\omega k})} \tag{5.2.11}$$

or, equivalently, as

$$H(\omega) = b_0 e^{j\omega(N-M)} \frac{\prod_{k=1}^{M}(e^{j\omega} - z_k)}{\prod_{k=1}^{N}(e^{j\omega} - p_k)} \tag{5.2.12}$$

Let us express the complex-valued factors in (5.2.12) in polar form as

$$e^{j\omega} - z_k = V_k(\omega)e^{j\Theta_k(\omega)} \tag{5.2.13}$$

and

$$e^{j\omega} - p_k = U_k(\omega)e^{j\Phi_k(\omega)} \tag{5.2.14}$$

where

$$V_k(\omega) \equiv |e^{j\omega} - z_k|, \quad \Theta_k(\omega) \equiv \measuredangle(e^{j\omega} - z_k) \tag{5.2.15}$$

and

$$U_k(\omega) \equiv |e^{j\omega} - p_k|, \quad \Phi_k(\omega) = \measuredangle(e^{j\omega} - p_k) \tag{5.2.16}$$

The magnitude of $H(\omega)$ is equal to the product of magnitudes of all terms in (5.2.12). Thus, using (5.2.13) through (5.2.16), we obtain

$$|H(\omega)| = |b_0|\frac{V_1(\omega)\cdots V_M(\omega)}{U_1(\omega)U_2(\omega)\cdots U_N(\omega)} \tag{5.2.17}$$

since the magnitude of $e^{j\omega(N-M)}$ is 1.

The phase of $H(\omega)$ is the sum of the phases of the numerator factors, minus the phases of the denominator factors. Thus, by combining (5.2.13) through (5.2.16), we have

$$\begin{aligned}\measuredangle H(\omega) = \measuredangle b_0 + \omega(N-M) + \Theta_1(\omega) + \Theta_2(\omega) + \cdots + \Theta_M(\omega)\\ - [\Phi_1(\omega) + \Phi_2(\omega) + \cdots + \Phi_N(\omega)]\end{aligned} \tag{5.2.18}$$

The phase of the gain term $b_0$ is zero or $\pi$, depending on whether $b_0$ is positive or negative. Clearly, if we know the zeros and the poles of the system function $H(z)$, we can evaluate the frequency response from (5.2.17) and (5.2.18).

There is a geometric interpretation of the quantities appearing in (5.2.17) and (5.2.18). Let us consider a pole $p_k$ and a zero $z_k$ located at points $A$ and $B$ of the $z$-plane, as shown in Fig. 5.2.1(a). Assume that we wish to compute $H(\omega)$ at a specific value of frequency $\omega$. The given value of $\omega$ determines the angle of $e^{j\omega}$ with the positive real axis. The tip of the vector $e^{j\omega}$ specifies a point $L$ on the unit circle. The evaluation of the Fourier transform for the given value of $\omega$ is equivalent to evaluating the $z$-transform at the point $L$ of the complex plane. Let us draw the vectors **AL** and **BL** from the pole and zero locations to the point L, at which we wish to compute the Fourier transform. From Fig. 5.2.1(a) it follows that

$$\mathbf{CL} = \mathbf{CA} + \mathbf{AL}$$

and

$$\mathbf{CL} = \mathbf{CB} + \mathbf{BL}$$

However, $\mathbf{CL} = e^{j\omega}$, $\mathbf{CA} = p_k$ and $\mathbf{CB} = z_k$. Thus

$$\mathbf{AL} = e^{j\omega} - p_k \tag{5.2.19}$$

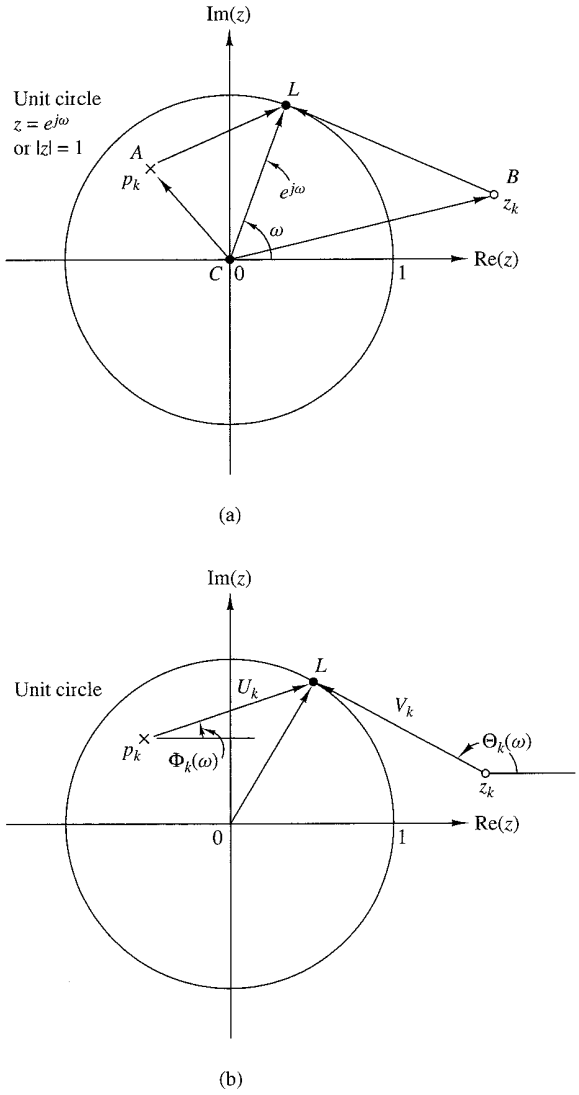



**Figure 5.2.1**
Geometric interpretation of the contribution of a pole and a zero to the Fourier transform (1) magnitude: the factor $V_k/U_k$, (2) phase: the factor $\Theta_k - \Phi_k$.

and

$$\mathbf{BL} = e^{j\omega} - z_k \tag{5.2.20}$$

By combining these relations with (5.2.13) and (5.2.14), we obtain

$$\mathbf{AL} = e^{j\omega} - p_k = U_k(\omega)e^{j\Phi_k(\omega)} \tag{5.2.21}$$

$$\mathbf{BL} = e^{j\omega} - z_k = V_k(\omega)e^{j\Theta_k(\omega)} \tag{5.2.22}$$

Thus $U_k(\omega)$ is the length of **AL**, that is, the distance of the pole $p_k$ from the point $L$ corresponding to $e^{j\omega}$, whereas $V_k(\omega)$ is the distance of the zero $z_k$ from the same

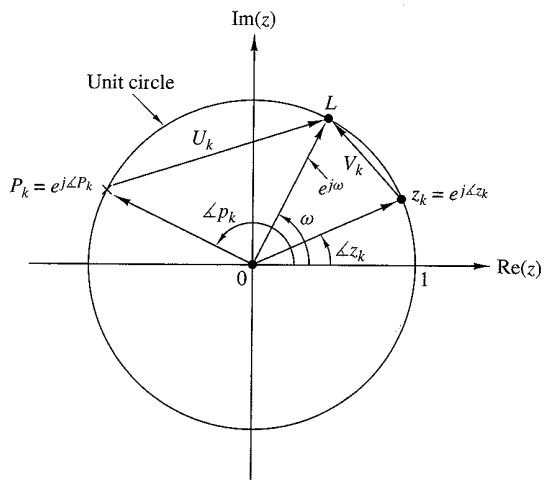



**Figure 5.2.2**
A zero on the unit circle causes $|H(\omega)| = 0$ and $\omega = \measuredangle z_k$. In contrast, a pole on the unit circle results in $|H(\omega)| = \infty$ at $\omega = \measuredangle p_k$.

point $L$. The phases $\Phi_k(\omega)$ and $\Theta_k(\omega)$ are the angles of the vectors **AL** and **BL** with the positive real axis, respectively. These geometric interpretations are shown in Fig. 5.2.1(b).

Geometric interpretations are very useful in understanding how the location of poles and zeros affects the magnitude and phase of the Fourier transform. Suppose that a zero, say $z_k$, and a pole, say $p_k$, are on the unit circle as shown in Fig. 5.2.2. We note that at $\omega = \measuredangle z_k$, $V_k(\omega)$ and consequently $|H(\omega)|$ become zero. Similarly, at $\omega = \measuredangle p_k$ the length $U_k(\omega)$ becomes zero and hence $|H(\omega)|$ becomes infinite. Clearly, the evaluation of phase in these cases has no meaning.

From this discussion we can easily see that the presence of a zero close to the unit circle causes the magnitude of the frequency response, at frequencies that correspond to points of the unit circle close to that point, to be small. In contrast, the presence of a pole close to the unit circle causes the magnitude of the frequency response to be large at frequencies close to that point. Thus poles have the opposite effect of zeros. Also, placing a zero close to a pole cancels the effect of the pole, and vice versa. This can be also seen from (5.2.12), since if $z_k = p_k$, the terms $e^{j\omega} - z_k$ and $e^{j\omega} - p_k$ cancel. Obviously, the presence of both poles and zeros in a transform results in a greater variety of shapes for $|H(\omega)|$ and $\measuredangle H(\omega)$. This observation is very important in the design of digital filters. We conclude our discussion with the following example illustrating these concepts.

### EXAMPLE 5.2.2

Evaluate the frequency response of the system described by the system function

$$H(z) = \frac{1}{1 - 0.8z^{-1}} = \frac{z}{z - 0.8}$$

**Solution.** Clearly, $H(z)$ has a zero at $z = 0$ and a pole at $p = 0.8$. Hence the frequency response of the system is

$$H(\omega) = \frac{e^{j\omega}}{e^{j\omega} - 0.8}$$

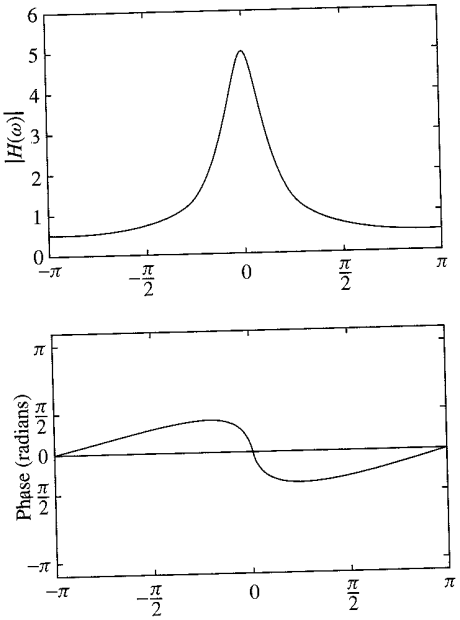



**Figure 5.2.3**
Magnitude and phase of system with $H(z) = 1/(1 - 0.8z^{-1})$.

The magnitude response is

$$|H(\omega)| = \frac{|e^{j\omega}|}{|e^{j\omega} - 0.8|} = \frac{1}{\sqrt{1.64 - 1.6\cos\omega}}$$

and the phase response is

$$\theta(\omega) = \omega - \tan^{-1}\frac{\sin\omega}{\cos\omega - 0.8}$$

The magnitude and phase responses are illustrated in Fig. 5.2.3. Note that the peak of the magnitude response occurs at $\omega = 0$, the point on the unit circle closest to the pole located at 0.8.

If the magnitude response in (5.2.17) is expressed in decibels,

$$|H(\omega)|_{dB} = 20\log_{10}|b_0| + 20\sum_{k=1}^{M}\log_{10}V_k(\omega) - 20\sum_{k=1}^{N}\log_{10}U_k(\omega) \qquad (5.2.23)$$

Thus the magnitude response is expressed as a sum of the magnitude factors in $|H(\omega)|$.

## 5.3 Correlation Functions and Spectra at the Output of LTI Systems

In this section, we derive the spectral relationships between the input and output signals of LTI systems. Section 5.3.1 describes the relationships for the energy density spectra of deterministic input and output signals. Section 5.3.2 is focused on the relationships for the power density spectra of random input and output signals.

### 5.3.1 Input–Output Correlation Functions and Spectra

In Section 2.6.4 we developed several correlation relationships between the input and the output sequences of an LTI system. Specifically, we derived the following equations:

$$r_{yy}(m) = r_{hh}(m) * r_{xx}(m) \tag{5.3.1}$$

$$r_{yx}(m) = h(m) * r_{xx}(m) \tag{5.3.2}$$

where $r_{xx}(m)$ is the autocorrelation sequence of the input signal $\{x(n)\}$, $r_{yy}(m)$ is the autocorrelation sequence of the output $\{y(n)\}$, $r_{hh}(m)$ is the autocorrelation sequence of the impulse response $\{h(n)\}$, and $r_{yx}(m)$ is the crosscorrelation sequence between the output and the input signals. Since (5.3.1) and (5.3.2) involve the convolution operation, the $z$-transform of these equations yields

$$\begin{aligned} S_{yy}(z) &= S_{hh}(z)S_{xx}(z) \\ &= H(z)H(z^{-1})S_{xx}(z) \end{aligned} \tag{5.3.3}$$

$$S_{yx}(z) = H(z)S_{xx}(z) \tag{5.3.4}$$

If we substitute $z = e^{j\omega}$ in (5.3.4), we obtain

$$\begin{aligned} S_{yx}(\omega) &= H(\omega)S_{xx}(\omega) \\ &= H(\omega)|X(\omega)|^2 \end{aligned} \tag{5.3.5}$$

where $S_{yx}(\omega)$ is the cross-energy density spectrum of $\{y(n)\}$ and $\{x(n)\}$. Similarly, evaluating $S_{yy}(z)$ on the unit circle yields the energy density spectrum of the output signal as

$$S_{yy}(\omega) = |H(\omega)|^2 S_{xx}(\omega) \tag{5.3.6}$$

where $S_{xx}(\omega)$ is the energy density spectrum of the input signal.

Since $r_{yy}(m)$ and $S_{yy}(\omega)$ are a Fourier transform pair, it follows that

$$r_{yy}(m) = \frac{1}{2\pi}\int_{-\pi}^{\pi} S_{yy}(\omega)e^{j\omega m}d\omega \tag{5.3.7}$$

The total energy in the output signal is simply

$$\begin{aligned} E_y &= \frac{1}{2\pi}\int_{-\pi}^{\pi} S_{yy}(\omega)d\omega = r_{yy}(0) \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} |H(\omega)|^2\, S_{xx}(\omega)d\omega \end{aligned} \tag{5.3.8}$$

The result in (5.3.8) may be used to easily prove that $E_y \geq 0$.

Finally, we note that if the input signal has a flat spectrum [i.e., $S_{xx}(\omega) = S_x =$ constant for $\pi \leq \omega \leq -\pi$], (5.3.5) reduces to

$$S_{yx}(\omega) = H(\omega)S_x \tag{5.3.9}$$

where $S_x$ is the constant value of the spectrum. Hence

$$H(\omega) = \frac{1}{S_x} S_{yx}(\omega) \tag{5.3.10}$$

or, equivalently,

$$h(n) = \frac{1}{S_x} r_{yx}(m) \tag{5.3.11}$$

The relation in (5.3.11) implies that $h(n)$ can be determined by exciting the input to the system by a spectrally flat signal $\{x(n)\}$, and crosscorrelating the input with the output of the system. This method is useful in measuring the impulse response of an unknown system.

### 5.3.2 Correlation Functions and Power Spectra for Random Input Signals

This development parallels the derivations in Section 5.3.1, with the exception that we now deal with the statistical mean and autocorrelation of the input and output signals of an LTI system.

Let us consider a discrete-time linear time-invariant system with unit sample response $\{h(n)\}$ and frequency response $H(f)$. For this development we assume that $\{h(n)\}$ is real. Let $x(n)$ be a sample function of a stationary random process $X(n)$ that excites the system and let $y(n)$ denote the response of the system to $x(n)$.

From the convolution summation that relates the output to the input we have

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)x(n-k) \tag{5.3.12}$$

Since $x(n)$ is a random input signal, the output is also a random sequence. In other words, for each sample sequence $x(n)$ of the process $X(n)$, there is a corresponding sample sequence $y(n)$ of the output random process $Y(n)$. We wish to relate the statistical characteristics of the output random process $Y(n)$ to the statistical characterization of the input process and the characteristics of the system.

The expected value of the output $y(n)$ is

$$\begin{aligned} m_y \equiv E[y(n)] &= E[\sum_{k=-\infty}^{\infty} h(k)x(n-k)] \\ &= \sum_{k=-\infty}^{\infty} h(k)E[x(n-k)] \end{aligned} \tag{5.3.13}$$

$$m_y = m_x \sum_{k=-\infty}^{\infty} h(k)$$

From the Fourier transform relationship

$$H(\omega) = \sum_{k=-\infty}^{\infty} h(k)e^{-j\omega k} \tag{5.3.14}$$

we have

$$H(0) = \sum_{k=-\infty}^{\infty} h(k) \tag{5.3.15}$$

which is the dc gain of the system. The relationship in (5.3.15) allows us to express the mean value in (5.3.13) as

$$m_y = m_x H(0) \tag{5.3.16}$$

The autocorrelation sequence for the output random process is defined as

$$\begin{aligned}
\gamma_{yy}(m) &= E[y^*(n)y(n+m)] \\
&= E\left[\sum_{k=-\infty}^{\infty} h(k)x^*(n-k) \sum_{j=-\infty}^{\infty} h(j)x(n+m-j)\right] \\
&= \sum_{k=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} h(k)h(j)E[x^*(n-k)x(n+m-j)] \\
&= \sum_{k=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} h(k)h(j)\gamma_{xx}(k-j+m)
\end{aligned} \tag{5.3.17}$$

This is the general form for the autocorrelation of the output in terms of the autocorrelation of the input and the impulse response of the system.

A special form of (5.3.17) is obtained when the input random process is white, that is, when $m_x = 0$ and

$$\gamma_{xx}(m) = \sigma_x^2 \delta(m) \tag{5.3.18}$$

where $\sigma_x^2 \equiv \gamma_{xx}(0)$ is the input signal power. Then (5.3.17) reduces to

$$\gamma_{yy}(m) = \sigma_x^2 \sum_{k=-\infty}^{\infty} h(k)h(k+m) \tag{5.3.19}$$

Under this condition the output process has the average power

$$\gamma_{yy}(0) = \sigma_x^2 \sum_{n=-\infty}^{\infty} h^2(n) = \sigma_x^2 \int_{-1/2}^{1/2} |H(f)|^2 df \tag{5.3.20}$$

where we have applied Parseval's theorem.

The relationship in (5.3.17) can be transformed into the frequency domain by determining the power density spectrum of $\gamma_{yy}(m)$. We have

$$\begin{aligned}
\Gamma_{yy}(\omega) &= \sum_{m=-\infty}^{\infty} \gamma_{yy}(m)e^{-j\omega m} \\
&= \sum_{m=-\infty}^{\infty} \left[ \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} h(k)h(l)\gamma_{xx}(k-l+m) \right] e^{-j\omega m} \\
&= \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} h(k)h(l) \left[ \sum_{m=-\infty}^{\infty} \gamma_{xx}(k-l+m)e^{-j\omega m} \right] \qquad (5.3.21) \\
&= \Gamma_{xx}(f) \left[ \sum_{k=-\infty}^{\infty} h(k)e^{j\omega k} \right] \left[ \sum_{l=-\infty}^{\infty} h(l)e^{-j\omega l} \right] \\
&= |H(\omega)|^2 \Gamma_{xx}(\omega)
\end{aligned}$$

This is the desired relationship for the power density spectrum of the output process, in terms of the power density spectrum of the input process and the frequency response of the system.

The equivalent expression for continuous-time systems with random inputs is

$$\Gamma_{yy}(F) = |H(F)|^2 \Gamma_{xx}(F) \qquad (5.3.22)$$

where the power density spectra $\Gamma_{yy}(F)$ and $\Gamma_{xx}(F)$ are the Fourier transforms of the autocorrelation functions $\gamma_{yy}(\tau)$ and $\gamma_{xx}(\tau)$, respectively, and where $H(F)$ is the frequency response of the system, which is related to the impulse response by the Fourier transform, that is,

$$H(F) = \int_{-\infty}^{\infty} h(t)e^{-j2\pi Ft}\, dt \qquad (5.3.23)$$

As a final exercise, we determine the crosscorrelation of the output $y(n)$ with the input signal $x(n)$. If we multiply both sides of (5.3.12) by $x^*(n-m)$ and take the expected value, we obtain

$$\begin{aligned}
E[y(n)x^*(n-m)] &= E\left[ \sum_{k=-\infty}^{\infty} h(k)x^*(n-m)x(n-k) \right] \\
\gamma_{yx}(m) &= \sum_{k=-\infty}^{\infty} h(k)E[x^*(n-m)x(n-k)] \qquad (5.3.24) \\
&= \sum_{k=-\infty}^{\infty} h(k)\gamma_{xx}(m-k)
\end{aligned}$$

Since (5.3.24) has the form of a convolution, the frequency-domain equivalent expression is

$$\Gamma_{yx}(\omega) = H(\omega)\Gamma_{xx}(\omega) \tag{5.3.25}$$

In the special case where $x(n)$ is white noise, (5.3.25) reduces to

$$\Gamma_{yx}(\omega) = \sigma_x^2 H(\omega) \tag{5.3.26}$$

where $\sigma_x^2$ is the input noise power. This result means that an unknown system with frequency response $H(\omega)$ can be identified by exciting the input with white noise, crosscorrelating the input sequence with the output sequence to obtain $\gamma_{yx}(m)$, and finally, computing the Fourier transform of $\gamma_{yx}(m)$. The result of these computations is proportional to $H(\omega)$.

## 5.4 Linear Time-Invariant Systems as Frequency-Selective Filters

The term *filter* is commonly used to describe a device that discriminates, according to some attribute of the objects applied at its input, what passes through it. For example, an air filter allows air to pass through it but prevents dust particles that are present in the air from passing through. An oil filter performs a similar function, with the exception that oil is the substance allowed to pass through the filter, while particles of dirt are collected at the input to the filter and prevented from passing through. In photography, an ultraviolet filter is often used to prevent ultraviolet light, which is present in sunlight and which is not a part of visible light, from passing through and affecting the chemicals on the film.

As we have observed in the preceding section, a linear time-invariant system also performs a type of discrimination or filtering among the various frequency components at its input. The nature of this filtering action is determined by the frequency response characteristics $H(\omega)$, which in turn depends on the choice of the system parameters (e.g., the coefficients $\{a_k\}$ and $\{b_k\}$ in the difference equation characterization of the system). Thus, by proper selection of the coefficients, we can design frequency-selective filters that pass signals with frequency components in some bands while they attenuate signals containing frequency components in other frequency bands.

In general, a linear time-invariant system modifies the input signal spectrum $X(\omega)$ according to its frequency response $H(\omega)$ to yield an output signal with spectrum $Y(\omega) = H(\omega)X(\omega)$. In a sense, $H(\omega)$ acts as a *weighting function* or a *spectral shaping function* to the different frequency components in the input signal. When viewed in this context, any linear time-invariant system can be considered to be a frequency-shaping filter, even though it may not necessarily completely block any or all frequency components. Consequently, the terms "linear time-invariant system" and "filter" are synonymous and are often used interchangeably.

We use the term *filter* to describe a linear time-invariant system used to perform spectral shaping or frequency-selective filtering. Filtering is used in digital signal processing in a variety of ways, such as removal of undesirable noise from desired signals, spectral shaping such as equalization of communication channels, signal detection in radar, sonar, and communications, and for performing spectral analysis of signals, and so on.

### 5.4.1 Ideal Filter Characteristics

Filters are usually classified according to their frequency-domain characteristics as lowpass, highpass, bandpass, and bandstop or band-elimination filters. The ideal magnitude response characteristics of these types of filters are illustrated in Fig. 5.4.1. As shown, these ideal filters have a constant-gain (usually taken as unity-gain) passband characteristic and zero gain in their stopband.

Another characteristic of an ideal filter is a linear phase response. To demonstrate this point, let us assume that a signal sequence $\{x(n)\}$ with frequency components confined to the frequency range $\omega_1 < \omega < \omega_2$ is passed through a filter with

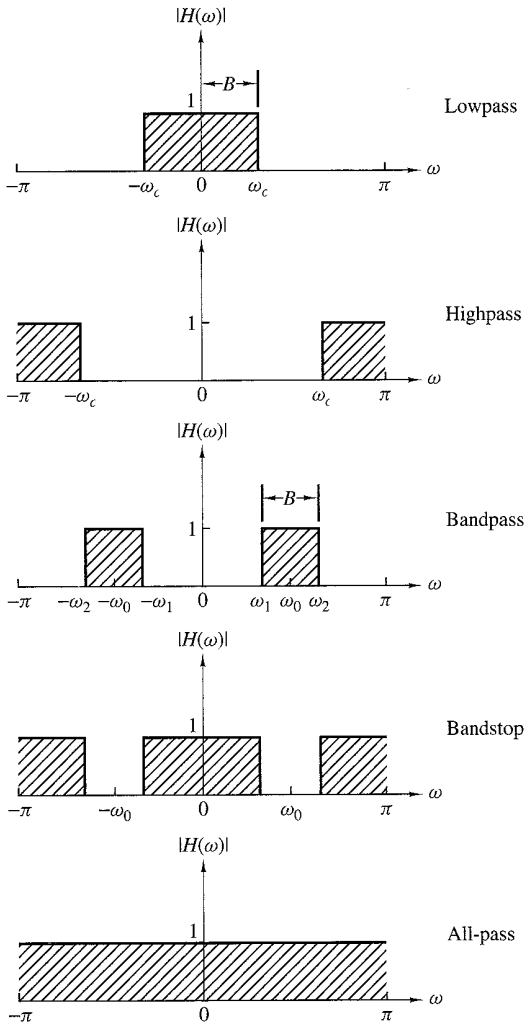



**Figure 5.4.1** Magnitude responses for some ideal frequency-selective discrete-time filters.

frequency response

$$H(\omega) = \begin{cases} Ce^{-j\omega n_0}, & \omega_1 < \omega < \omega_2 \\ 0, & \text{otherwise} \end{cases} \tag{5.4.1}$$

where $C$ and $n_0$ are constants. The signal at the output of the filter has a spectrum

$$\begin{aligned} Y(\omega) &= X(\omega)H(\omega) \\ &= CX(\omega)e^{-j\omega n_0}, \qquad \omega_1 < \omega < \omega_2 \end{aligned} \tag{5.4.2}$$

By applying the scaling and time-shifting properties of the Fourier transform, we obtain the time-domain output

$$y(n) = Cx(n - n_0) \tag{5.4.3}$$

Consequently, the filter output is simply a delayed and amplitude-scaled version of the input signal. A pure delay is usually tolerable and is not considered a distortionof the signal. Neither is amplitude scaling. Therefore, ideal filters have a linear phase characteristic within their passband, that is,

$$\Theta(\omega) = -\omega n_0 \tag{5.4.4}$$

The derivative of the phase with respect to frequency has the units of delay. Hence we can define the signal delay as a function of frequency as

$$\tau_g(\omega) = -\frac{d\Theta(\omega)}{d\omega} \tag{5.4.5}$$

$\tau_g(\omega)$ is usually called the *envelope delay*or the *group delay* of the filter. We interpret $\tau_g(\omega)$ as the time delay that a signal component of frequency $\omega$ undergoes as it passes from the input to the output of the system. Note that when $\Theta(\omega)$ is linear as in (5.4.4), $\tau_g(\omega) = n_0 =$ constant. In this case all frequency components of the input signal undergo the same time delay.

In conclusion, ideal filters have a constant magnitude characteristic and a linear phase characteristic within their passband. In all cases, such filters are not physically realizable but serve as a mathematical idealization of practical filters. For example, the ideal lowpass filter has an impulse response

$$h_{lp}(n) = \frac{\sin \omega_c \pi n}{\pi n}, \qquad -\infty < n < \infty \tag{5.4.6}$$

We note that this filter is not causal and it is not absolutely summable and therefore it is also unstable. Consequently, this ideal filter is physically unrealizable. Nevertheless, its frequency response characteristics can be approximated very closely by practical, physically realizable filters, as will be demonstrated in Chapter 10.

In the following discussion, we treat the design of some simple digital filters by the placement of poles and zeros in the $z$-plane. We have already described how

the location of poles and zeros affects the frequency response characteristics of the system. In particular, in Section 5.2.2 we presented a graphical method for computing the frequency response characteristics from the pole–zero plot. This same approach can be used to design a number of simple but important digital filters with desirable frequency response characteristics.

The basic principle underlying the pole–zero placement method is to locate poles near points of the unit circle corresponding to frequencies to be emphasized, and to place zeros near the frequencies to be deemphasized. Furthermore, the following constraints must be imposed:

1. All poles should be placed inside the unit circle in order for the filter to be stable. However, zeros can be placed anywhere in the $z$-plane.
2. All complex zeros and poles must occur in complex-conjugate pairs in order for the filter coefficients to be real.

From our previous discussion we recall that for a given pole–zero pattern, the system function $H(z)$ can be expressed as

$$H(z) = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}} = b_0 \frac{\prod_{k=1}^{M}(1 - z_k z^{-1})}{\prod_{k=1}^{N}(1 - p_k z^{-1})} \tag{5.4.7}$$

where $b_0$ is a gain constant selected to normalize the frequency response at some specified frequency. That is, $b_0$ is selected such that

$$|H(\omega_0)| = 1 \tag{5.4.8}$$

where $\omega_0$ is a frequency in the passband of the filter. Usually, $N$ is selected to equal or exceed $M$, so that the filter has more nontrivial poles than zeros.

In the next section, we illustrate the method of pole–zero placement in the design of some simple lowpass, highpass, and bandpass filters, digital resonators, and comb filters. The design procedure is facilitated when carried out interactively on a digital computer with a graphics terminal.

### 5.4.2 Lowpass, Highpass, and Bandpass Filters

In the design of lowpass digital filters, the poles should be placed near the unit circle at points corresponding to low frequencies (near $\omega = 0$) and zeros should be placed near or on the unit circle at points corresponding to high frequencies (near $\omega = \pi$). The opposite holds true for highpass filters.

Figure 5.4.2 illustrates the pole–zero placement of three lowpass and three highpass filters. The magnitude and phase responses for the single-pole filter with system function

$$H_1(z) = \frac{1 - a}{1 - az^{-1}} \tag{5.4.9}$$

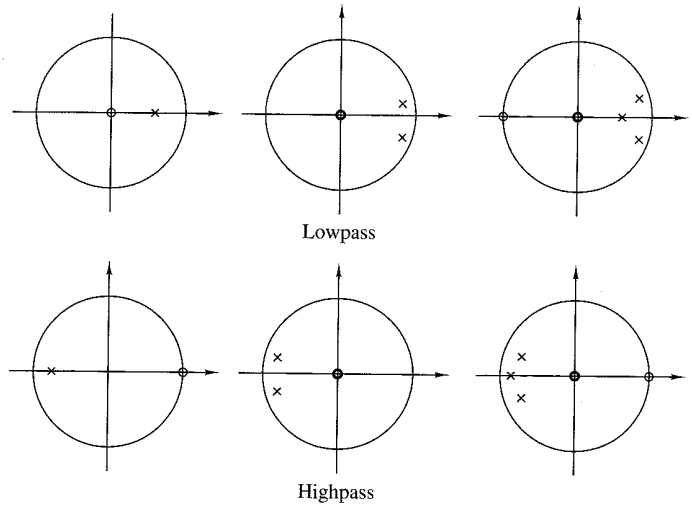



**Figure 5.4.2** Pole–zero patterns for several lowpass and highpass filters.

are illustrated in Fig. 5.4.3 for $a = 0.9$. The gain $G$ was selected as $1 - a$, so that the filter has unity gain at $\omega = 0$. The gain of this filter at high frequencies is relatively small.

The addition of a zero at $z = -1$ further attenuates the response of the filter at high frequencies. This leads to a filter with a system function

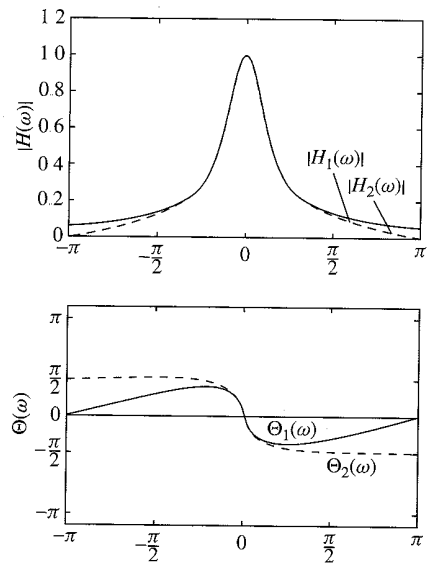



**Figure 5.4.3** Magnitude and phase response of (1) a single-pole filter and (2) a one-pole, one-zero filter; $H_1(z) = (1 - a)/(1 - az^{-1})$, $H_2(z) = [(1 - a)/2][(1 + z^{-1})/(1 - az^{-1})]$ and $a = 0.9$.

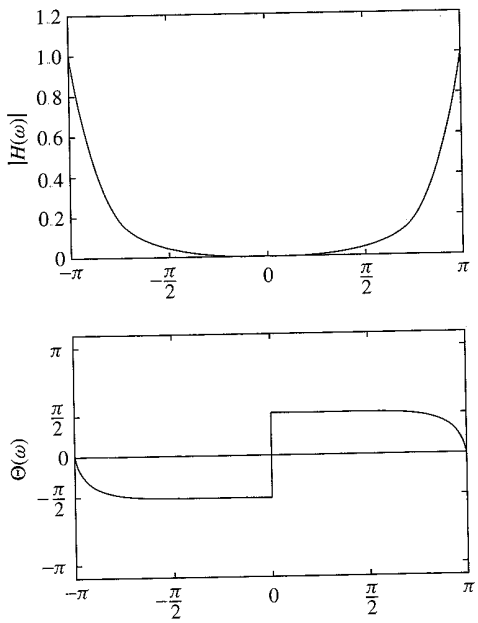



**Figure 5.4.4**
Magnitude and phase response of a simple highpass filter; $H(z) = [(1-a)/2][(1-z^{-1})/(1+az^{-1})]$ with $a = 0.9$.

$$H_2(z) = \frac{1-a}{2}\frac{1+z^{-1}}{1-az^{-1}} \tag{5.4.10}$$

and a frequency response characteristic that is also illustrated in Fig. 5.4.3. In this case the magnitude of $H_2(\omega)$ goes to zero at $\omega = \pi$.

Similarly, we can obtain simple highpass filters by reflecting (folding) the pole–zero locations of the lowpass filters about the imaginary axis in the $z$-plane. Thus we obtain the system function

$$H_3(z) = \frac{1-a}{2}\frac{1-z^{-1}}{1+az^{-1}} \tag{5.4.11}$$

which has the frequency response characteristics illustrated in Fig. 5.4.4 for $a = 0.9$.

### EXAMPLE 5.4.1

A two-pole lowpass filter has the system function

$$H(z) = \frac{b_0}{(1-pz^{-1})^2}$$

Determine the values of $b_0$ and $p$ such that the frequency response $H(\omega)$ satisfies the conditions

$$H(0) = 1$$

and

$$\left|H\left(\frac{\pi}{4}\right)\right|^2 = \frac{1}{2}$$

**Solution.** At $\omega = 0$ we have

$$H(0) = \frac{b_0}{(1-p)^2} = 1$$

Hence

$$b_0 = (1-p)^2$$

At $\omega = \pi/4$,

$$\begin{aligned} H\left(\frac{\pi}{4}\right) &= \frac{(1-p)^2}{(1-pe^{-j\pi/4})^2} \\ &= \frac{(1-p)^2}{(1-p\cos(\pi/4)+jp\sin(\pi/4))^2} \\ &= \frac{(1-p)^2}{(1-p/\sqrt{2}+jp/\sqrt{2})^2} \end{aligned}$$

Hence

$$\frac{(1-p)^4}{[(1-p/\sqrt{2})^2+p^2/2]^2} = \frac{1}{2}$$

or, equivalently,

$$\sqrt{2}(1-p)^2 = 1 + p^2 - \sqrt{2}p$$

The value of $p = 0.32$ satisfies this equation. Consequently, the system function for the desired filter is

$$H(z) = \frac{0.46}{(1-0.32z^{-1})^2}$$

---

The same principles can be applied for the design of bandpass filters. Basically, the bandpass filter should contain one or more pairs of complex-conjugate poles near the unit circle, in the vicinity of the frequency band that constitutes the passband of the filter. The following example serves to illustrate the basic ideas.

### EXAMPLE 5.4.2

Design a two-pole bandpass filter that has the center of its passband at $\omega = \pi/2$, zero in its frequency response characteristic at $\omega = 0$ and $\omega = \pi$, and a magnitude response of $1/\sqrt{2}$ at $\omega = 4\pi/9$.

**Solution.** Clearly, the filter must have poles at

$$p_{1,2} = re^{\pm j\pi/2}$$

and zeros at $z = 1$ and $z = -1$. Consequently, the system function is

$$\begin{aligned} H(z) &= G\frac{(z-1)(z+1)}{(z-jr)(z+jr)} \\ &= G\frac{z^2-1}{z^2+r^2} \end{aligned}$$

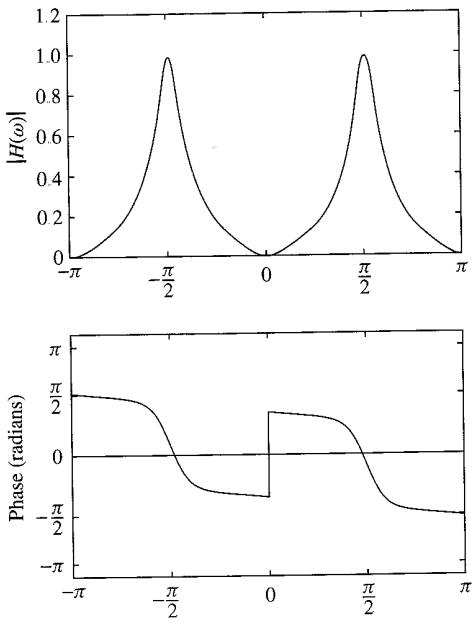



**Figure 5.4.5**
Magnitude and phase response of a simple bandpass filter in Example 5.4.2; $H(z) = 0.15[(1 - z^{-2})/(1 + 0.7z^{-2})]$.

The gain factor is determined by evaluating the frequency response $H(\omega)$ of the filter at $\omega = \pi/2$. Thus we have

$$H\left(\frac{\pi}{2}\right) = G\frac{2}{1 - r^2} = 1$$

$$G = \frac{1 - r^2}{2}$$

The value of $r$ is determined by evaluating $H(\omega)$ at $\omega = 4\pi/9$. Thus we have

$$\left|H\left(\frac{4\pi}{9}\right)\right|^2 = \frac{(1 - r^2)^2}{4}\frac{2 - 2\cos(8\pi/9)}{1 + r^4 + 2r^2\cos(8\pi/9)} = \frac{1}{2}$$

or, equivalently,

$$1.94(1 - r^2)^2 = 1 - 1.88r^2 + r^4$$

The value of $r^2 = 0.7$ satisfies this equation. Therefore, the system function for the desired filter is

$$H(z) = 0.15\frac{1 - z^{-2}}{1 + 0.7z^{-2}}$$

Its frequency response is illustrated in Fig. 5.4.5.

---

It should be emphasized that the main purpose of the foregoing methodology for designing simple digital filters by pole–zero placement is to provide insight into the effect that poles and zeros have on the frequency response characteristic of

systems. The methodology is not intended as a good method for designing digital filters with well-specified passband and stopband characteristics. Systematic methods for the design of sophisticated digital filters for practical applications are discussed in Chapter 10.

**A simple lowpass-to-highpass filter transformation.** Suppose that we have designed a prototype lowpass filter with impulse response $h_{\text{lp}}(n)$. By using the frequency translation property of the Fourier transform, it is possible to convert the prototype filter to either a bandpass or a highpass filter. Frequency transformations for converting a prototype lowpass filter into a filter of another type are described in detail in Section 10.3. In this section we present a simple frequency transformation for converting a lowpass filter into a highpass filter, and vice versa.

If $h_{\text{lp}}(n)$ denotes the impulse response of a lowpass filter with frequency response $H_{\text{lp}}(\omega)$, a highpass filter can be obtained by translating $H_{\text{lp}}(\omega)$ by $\pi$ radians (i.e., replacing $\omega$ by $\omega - \pi$). Thus

$$H_{\text{hp}}(\omega) = H_{\text{lp}}(\omega - \pi) \tag{5.4.12}$$

where $H_{\text{hp}}(\omega)$ is the frequency response of the highpass filter. Since a frequency translation of $\pi$ radians is equivalent to multiplication of the impulse response $h_{\text{lp}}(n)$ by $e^{j\pi n}$, the impulse response of the highpass filter is

$$h_{\text{hp}}(n) = (e^{j\pi})^n h_{\text{lp}}(n) = (-1)^n h_{\text{lp}}(n) \tag{5.4.13}$$

Therefore, the impulse response of the highpass filter is simply obtained from the impulse response of the lowpass filter by changing the signs of the odd-numbered samples in $h_{\text{lp}}(n)$. Conversely,

$$h_{\text{lp}}(n) = (-1)^n h_{\text{hp}}(n) \tag{5.4.14}$$

If the lowpass filter is described by the difference equation

$$y(n) = -\sum_{k=1}^{N} a_k y(n-k) + \sum_{k=0}^{M} b_k x(n-k) \tag{5.4.15}$$

its frequency response is

$$H_{\text{lp}}(\omega) = \frac{\sum_{k=0}^{M} b_k e^{-j\omega k}}{1 + \sum_{k=1}^{N} a_k e^{-j\omega k}} \tag{5.4.16}$$

Now, if we replace $\omega$ by $\omega - \pi$, in (5.4.16), then

$$H_{\text{hp}}(\omega) = \frac{\sum_{k=0}^{M} (-1)^k b_k e^{-j\omega k}}{1 + \sum_{k=1}^{N} (-1)^k a_k e^{-j\omega k}} \tag{5.4.17}$$

which corresponds to the difference equation

$$y(n) = -\sum_{k=1}^{N}(-1)^k a_k y(n-k) + \sum_{k=0}^{M}(-1)^k b_k x(n-k) \tag{5.4.18}$$

EXAMPLE 5.4.3

Convert the lowpass filter described by the difference equation

$$y(n) = 0.9y(n-1) + 0.1x(n)$$

into a highpass filter.

**Solution.** The difference equation for the highpass filter, according to (5.4.18), is

$$y(n) = -0.9y(n-1) + 0.1x(n)$$

and its frequency response is

$$H_{\text{hp}}(\omega) = \frac{0.1}{1 + 0.9e^{-j\omega}}$$

The reader may verify that $H_{\text{hp}}(\omega)$ is indeed highpass.

### 5.4.3 Digital Resonators

A *digital resonator* is a special two-pole bandpass filter with the pair of complex-conjugate poles located near the unit circle as shown in Fig. 5.4.6(a). The magnitude of the frequency response of the filter is shown in Fig. 5.4.6(b). The name resonator refers to the fact that the filter has a large magnitude response (i.e., it resonates) in the vicinity of the pole location. The angular position of the pole determines the resonant frequency of the filter. Digital resonators are useful in many applications, including simple bandpass filtering and speech generation.

In the design of a digital resonator with a resonant peak at or near $\omega = \omega_0$, we select the complex-conjugate poles at

$$p_{1,2} = re^{\pm j\omega_0}, \qquad 0 < r < 1$$

In addition, we can select up to two zeros. Although there are many possible choices, two cases are of special interest. One choice is to locate the zeros at the origin. The other choice is to locate a zero at $z = 1$ and a zero at $z = -1$. This choice completely eliminates the response of the filter at frequencies $\omega = 0$ and $\omega = \pi$, and it is useful in many practical applications.

The system function of the digital resonator with zeros at the origin is

$$H(z) = \frac{b_0}{(1 - re^{j\omega_0}z^{-1})(1 - re^{-j\omega_0}z^{-1})} \tag{5.4.19}$$

$$H(z) = \frac{b_0}{1 - (2r\cos\omega_0)z^{-1} + r^2z^{-2}} \tag{5.4.20}$$

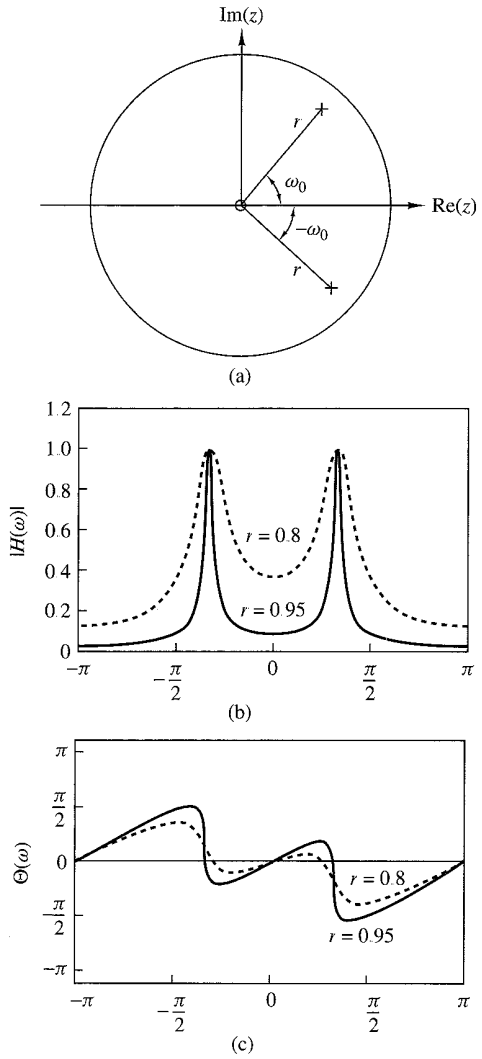



**Figure 5.4.6**
(a) Pole–zero pattern and (b) the corresponding magnitude and phase response of a digital resonator with (1) $r = 0.8$ and (2) $r = 0.95$.

Since $|H(\omega)|$ has its peak at or near $\omega = \omega_0$, we select the gain $b_0$ so that $|H(\omega_0)| = 1$. From (5.4.19) we obtain

$$\begin{aligned} H(\omega_0) &= \frac{b_0}{(1 - re^{j\omega_0}e^{-j\omega_0})(1 - re^{-j\omega_0}e^{-j\omega_0})} \\ &= \frac{b_0}{(1 - r)(1 - re^{-j2\omega_0})} \end{aligned} \tag{5.4.21}$$

and hence

$$|H(\omega_0)| = \frac{b_0}{(1 - r)\sqrt{1 + r^2 - 2r\cos 2\omega_0}} = 1$$

Thus the desired normalization factor is

$$b_0 = (1 - r)\sqrt{1 + r^2 - 2r\cos 2\omega_0} \tag{5.4.22}$$

The frequency response of the resonator in (5.4.19) can be expressed as

$$\begin{aligned} |H(\omega)| &= \frac{b_0}{U_1(\omega)U_2(\omega)} \\ \Theta(\omega) &= 2\omega - \Phi_1(\omega) - \Phi_2(\omega) \end{aligned} \tag{5.4.23}$$

where $U_1(\omega)$ and $U_2(\omega)$ are the magnitudes of the vectors from $p_1$ and $p_2$ to the point $\omega$ in the unit circle and $\Phi_1(\omega)$ and $\Phi_2(\omega)$ are the corresponding angles of these two vectors. The magnitudes $U_1(\omega)$ and $U_2(\omega)$ may be expressed as

$$\begin{aligned} U_1(\omega) &= \sqrt{1 + r^2 - 2r\cos(\omega_0 - \omega)} \\ U_2(\omega) &= \sqrt{1 + r^2 - 2r\cos(\omega_0 + \omega)} \end{aligned} \tag{5.4.24}$$

For any value of $r$, $U_1(\omega)$ takes its minimum value $(1 - r)$ at $\omega = \omega_0$. The product $U_1(\omega)U_2(\omega)$ reaches a minimum value at the frequency

$$\omega_r = \cos^{-1}\left(\frac{1 + r^2}{2r}\cos\omega_0\right) \tag{5.4.25}$$

which defines precisely the resonant frequency of the filter. We observe that when $r$ is very close to unity, $\omega_r \approx \omega_0$, which is the angular position of the pole. We also observe that as $r$ approaches unity, the resonance peak becomes sharper because $U_1(\omega)$ changes more rapidly in relative size in the vicinity of $\omega_0$. A quantitative measure of the sharpness of the resonance is provided by the 3-dB bandwidth $\Delta\omega$ of the filter. For values of $r$ close to unity,

$$\Delta\omega \approx 2(1 - r) \tag{5.4.26}$$

Figure 5.4.6 illustrates the magnitude and phase of digital resonators with $\omega_0 = \pi/3$, $r = 0.8$ and $\omega_0 = \pi/3$, $r = 0.95$. We note that the phase response undergoes its greatest rate of change near the resonant frequency.

If the zeros of the digital resonator are placed at $z = 1$ and $z = -1$, the resonator has the system function

$$\begin{aligned} H(z) &= G\frac{(1 - z^{-1})(1 + z^{-1})}{(1 - re^{j\omega_0}z^{-1})(1 - re^{-j\omega_0}z^{-1})} \\ &= G\frac{1 - z^{-2}}{1 - (2r\cos\omega_0)z^{-1} + r^2z^{-2}} \end{aligned} \tag{5.4.27}$$

and a frequency response characteristic

$$H(\omega) = b_0 \frac{1 - e^{-j2\omega}}{[1 - re^{j(\omega_0 - \omega)}][1 - re^{-j(\omega_0 + \omega)}]} \tag{5.4.28}$$

We observe that the zeros at $z = \pm 1$ affect both the magnitude and phase response of the resonator. For example, the magnitude response is

$$|H(\omega)| = b_0 \frac{N(\omega)}{U_1(\omega)U_2(\omega)} \tag{5.4.29}$$

where $N(\omega)$ is defined as

$$N(\omega) = \sqrt{2(1 - \cos 2\omega)}$$

Due to the presence of the zero factor, the resonant frequency is altered from that given by the expression in (5.4.25). The bandwidth of the filter is also altered. Although exact values for these two parameters are rather tedious to derive, we can easily compute the frequency response in (5.4.28) and compare the result with the previous case in which the zeros are located at the origin.

Figure 5.4.7 illustrates the magnitude and phase characteristics for $\omega_0 = \pi/3$, $r = 0.8$ and $\omega_0 = \pi/3$, $r = 0.95$. We observe that this filter has a slightly smaller bandwidth than the resonator, which has zeros at the origin. In addition, there appears to be a very small shift in the resonant frequency due to the presence of the zeros.

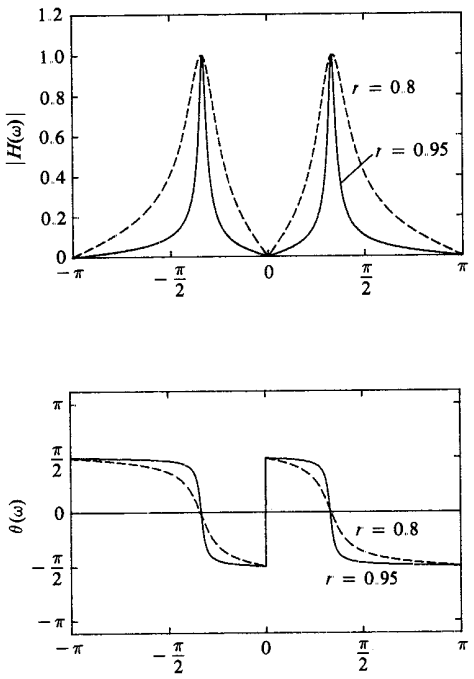



**Figure 5.4.7**
Magnitude and phase response of digital resonator with zeros at $\omega = 0$ and $\omega = \pi$ and (1) $r = 0.8$ and (2) $r = 0.95$.

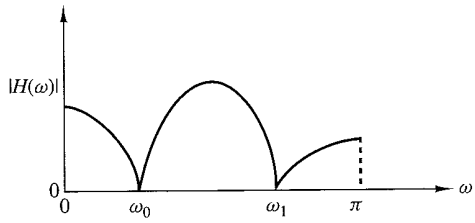



**Figure 5.4.8**
Frequency response characteristic of a notch filter.

### 5.4.4 Notch Filters

A notch filter is a filter that contains one or more deep notches or, ideally, perfect nulls in its frequency response characteristic. Figure 5.4.8 illustrates the frequency response characteristic of a notch filter with nulls at frequencies $\omega_0$ and $\omega_1$. Notch filters are useful in many applications where specific frequency components must be eliminated. For example, instrumentation and recording systems require that the power-line frequency of 60 Hz and its harmonics be eliminated.

To create a null in the frequency response of a filter at a frequency $\omega_0$, we simply introduce a pair of complex-conjugate zeros on the unit circle at an angle $\omega_0$. That is,

$$z_{1,2} = e^{\pm j\omega_0}$$

Thus the system function for an FIR notch filter is simply

$$\begin{aligned} H(z) &= b_0(1 - e^{j\omega_0}z^{-1})(1 - e^{-j\omega_0}z^{-1}) \\ &= b_0(1 - 2\cos\omega_0 z^{-1} + z^{-2}) \end{aligned} \tag{5.4.30}$$

As an illustration, Fig. 5.4.9 shows the magnitude response for a notch filter having a null at $\omega = \pi/4$.

The problem with the FIR notch filter is that the notch has a relatively large bandwidth, which means that other frequency components around the desired null are severely attenuated. To reduce the bandwidth of the null, we can resort to a more sophisticated, longer FIR filter designed according to criteria described in Chapter 10. Alternatively, we could, in an ad hoc manner, attempt to improve on the frequency response characteristics by introducing poles in the system function.

Suppose that we place a pair of complex-conjugate poles at

$$p_{1,2} = re^{\pm j\omega_0}$$

The effect of the poles is to introduce a resonance in the vicinity of the null and thus to reduce the bandwidth of the notch. The system function for the resulting filter is

$$H(z) = b_0\frac{1 - 2\cos\omega_0 z^{-1} + z^{-2}}{1 - 2r\cos\omega_0 z^{-1} + r^2 z^{-2}} \tag{5.4.31}$$

The magnitude response $|H(\omega)|$ of the filter in (5.4.31) is plotted in Fig. 5.4.10 for $\omega_0 = \pi/4$, $r = 0.85$, and for $\omega_0 = \pi/4$, $r = 0.95$. When compared with the frequency response of the FIR filter in Fig. 5.4.9, we note that the effect of the poles is to reduce the bandwidth of the notch.

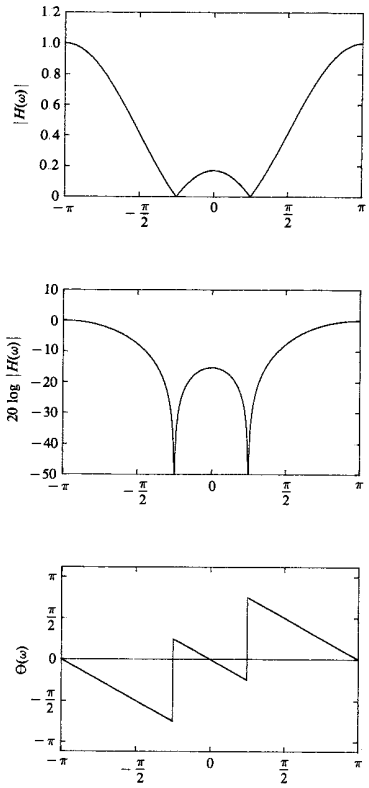



**Figure 5.4.9**
Frequency response characteristics of a notch filter with a notch at $\omega = \pi/4$ or $f = 1/8$; $H(z) = G[1 - 2\cos\omega_0 z^{-1} + z^{-2}]$.

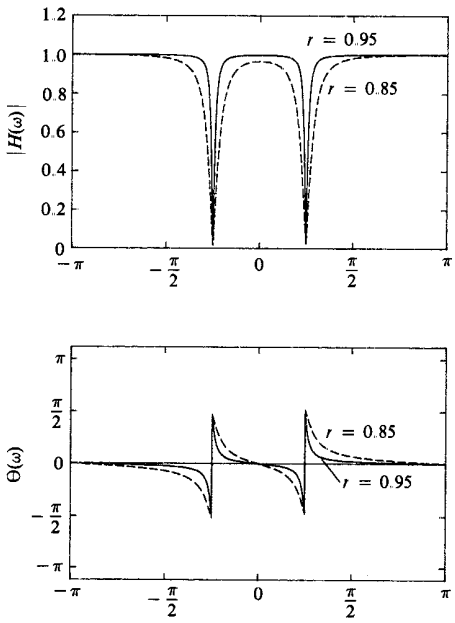



**Figure 5.4.10**
Frequency response characteristics of two notch filters with poles at (1) $r = 0.85$ and (2) $r = 0.95$; $H(z) = b_0[(1-2\cos\omega_0 z^{-1}+z^{-2})/(1-2r\cos\omega_0 z^{-1}+r^2 z^{-2})]$.

In addition to reducing the bandwidth of the notch, the introduction of a pole in the vicinity of the null may result in a small ripple in the passband of the filter due to the resonance created by the pole. The effect of the ripple can be reduced by introducing additional poles and/or zeros in the system function of the notch filter. The major problem with this approach is that it is basically an ad hoc, trial-and-error method.

### 5.4.5 Comb Filters

In its simplest form, a comb filter can be viewed as a notch filter in which the nulls occur periodically across the frequency band, hence the analogy to an ordinary comb that has periodically spaced teeth. Comb filters find applications in a wide range of practical systems such as in the rejection of power-line harmonics, in the separation of solar and lunar components from ionospheric measurements of electron concentration, and in the suppression of clutter from fixed objects in moving-target-indicator (MTI) radars.

To illustrate a simple form of a comb filter, consider a moving average (FIR) filter described by the difference equation

$$y(n) = \frac{1}{M+1}\sum_{k=0}^{M} x(n-k) \tag{5.4.32}$$

The system function of this FIR filter is

$$\begin{aligned} H(z) &= \frac{1}{M+1}\sum_{k=0}^{M} z^{-k} \\ &= \frac{1}{M+1}\frac{[1-z^{-(M+1)}]}{(1-z^{-1})} \end{aligned} \tag{5.4.33}$$

and its frequency response is

$$H(\omega) = \frac{e^{-j\omega M/2}}{M+1}\frac{\sin\omega(\frac{M+1}{2})}{\sin(\omega/2)} \tag{5.4.34}$$

From (5.4.33) we observe that the filter has zeros on the unit circle at

$$z = e^{j2\pi k/(M+1)}, \qquad k = 1, 2, 3, \ldots, M \tag{5.4.35}$$

Note that the pole at $z = 1$ is actually canceled by the zero at $z = 1$, so that in effect the FIR filter does not contain poles outside $z = 0$.

A plot of the magnitude characteristic of (5.4.34) clearly illustrates the existence of the periodically spaced zeros in frequency at $\omega_k = 2\pi k/(M+1)$ for $k = 1, 2, \ldots, M$. Figure 5.4.11 shows $|H(\omega)|$ for $M = 10$.

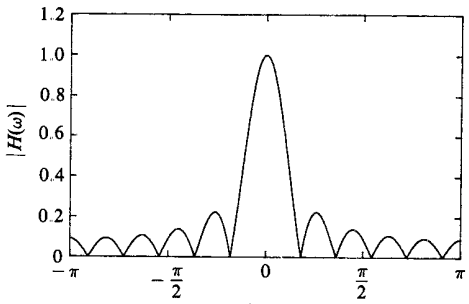



**Figure 5.4.11**
Magnitude response characteristic for the comb filter given by (5.4.34) with $M = 10$.

In more general terms, we can create a comb filter by taking an FIR filter with system function

$$H(z) = \sum_{k=0}^{M} h(k)z^{-k} \tag{5.4.36}$$

and replacing $z$ by $z^L$, where $L$ is a positive integer. Thus the new FIR filter has a system function

$$H_L(z) = \sum_{k=0}^{M} h(k)z^{-kL} \tag{5.4.37}$$

If the frequency response of the original FIR filter is $H(\omega)$, the frequency response of the FIR in (5.4.37) is

$$\begin{aligned} H_L(\omega) &= \sum_{k=0}^{M} h(k)e^{-jkL\omega} \\ &= H(L\omega) \end{aligned} \tag{5.4.38}$$

Consequently, the frequency response characteristic $H_L(\omega)$ is simply an $L$-order repetition of $H(\omega)$ in the range $0 \le \omega \le 2\pi$. Figure 5.4.12 illustrates the relationship between $H_L(\omega)$ and $H(\omega)$ for $L = 5$.

Now, suppose that the original FIR filter with system function $H(z)$ has a spectral null (i.e., a zero), at some frequency $\omega_0$. Then the filter with system function $H_L(z)$ has periodically spaced nulls at $\omega_k = \omega_0 + 2\pi k/L$, $k = 0, 1, 2, \ldots, L-1$. As an illustration, Fig. 5.4.13 shows an FIR comb filter with $M = 3$ and $L = 3$. This FIR filter can be viewed as an FIR filter of length 10, but only four of the 10 filter coefficients are nonzero.

Let us now return to the moving average filter with system function given by (5.4.33). Suppose that we replace $z$ by $z^L$. Then the resulting comb filter has the system function

$$H_L(z) = \frac{1}{M+1}\,\frac{1 - z^{-L(M+1)}}{1 - z^{-L}} \tag{5.4.39}$$

and a frequency response

$$H_L(\omega) = \frac{1}{M+1}\,\frac{\sin[\omega L(M+1)/2]}{\sin(\omega L/2)}\,e^{-j\omega LM/2} \tag{5.4.40}$$

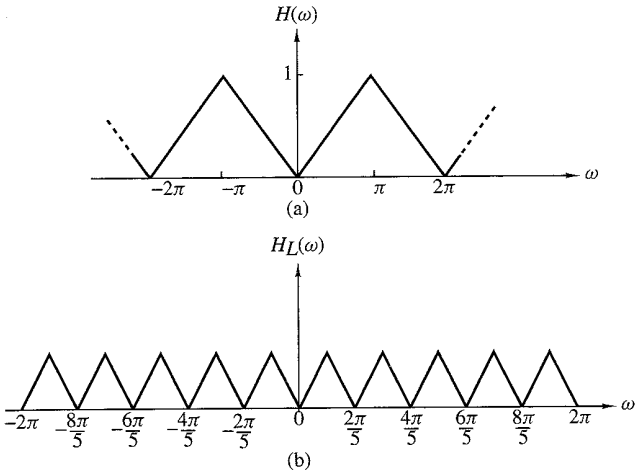



**Figure 5.4.12** Comb filter with frequency response $H_L(\omega)$ obtained from $H(\omega)$.

This filter has zeros on the unit circle at

$$z_k = e^{j2\pi k/L(M+1)} \tag{5.4.41}$$

for all integer values of $k$ except $k = 0, L, 2L, \ldots, ML$. Figure 5.4.14 illustrates $|H_L(\omega)|$ for $L = 3$ and $M = 10$.

The comb filter described by (5.4.39) finds application in the separation of solar and lunar spectral components in ionospheric measurements of electron concentration as described in the paper by Bernhardt et al. (1976). The solar period is $T_s = 24$ hours and results in a solar component of one cycle per day and its harmonics. The lunar period is $T_L = 24.84$ hours and provides spectral lines at 0.96618 cycle per day and its harmonics. Figure 5.4.15(a) shows a plot of the power density spectrum of the unfiltered ionospheric measurements of the electron concentration. Note that the weak lunar spectral components are almost hidden by the strong solar spectral components.

The two sets of spectral components can be separated by the use of comb filters. If we wish to obtain the solar components, we can use a comb filter with a narrow passband at multiples of one cycle per day. This can be achieved by selecting $L$ such that $F_s/L = 1$ cycle per day, where $F_s$ is the corresponding sampling frequency. The

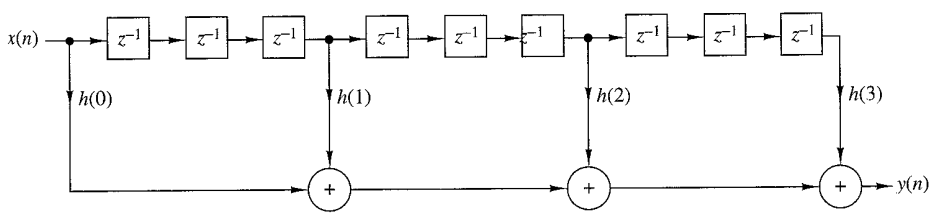



**Figure 5.4.13** Realization of an FIR comb filter having $M = 3$ and $L = 3$.

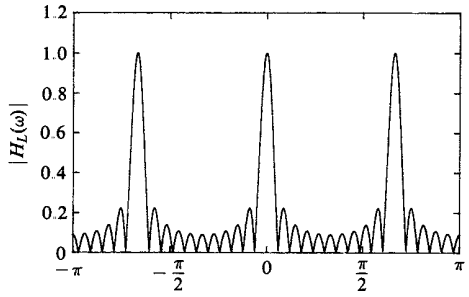



**Figure 5.4.14** Magnitude response characteristic for a comb filter given by (5.4.40), with $L = 3$ and $M = 10$.

result is a filter that has peaks in its frequency response at multiples of one cycle per day. By selecting $M = 58$, the filter will have nulls at multiples of $(F_s/L)/(M+1) = 1/59$ cycle per day. These nulls are very close to the lunar components and result in good rejection. Figure 5.4.15(b) illustrates the power spectral density of the output of the comb filter that isolates the solar components. A comb filter that rejects the solar components and passes the lunar components can be designed in a similar manner. Figure 5.4.15(c) illustrates the power spectral density at the output of such a lunar filter.

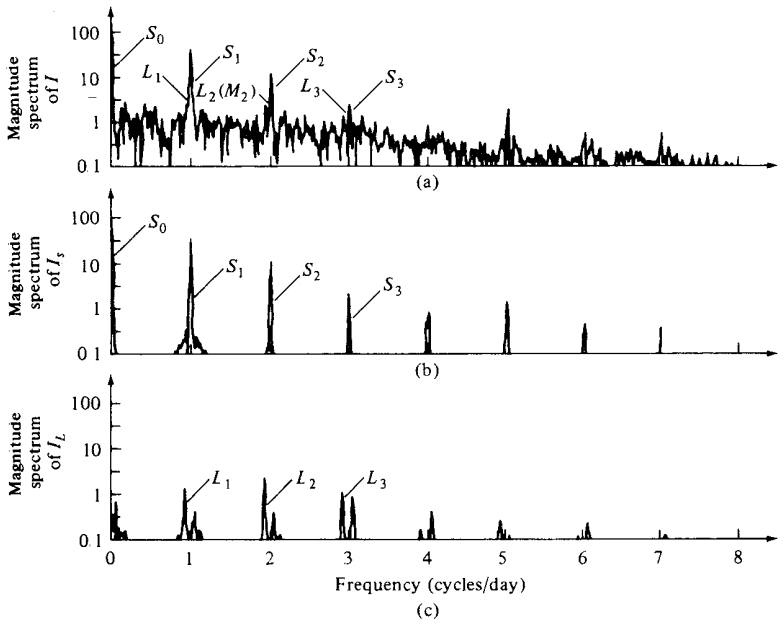



**Figure 5.4.15** (a) Spectrum of unfiltered electron content data; (b) spectrum of output of solar filter; (c) spectrum of output of lunar filter. [From paper by Bernhardt et al. (1976). Reprinted with permission of the American Geophysical Union.]

### 5.4.6 All-Pass Filters

An all-pass filter is defined as a system that has a constant magnitude response for all frequencies, that is,

$$|H(\omega)| = 1, \qquad 0 \le \omega \le \pi \tag{5.4.42}$$

The simplest example of an all-pass filter is a pure delay system with system function

$$H(z) = z^{-k}$$

This system passes all signals without modification except for a delay of $k$ samples. This is a trivial all-pass system that has a linear phase response characteristic.

A more interesting all-pass filter is described by the system function

$$\begin{aligned} H(z) &= \frac{a_N + a_{N-1}z^{-1} + \cdots + a_1 z^{-N+1} + z^{-N}}{1 + a_1 z^{-1} + \cdots + a_N z^{-N}} \\ &= \frac{\sum_{k=0}^{N} a_k z^{-N+k}}{\sum_{k=0}^{N} a_k z^{-k}}, \qquad a_0 = 1 \end{aligned} \tag{5.4.43}$$

where all the filter coefficients $\{a_k\}$ are real. If we define the polynomial $A(z)$ as

$$A(z) = \sum_{k=0}^{N} a_k z^{-k}, \qquad a_0 = 1$$

then (5.4.43) can be expressed as

$$H(z) = z^{-N}\frac{A(z^{-1})}{A(z)} \tag{5.4.44}$$

Since

$$|H(\omega)|^2 = H(z)H(z^{-1})|_{z=e^{j\omega}} = 1$$

the system given by (5.4.44) is an all-pass system. Furthermore, if $z_0$ is a pole of $H(z)$, then $1/z_0$ is a zero of $H(z)$ (i.e., the poles and zeros are reciprocals of one another). Figure 5.4.16 illustrates typical pole–zero patterns for a single-pole, single-zero filter and a two-pole, two-zero filter. A plot of the phase characteristics of these filters is shown in Fig. 5.4.17 for $a = 0.6$ and $r = 0.9$, $\omega_0 = \pi/4$.

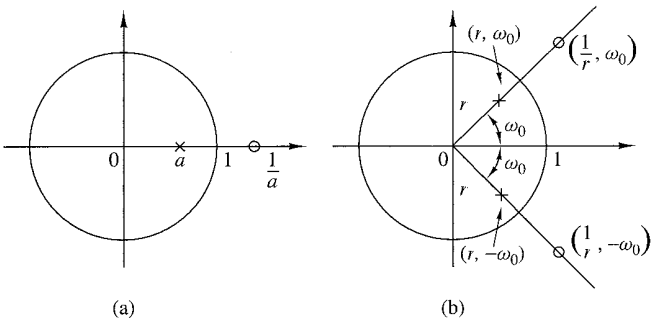



**Figure 5.4.16** Pole–zero patterns of (a) a first-order and (b) a second-order all-pass filter.

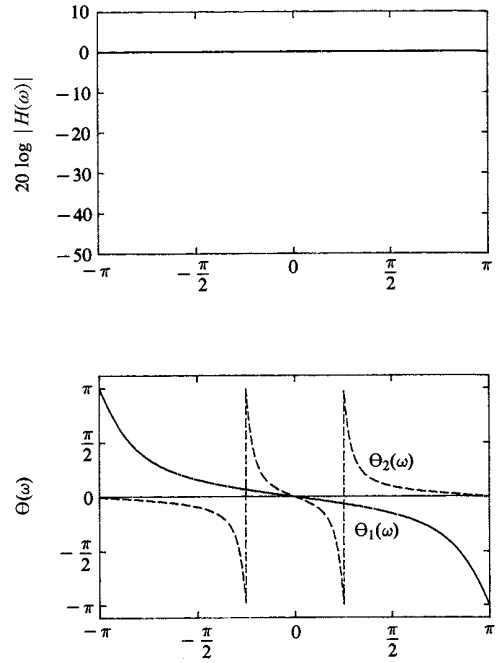



**Figure 5.4.17** Frequency response characteristics of an all-pass filter with system functions (1) $H(z) = (0.6 + z^{-1})/(1 + 0.6z^{-1})$, (2) $H(z) = (r^2 - 2r\cos\omega_0 z^{-1} + z^{-2})/(1 - 2r\cos\omega_0 z^{-1} + r^2 z^{-2})$, $r = 0.9$, $\omega_0 = \pi/4$.

The most general form for the system function of an all-pass system with real coefficients, expressed in factored form in terms of poles and zeros, is

$$H_{\text{ap}}(z) = \prod_{k=1}^{N_R} \frac{z^{-1} - \alpha_k}{1 - \alpha_k z^{-1}} \prod_{k=1}^{N_c} \frac{(z^{-1} - \beta_k)(z^{-1} - \beta_k^*)}{(1 - \beta_k z^{-1})(1 - \beta_k^* z^{-1})} \tag{5.4.45}$$

where there are $N_R$ real poles and zeros and $N_c$ complex-conjugate pairs of poles and zeros. For causal and stable systems we require that $-1 < \alpha_k < 1$ and $|\beta_k| < 1$.

Expressions for the phase response and group delay of all-pass systems can easily be obtained using the method described in Section 5.2.1. For a single pole–single zero all-pass system we have

$$H_{\text{ap}}(\omega) = \frac{e^{j\omega} - re^{-j\theta}}{1 - re^{j\theta}e^{-j\omega}}$$

Hence

$$\Theta_{\text{ap}}(\omega) = -\omega - 2\tan^{-1}\frac{r\sin(\omega - \theta)}{1 - r\cos(\omega - \theta)}$$

and

$$\tau_g(\omega) = -\frac{d\Theta_{\text{ap}}(\omega)}{d\omega} = \frac{1 - r^2}{1 + r^2 - 2r\cos(\omega - \theta)} \tag{5.4.46}$$

We note that for a causal and stable system, $r < 1$ and hence $\tau_g(\omega) \geq 0$. Since the group delay of a higher-order pole–zero system consists of a sum of positive terms as in (5.4.46), the group delay will always be positive.

All-pass filters find application as phase equalizers. When placed in cascade with a system that has an undesired phase response, a phase equalizer is designed to compensate for the poor phase characteristics of the system and therefore to produce an overall linear-phase response.

### 5.4.7 Digital Sinusoidal Oscillators

A *digital sinusoidal oscillator* can be viewed as a limiting form of a two-pole resonator for which the complex-conjugate poles lie on the unit circle. From our previous discussion of second-order systems, we recall that a system with system function

$$H(z) = \frac{b_0}{1 + a_1 z^{-1} + a_2 z^{-2}} \tag{5.4.47}$$

and parameters

$$a_1 = -2r\cos\omega_0 \quad \text{and} \quad a_2 = r^2 \tag{5.4.48}$$

has complex-conjugate poles at $p = re^{\pm j\omega_0}$, and a unit sample response

$$h(n) = \frac{b_0 r^n}{\sin\omega_0}\sin(n+1)\omega_0 u(n) \tag{5.4.49}$$

If the poles are placed on the unit circle ($r = 1$) and $b_0$ is set to $A\sin\omega_0$, then

$$h(n) = A\sin(n+1)\omega_0 u(n) \tag{5.4.50}$$

Thus the impulse response of the second-order system with complex-conjugate poles on the unit circle is a sinusoid and the system is called a digital sinusoidal oscillator or a *digital sinusoidal generator.*

A digital sinusoidal generator is a basic component of a digital frequency synthesizer.

The block diagram representation of the system function given by (5.4.47) is illustrated in Fig. 5.4.18. The corresponding difference equation for this system is

$$y(n) = -a_1 y(n-1) - y(n-2) + b_0\delta(n) \tag{5.4.51}$$

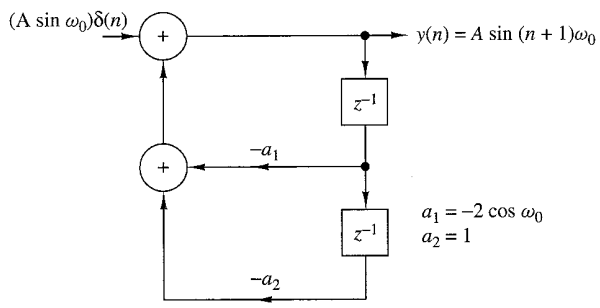



**Figure 5.4.18**
Digital sinusoidal generator.

where the parameters are $a_1 = -2\cos\omega_0$ and $b_0 = A\sin\omega_0$, and the initial conditions are $y(-1) = y(-2) = 0$. By iterating the difference equation in (5.4.51), we obtain

$$\begin{aligned}
y(0) &= A\sin\omega_0 \\
y(1) &= 2\cos\omega_0 y(0) = 2A\sin\omega_0\cos\omega_0 = A\sin 2\omega_0 \\
y(2) &= 2\cos\omega_0 y(1) - y(0) \\
&= 2A\cos\omega_0\sin 2\omega_0 - A\sin\omega_0 \\
&= A(4\cos^2\omega_0 - 1)\sin\omega_0 \\
&= 3A\sin\omega_0 - 4\sin^3\omega_0 = A\sin 3\omega_0
\end{aligned}$$

and so forth. We note that the application of the impulse at $n = 0$ serves the purpose of beginning the sinusoidal oscillation. Thereafter, the oscillation is self-sustaining because the system has no damping (i.e., $r = 1$).

It is interesting to note that the sinusoidal oscillation obtained from the system in (5.4.51) can also be obtained by setting the input to zero and setting the initial conditions to $y(-1) = 0$, $y(-2) = -A\sin\omega_0$. Thus the zero-input response to the second-order system described by the homogeneous difference equation

$$y(n) = -a_1 y(n-1) - y(n-2) \tag{5.4.52}$$

with initial conditions $y(-1) = 0$ and $y(-2) = -A\sin\omega_0$, is exactly the same as the response of (5.4.51) to an impulse excitation. In fact, the difference equation in (5.4.52) can be obtained directly from the trigonometric identity

$$\sin\alpha + \sin\beta = 2\sin\frac{\alpha+\beta}{2}\cos\frac{\alpha-\beta}{2} \tag{5.4.53}$$

where, by definition, $\alpha = (n+1)\omega_0$, $\beta = (n-1)\omega_0$, and $y(n) = \sin(n+1)\omega_0$.

In some practical applications involving modulation of two sinusoidal carrier signals in phase quadrature, there is a need to generate the sinusoids $A\sin\omega_0 n$ and $A\cos\omega_0 n$. These signals can be generated from the so-called *coupled-form oscillator,* which can be obtained from the trigonometric formulas

$$\begin{aligned}
\cos(\alpha+\beta) &= \cos\alpha\cos\beta - \sin\alpha\sin\beta \\
\sin(\alpha+\beta) &= \sin\alpha\cos\beta + \cos\alpha\sin\beta
\end{aligned}$$

where, by definition, $\alpha = n\omega_0$, $\beta = \omega_0$, and

$$y_c(n) = \cos n\omega_0 u(n) \tag{5.4.54}$$

$$y_s(n) = \sin n\omega_0 u(n) \tag{5.4.55}$$

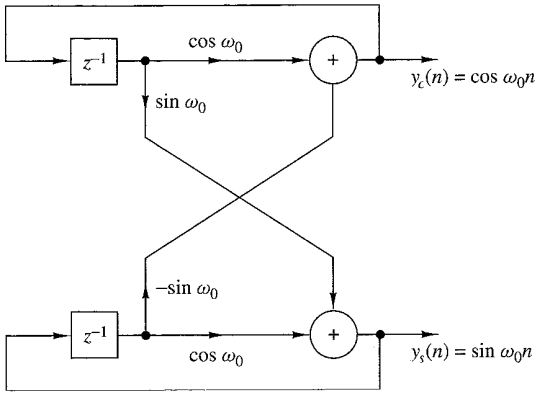



**Figure 5.4.19** Realization of the coupled-form oscillator.

Thus we obtain the two coupled difference equations

$$y_c(n) = (\cos\omega_0)y_c(n-1) - (\sin\omega_0)y_s(n-1) \tag{5.4.56}$$

$$y_s(n) = (\sin\omega_0)y_c(n-1) + (\cos\omega_0)y_s(n-1) \tag{5.4.57}$$

which can also be expressed in matrix form as

$$\begin{bmatrix} y_c(n) \\ y_s(n) \end{bmatrix} = \begin{bmatrix} \cos\omega_0 & -\sin\omega_0 \\ \sin\omega_0 & \cos\omega_0 \end{bmatrix} \begin{bmatrix} y_c(n-1) \\ y_s(n-1) \end{bmatrix} \tag{5.4.58}$$

The structure for the realization of the coupled-form oscillator is illustrated in Fig. 5.4.19. We note that this is a two-output system which is not driven by any input, but which requires the initial conditions $y_c(-1) = A\cos\omega_0$ and $y_s(-1) = -A\sin\omega_0$ in order to begin its self-sustaining oscillations.

Finally, it is interesting to note that (5.4.58) corresponds to vector rotation in the two-dimensional coordinate system with coordinates $y_c(n)$ and $y_s(n)$. As a consequence, the coupled-form oscillator can also be implemented by use of the so-called CORDIC algorithm [see the book by Kung et al. (1985)].

## 5.5 Inverse Systems and Deconvolution

As we have seen, a linear time-invariant system takes an input signal $x(n)$ and produces an output signal $y(n)$, which is the convolution of $x(n)$ with the unit sample response $h(n)$ of the system. In many practical applications we are given an output signal from a system whose characteristics are unknown and we are asked to determine the input signal. For example, in the transmission of digital information at high data rates over telephone channels, it is well known that the channel distorts the signal and causes intersymbol interference among the data symbols. The intersymbol interference may cause errors when we attempt to recover the data. In such a case the problem is to design a corrective system which, when cascaded with the channel, produces an output that, in some sense, corrects for the distortion caused

by the channel, and thus yields a replica of the desired transmitted signal. In digital communications such a corrective system is called an *equalizer*. In the general context of linear systems theory, however, we call the corrective system an *inverse system*, because the corrective system has a frequency response which is basically the reciprocal of the frequency response of the system that caused the distortion. Furthermore, since the distortive system yields an output $y(n)$ that is the convolution of the input $x(n)$ with the impulse response $h(n)$, the inverse system operation that takes $y(n)$ and produces $x(n)$ is called *deconvolution*.

If the characteristics of the distortive system are unknown, it is often necessary, when possible, to excite the system with a known signal, observe the output, compare it with the input, and in some manner, determine the characteristics of the system. For example, in the digital communication problem just described, where the frequency response of the channel is unknown, the measurement of the channel frequency response can be accomplished by transmitting a set of equal-amplitude sinusoids, at different frequencies with a specified set of phases, within the frequency band of the channel. The channel will attenuate and phase shift each of the sinusoids. By comparing the received signal with the transmitted signal, the receiver obtains a measurement of the channel frequency response which can be used to design the inverse system. The process of determining the characteristics of the unknown system, either $h(n)$ or $H(\omega)$, by a set of measurements performed on the system is called *system identification*.

The term "deconvolution" is often used in seismic signal processing, and more generally, in geophysics to describe the operation of separating the input signal from the characteristics of the system which we wish to measure. The deconvolution operation is actually intended to identify the characteristics of the system, which in this case, is the earth, and can also be viewed as a system identification problem. The "inverse system," in this case, has a frequency response that is the reciprocal of the input signal spectrum that has been used to excite the system.

### 5.5.1 Invertibility of Linear Time-Invariant Systems

A system is said to be *invertible* if there is a one-to-one correspondence between its input and output signals. This definition implies that if we know the output sequence $y(n)$, $-\infty < n < \infty$, of an invertible system $\mathcal{T}$, we can uniquely determine its input $x(n)$, $-\infty < n < \infty$. The *inverse system* with input $y(n)$ and output $x(n)$ is denoted by $\mathcal{T}^{-1}$. Clearly, the cascade connection of a system and its inverse is equivalent to the identity system, since

$$w(n) = \mathcal{T}^{-1}[y(n)] = \mathcal{T}^{-1}\{\mathcal{T}[x(n)]\} = x(n) \tag{5.5.1}$$

as illustrated in Fig. 5.5.1. For example, the systems defined by the input–output relations $y(n) = ax(n)$ and $y(n) = x(n-5)$ are invertible, whereas the input–output relations $y(n) = x^2(n)$ and $y(n) = 0$ represent noninvertible systems.

As indicated above, inverse systems are important in many practical applications, including geophysics and digital communications. Let us begin by considering the problem of determining the inverse of a given system. We limit our discussion to the class of linear time-invariant discrete-time systems.

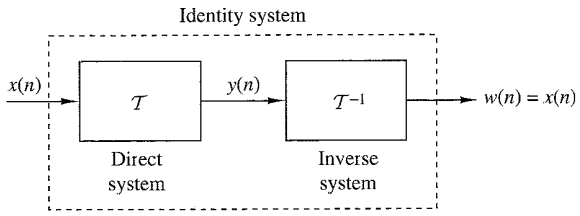



**Figure 5.5.1**
System $\mathcal{T}$ in cascade with its inverse $\mathcal{T}^{-1}$.

Now, suppose that the linear time-invariant system $\mathcal{T}$ has an impulse response $h(n)$ and let $h_I(n)$ denote the impulse response of the inverse system $\mathcal{T}^{-1}$. Then (5.5.1) is equivalent to the convolution equation

$$w(n) = h_I(n) * h(n) * x(n) = x(n) \tag{5.5.2}$$

But (5.5.2) implies that

$$h(n) * h_I(n) = \delta(n) \tag{5.5.3}$$

The convolution equation in (5.5.3) can be used to solve for $h_I(n)$ for a given $h(n)$. However, the solution of (5.5.3) in the time domain is usually difficult. A simpler approach is to transform (5.5.3) into the $z$-domain and solve for $\mathcal{T}^{-1}$. Thus in the $z$-transform domain, (5.5.3) becomes

$$H(z)H_I(z) = 1$$

and therefore the system function for the inverse system is

$$H_I(z) = \frac{1}{H(z)} \tag{5.5.4}$$

If $H(z)$ has a rational system function

$$H(z) = \frac{B(z)}{A(z)} \tag{5.5.5}$$

then

$$H_I(z) = \frac{A(z)}{B(z)} \tag{5.5.6}$$

Thus the zeros of $H(z)$ become the poles of the inverse system, and vice versa. Furthermore, if $H(z)$ is an FIR system, then $H_I(z)$ is an all-pole system, or if $H(z)$ is an all-pole system, then $H_I(z)$ is an FIR system.

### EXAMPLE 5.5.1

Determine the inverse of the system with impulse response

$$h(n) = (\frac{1}{2})^n u(n)$$

**Solution.** The system function corresponding to $h(n)$ is

$$H(z) = \frac{1}{1 - \frac{1}{2}z^{-1}}, \qquad \text{ROC: } |z| > \frac{1}{2}$$

This system is both causal and stable. Since $H(z)$ is an all-pole system, its inverse is FIR and is given by the system function

$$H_I(z) = 1 - \frac{1}{2}z^{-1}$$

Hence its impulse response is

$$h_I(n) = \delta(n) - \frac{1}{2}\delta(n-1)$$

---

EXAMPLE 5.5.2

Determine the inverse of the system with impulse response

$$h(n) = \delta(n) - \frac{1}{2}\delta(n-1)$$

This is an FIR system and its system function is

$$H(z) = 1 - \frac{1}{2}z^{-1}, \qquad \text{ROC: } |z| > 0$$

The inverse system has the system function

$$H_I(z) = \frac{1}{H(z)} = \frac{1}{1 - \frac{1}{2}z^{-1}} = \frac{z}{z - \frac{1}{2}}$$

Thus $H_I(z)$ has a zero at the origin and a pole at $z = \frac{1}{2}$. In this case there are two possible regions of convergence and hence two possible inverse systems, as illustrated in Fig. 5.5.2. If we take the ROC of $H_I(z)$ as $|z| > \frac{1}{2}$, the inverse transform yields

$$h_I(n) = (\frac{1}{2})^n u(n)$$

which is the impulse response of a causal and stable system. On the other hand, if the ROC is assumed to be $|z| < \frac{1}{2}$, the inverse system has an impulse response

$$h_I(n) = -\left(\frac{1}{2}\right)^n u(-n-1)$$

In this case the inverse system is anticausal and unstable.

---

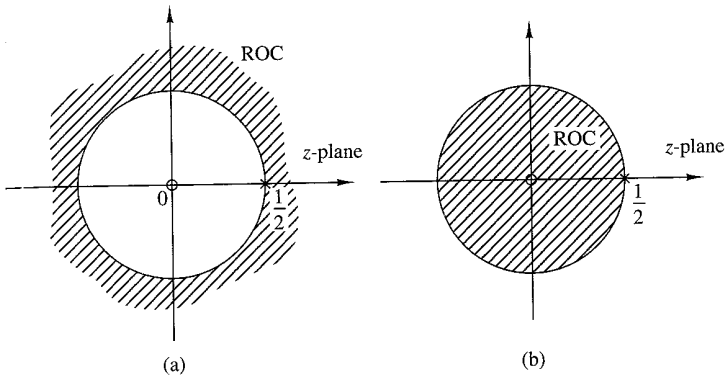



**Figure 5.5.2** Two possible regions of convergence for $H(z) = z/(z - \frac{1}{2})$.

We observe that (5.5.3) cannot be solved uniquely by using (5.5.6) unless we specify the region of convergence for the system function of the inverse system.

In some practical applications the impulse response $h(n)$ does not possess a $z$-transform that can be expressed in closed form. As an alternative we may solve (5.5.3) directly using a digital computer. Since (5.5.3) does not, in general, possess a unique solution, we assume that the system and its inverse are causal. Then (5.5.3) simplifies to the equation

$$\sum_{k=0}^{n} h(k)h_I(n-k) = \delta(n) \tag{5.5.7}$$

By assumption, $h_I(n) = 0$ for $n < 0$. For $n = 0$ we obtain

$$h_I(0) = 1/h(0) \tag{5.5.8}$$

The values of $h_I(n)$ for $n \geq 1$ can be obtained recursively from the equation

$$h_I(n) = \sum_{k=1}^{n} \frac{h(n)h_I(n-k)}{h(0)}, \qquad n \geq 1 \tag{5.5.9}$$

This recursive relation can easily be programmed on a digital computer.

There are two problems associated with (5.5.9). First, the method does not work if $h(0) = 0$. However, this problem can easily be remedied by introducing an appropriate delay in the right-hand side of (5.5.7), that is, by replacing $\delta(n)$ by $\delta(n-m)$, where $m = 1$ if $h(0) = 0$ and $h(1) \neq 0$, and so on. Second, the recursion in (5.5.9) gives rise to round-off errors which grow with $n$ and, as a result, the numerical accuracy of $h(n)$ deteriorates for large $n$.

EXAMPLE 5.5.3

Determine the causal inverse of the FIR system with impulse response

$$h(n) = \delta(n) - \alpha\delta(n-1)$$

Since $h(0) = 1$, $h(1) = -\alpha$, and $h(n) = 0$ for $n \geq \alpha$, we have

$$h_I(0) = 1/h(0) = 1$$

and

$$h_I(n) = \alpha h_I(n-1), \qquad n \geq 1$$

Consequently,

$$h_I(1) = \alpha, \quad h_I(2) = \alpha^2, \quad \ldots, \quad h_I(n) = \alpha^n$$

which corresponds to a causal IIR system as expected.

---

### 5.5.2 Minimum-Phase, Maximum-Phase, and Mixed-Phase Systems

The invertibility of a linear time-invariant system is intimately related to the characteristics of the phase spectral function of the system. To illustrate this point, let us consider two FIR systems, characterized by the system functions

$$H_1(z) = 1 + \frac{1}{2}z^{-1} = z^{-1}(z + \frac{1}{2}) \tag{5.5.10}$$

$$H_2(z) = \frac{1}{2} + z^{-1} = z^{-1}(\frac{1}{2}z + 1) \tag{5.5.11}$$

The system in (5.5.10) has a zero at $z = -\frac{1}{2}$ and an impulse response $h(0) = 1$, $h(1) = 1/2$. The system in (5.5.11) has a zero at $z = -2$ and an impulse response $h(0) = 1/2$, $h(1) = 1$, which is the reverse of the system in (5.5.10). This is due to the reciprocal relationship between the zeros of $H_1(z)$ and $H_2(z)$.

In the frequency domain, the two systems are characterized by their frequency response functions, which can be expressed as

$$|H_1(\omega)| = |H_2(\omega)| = \sqrt{\frac{5}{4} + \cos\omega} \tag{5.5.12}$$

and

$$\Theta_1(\omega) = -\omega + \tan^{-1}\frac{\sin\omega}{\frac{1}{2} + \cos\omega} \tag{5.5.13}$$

$$\Theta_2(\omega) = -\omega + \tan^{-1}\frac{\sin\omega}{2 + \cos\omega} \tag{5.5.14}$$

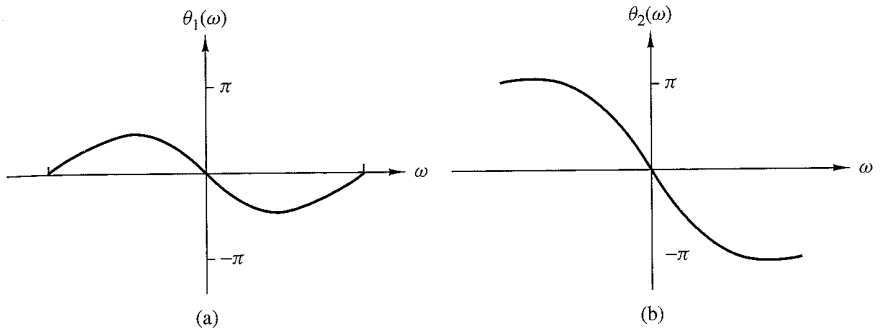



**Figure 5.5.3** Phase response characteristics for the systems in (5.5.10) and (5.5.11).

The magnitude characteristics for the two systems are identical because the zeros of $H_1(z)$ and $H_2(z)$ are reciprocals.

The graphs of $\Theta_1(\omega)$ and $\Theta_2(\omega)$ are illustrated in Fig. 5.5.3. We observe that the phase characteristic $\Theta_1(\omega)$ for the first system begins at zero phase at the frequency $\omega = 0$ and terminates at zero phase at the frequency $\omega = \pi$. Hence the net phase change, $\Theta_1(\pi) - \Theta_1(0)$, is zero. On the other hand, the phase characteristic for the system with the zero outside the unit circle undergoes a net phase change $\Theta_2(\pi) - \Theta_2(0) = \pi$ radians. As a consequence of these different phase characteristics, we call the first system a *minimum-phase system* and the second system a *maximum-phase system*.

These definitions are easily extended to an FIR system of arbitrary length. To be specific, an FIR system of length $M + 1$ has $M$ zeros. Its frequency response can be expressed as

$$H(\omega) = b_0(1 - z_1 e^{-j\omega})(1 - z_2 e^{-j\omega}) \cdots (1 - z_M e^{-j\omega}) \tag{5.5.15}$$

where $\{z_i\}$ denote the zeros and $b_0$ is an arbitrary constant. When all the zeros are inside the unit circle, each term in the product of (5.5.15), corresponding to a real-valued zero, will undergo a net phase change of zero between $\omega = 0$ and $\omega = \pi$. Also, each pair of complex-conjugate factors in $H(\omega)$ will undergo a net phase change of zero. Therefore,

$$\measuredangle H(\pi) - \measuredangle H(0) = 0 \tag{5.5.16}$$

and hence the system is called a minimum-phase system. On the other hand, when all the zeros are outside the unit circle, a real-valued zero will contribute a net phase change of $\pi$ radians as the frequency varies from $\omega = 0$ to $\omega = \pi$, and each pair of complex-conjugate zeros will contribute a net phase change of $2\pi$ radians over the same range of $\omega$. Therefore,

$$\measuredangle H(\pi) - \measuredangle H(0) = M\pi \tag{5.5.17}$$

which is the largest possible phase change for an FIR system with $M$ zeros. Hence the system is called maximum phase. It follows from the discussion above that

$$\measuredangle H_{\max}(\pi) \geq \measuredangle H_{\min}(\pi) \tag{5.5.18}$$

If the FIR system with $M$ zeros has some of its zeros inside the unit circle and the remaining zeros outside the unit circle, it is called a *mixed-phase system* or a *nonminimum-phase system*.

Since the derivative of the phase characteristic of the system is a measure of the time delay that signal frequency components undergo in passing through the system, a minimum-phase characteristic implies a minimum delay function, while a maximum-phase characteristic implies that the delay characteristic is also maximum.

Now suppose that we have an FIR system with real coefficients. Then the magnitude square value of its frequency response is

$$|H(\omega)|^2 = H(z)H(z^{-1})|_{z=e^{j\omega}} \tag{5.5.19}$$

This relationship implies that if we replace a zero $z_k$ of the system by its inverse $1/z_k$, the magnitude characteristic of the system does not change. Thus if we reflect a zero $z_k$ that is inside the unit circle into a zero $1/z_k$ outside the unit circle, we see that the magnitude characteristic of the frequency response is invariant to such a change.

It is apparent from this discussion that if $|H(\omega)|^2$ is the magnitude square frequency response of an FIR system having $M$ zeros, there are $2^M$ possible configurations for the $M$ zeros, of which some are inside the unit circle and the remaining are outside the unit circle. Clearly, one configuration has all the zeros inside the unit circle, which corresponds to the minimum-phase system. A second configuration has all the zeros outside the unit circle, which corresponds to the maximum-phase system. The remaining $2^M - 2$ configurations correspond to mixed-phase systems. However, not all $2^M - 2$ mixed-phase configurations necessarily correspond to FIR systems with real-valued coefficients. Specifically, any pair of complex-conjugate zeros results in only two possible configurations, whereas a pair of real-valued zeros yields four possible configurations.

### EXAMPLE 5.5.4

Determine the zeros for the following FIR systems and indicate whether the system is minimum phase, maximum phase, or mixed phase.

$$H_1(z) = 6 + z^{-1} - z^{-2}$$

$$H_2(z) = 1 - z^{-1} - 6z^{-2}$$

$$H_3(z) = 1 - \frac{5}{2}z^{-1} - \frac{3}{2}z^{-2}$$

$$H_4(z) = 1 + \frac{5}{3}z^{-1} - \frac{2}{3}z^{-2}$$

**Solution.** By factoring the system functions we find the zeros for the four systems are

$$H_1(z) \longrightarrow z_{1,2} = -\frac{1}{2}, \frac{1}{3} \longrightarrow \text{minimum phase}$$

$$H_2(z) \longrightarrow z_{1,2} = -2, 3 \longrightarrow \text{maximum phase}$$

$$H_3(z) \longrightarrow z_{1,2} = -\frac{1}{2}, 3 \longrightarrow \text{mixed phase}$$

$$H_4(z) \longrightarrow z_{1,2} = -2, \frac{1}{3} \longrightarrow \text{mixed phase}$$

Since the zeros of the four systems are reciprocals of one another, it follows that all four systems have identical magnitude frequency response characteristics but different phase characteristics.

---

The minimum-phase property of FIR systems carries over to IIR systems that have rational system functions. Specifically, an IIR system with system function

$$H(z) = \frac{B(z)}{A(z)} \tag{5.5.20}$$

is called *minimum phase* if all its poles and zeros are inside the unit circle. For a stable and causal system [all roots of $A(z)$ fall inside the unit circle] the system is called *maximum phase* if all the zeros are outside the unit circle, and *mixed phase* if some, but not all, of the zeros are outside the unit circle.

This discussion brings us to an important point that should be emphasized. That is, a *stable* pole–zero system that is minimum phase has a stable inverse which is also minimum phase. The inverse system has the system function

$$H^{-1}(z) = \frac{A(z)}{B(z)} \tag{5.5.21}$$

Hence the minimum-phase property of $H(z)$ ensures the stability of the inverse system $H^{-1}(z)$ and the stability of $H(z)$ implies the minimum-phase property of $H^{-1}(z)$. Mixed-phase systems and maximum-phase systems result in unstable inverse systems.

**Decomposition of nonminimum-phase pole–zero systems.** Any nonminimum-phase pole–zero system can be expressed as

$$H(z) = H_{\min}(z) H_{\text{ap}}(z) \tag{5.5.22}$$

where $H_{\min}(z)$ is a minimum-phase system and $H_{\text{ap}}(z)$ is an all-pass system. We demonstrate the validity of this assertion for the class of causal and stable systems with a rational system function $H(z) = B(z)/A(z)$. In general, if $B(z)$ has one or more roots outside the unit circle, we factor $B(z)$ into the product $B_1(z)B_2(z)$, where $B_1(z)$ has all its roots inside the unit circle and $B_2(z)$ has all its roots outside the unit circle. Then $B_2(z^{-1})$ has all its roots inside the unit circle. We define the minimum-phase system

$$H_{\min}(z) = \frac{B_1(z)B_2(z^{-1})}{A(z)}$$

and the all-pass system

$$H_{\text{ap}}(z) = \frac{B_2(z)}{B_2(z^{-1})}$$

Thus $H(z) = H_{\min}(z)H_{\text{ap}}(z)$. Note that $H_{\text{ap}}(z)$ is a stable, all-pass, maximum-phase system.

**Group delay of nonminimum-phase system.** Based on the decomposition of a nonminimum-phase system given by (5.5.22), we can express the group delay of $H(z)$ as

$$\tau_g(\omega) = \tau_g^{\min}(\omega) + \tau_g^{\text{ap}}(\omega) \tag{5.5.23}$$

Since $\tau_g^{ap}(\omega) \geq 0$ for $0 \leq \omega \leq \pi$, it follows that $\tau_g(\omega) \geq \tau_g^{\min}(\omega)$, $0 \leq \omega \leq \pi$. From (5.5.23) we conclude that among all pole–zero systems having the same magnitude response, the minimum-phase system has the smallest group delay.

**Partial energy of nonminimum-phase system.** The *partial energy* of a causal system with impulse response $h(n)$ is defined as

$$E(n) = \sum_{k=0}^{n} |h(k)|^2 \tag{5.5.24}$$

It can be shown that among all systems having the same magnitude response and the same total energy $E(\infty)$, the minimum-phase system has the largest partial energy [i.e., $E_{\min}(n) \geq E(n)$, where $E_{\min}(n)$ is the partial energy of the minimum-phase system].

### 5.5.3 System Identification and Deconvolution

Suppose that we excite an unknown linear time-invariant system with an input sequence $x(n)$ and we observe the output sequence $y(n)$. From the output sequence we wish to determine the impulse response of the unknown system. This is a problem in *system identification,* which can be solved by *deconvolution.* Thus we have

$$\begin{aligned} y(n) &= h(n) * x(n) \\ &= \sum_{k=-\infty}^{\infty} h(k)x(n-k) \end{aligned} \tag{5.5.25}$$

An analytical solution of the deconvolution problem can be obtained by working with the $z$-transform of (5.5.25). In the $z$-transform domain we have

$$Y(z) = H(z)X(z)$$

and hence

$$H(z) = \frac{Y(z)}{X(z)} \tag{5.5.26}$$

$X(z)$ and $Y(z)$ are the $z$-transforms of the available input signal $x(n)$ and the observed output signal $y(n)$, respectively. This approach is appropriate only when there are closed-form expressions for $X(z)$ and $Y(z)$.

EXAMPLE 5.5.5

A causal system produces the output sequence

$$y(n) = \begin{cases} 1, & n = 0 \\ \frac{7}{10}, & n = 1 \\ 0, & \text{otherwise} \end{cases}$$

when excited by the input sequence

$$x(n) = \begin{cases} 1, & n = 0 \\ -\frac{7}{10}, & n = 1 \\ \frac{1}{10}, & n = 2 \\ 0, & \text{otherwise} \end{cases}$$

Determine its impulse response and its input–output equation.

**Solution.** The system function is easily determined by taking the $z$-transforms of $x(n)$ and $y(n)$. Thus we have

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 + \frac{7}{10}z^{-1}}{1 - \frac{7}{10}z^{-1} + \frac{1}{10}z^{-2}}$$

$$= \frac{1 + \frac{7}{10}z^{-1}}{(1 - \frac{1}{2}z^{-1})(1 - \frac{1}{5}z^{-1})}$$

Since the system is causal, its ROC is $|z| > \frac{1}{2}$. The system is also stable since its poles lie inside the unit circle.

The input–output difference equation for the system is

$$y(n) = \frac{7}{10}y(n-1) - \frac{1}{10}y(n-2) + x(n) + \frac{7}{10}x(n-1)$$

Its impulse response is determined by performing a partial-fraction expansion of $H(z)$ and inverse transforming the result. This computation yields

$$h(n) = [4(\frac{1}{2})^n - 3(\frac{1}{5})^n]u(n)$$

---

We observe that (5.5.26) determines the unknown system uniquely if it is known that the system is causal. However, the example above is artificial, since the system response $\{y(n)\}$ is very likely to be infinite in duration. Consequently, this approach is usually impractical.

As an alternative, we can deal directly with the time-domain expression given by (5.5.25). If the system is causal, we have

$$y(n) = \sum_{k=0}^{n} h(k)x(n-k), \qquad n \geq 0$$

and hence

$$h(0) = \frac{y(0)}{x(0)}$$

$$h(n) = \frac{y(n) - \sum_{k=0}^{n-1} h(k)x(n-k)}{x(0)}, \qquad n \geq 1 \tag{5.5.27}$$

This recursive solution requires that $x(0) \neq 0$. However, we note again that when $\{h(n)\}$ has infinite duration, this approach may not be practical unless we truncate the recursive solution at same stage [i.e., truncate $\{h(n)\}$].

Another method for identifying an unknown system is based on a crosscorrelation technique. Recall that the input–output crosscorrelation function derived in Section 2.6.4 is given as

$$r_{yx}(m) = \sum_{k=0}^{\infty} h(k) r_{xx}(m-k) = h(n) * r_{xx}(m) \tag{5.5.28}$$

where $r_{yx}(m)$ is the crosscorrelation sequence of the input $\{x(n)\}$ to the system with the output $\{y(n)\}$ of the system, and $r_{xx}(m)$ is the autocorrelation sequence of the input signal. In the frequency domain, the corresponding relationship is

$$S_{yx}(\omega) = H(\omega) Sxx(\omega) = H(\omega)|X(\omega)|^2$$

Hence

$$H(\omega) = \frac{S_{yx}(\omega)}{S_{xx}(\omega)} = \frac{S_{yx}(\omega)}{|X(\omega)|^2} \tag{5.5.29}$$

These relations suggest that the impulse response $\{h(n)\}$ or the frequency response of an unknown system can be determined (measured) by crosscorrelating the input sequence $\{x(n)\}$ with the output sequence $\{y(n)\}$, and then solving the deconvolution problem in (5.5.28) by means of the recursive equation in (5.5.27). Alternatively, we could simply compute the Fourier transform of (5.5.28) and determine the frequency response given by (5.5.29). Furthermore, if we select the input sequence $\{x(n)\}$ such that its autocorrelation sequence $\{r_{xx}(n)\}$, is a unit sample sequence, or equivalently, that its spectrum is flat (constant) over the passband of $H(\omega)$, the values of the impulse response $\{h(n)\}$ are simply equal to the values of the crosscorrelation sequence $\{r_{yx}(n)\}$.

In general, the crosscorrelation method described above is an effective and practical method for system identification. Another practical approach based on least-squares optimization is described in Chapter 13.

### 5.5.4 Homomorphic Deconvolution

The complex cepstrum, introduced in Section 4.2.7, is a useful tool for performing deconvolution in some applications such as seismic signal processing. To describe this

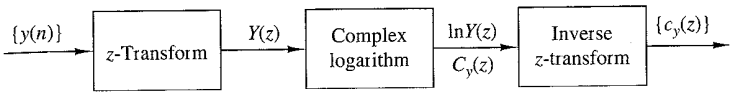



**Figure 5.5.4** Homomorphic system for obtaining the cepstrum $\{c_y(n)\}$ of the sequence $\{y(n)\}$.

method, let us suppose that $\{y(n)\}$ is the output sequence of a linear time-invariant system which is excited by the input sequence $\{x(n)\}$. Then

$$Y(z) = X(z)H(z) \tag{5.5.30}$$

where $H(z)$ is the system function. The logarithm of $Y(z)$ is

$$\begin{aligned} C_y(z) &= \ln Y(z) \\ &= \ln X(z) + \ln H(z) \\ &= C_x(z) + C_h(z) \end{aligned} \tag{5.5.31}$$

Consequently, the complex cepstrum of the output sequence $\{y(n)\}$ is expressed as the sum of the cepstrum of $\{x(n)\}$ and $\{h(n)\}$, that is,

$$c_y(n) = c_x(n) + c_h(n) \tag{5.5.32}$$

Thus we observe that convolution of the two sequences in the time domain corresponds to the summation of the cepstrum sequences in the cepstral domain. The system for performing these transformations is called a *homormorphic system*and is illustrated in Fig. 5.5.4.

In some applications, such as seismic signal processing and speech signal processing, the characteristics of the cepstral sequences $\{c_x(n)\}$ and $\{c_h(n)\}$ are sufficiently different so that they can be separated in the cepstral domain. Specifically, suppose that $\{c_h(n)\}$ has its main components (main energy) in the vicinity of small values of $n$, whereas $\{c_x(n)\}$ has its components concentrated at large values of $n$. We may say that $\{c_h(n)\}$ is "lowpass" and $\{c_x(n)\}$ is "highpass." We can then separate $\{c_h(n)\}$ from $\{c_x(n)\}$ using appropriate "lowpass" and "highpass" windows, as illustrated in Fig. 5.5.5. Thus

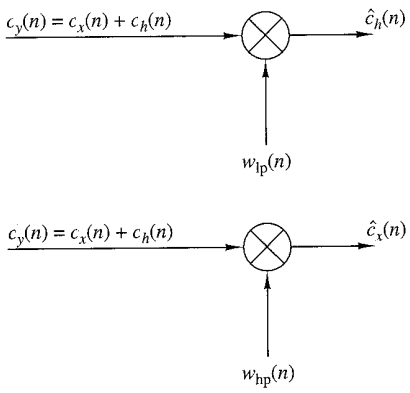



**Figure 5.5.5**
Separating the two cepstral components by "lowpass" and "highpass" windows.

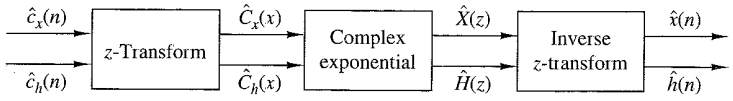



**Figure 5.5.6** Inverse homomorphic system for recovering the sequences $\{x(n)\}$ and $\{h(n)\}$ from the corresponding cepstra.

$$\hat{c}_h(n) = c_y(n)w_{\text{lp}}(n) \tag{5.5.33}$$

and

$$\hat{c}_x(n) = c_y(n)w_{\text{hp}}(n) \tag{5.5.34}$$

where

$$w_{\text{lp}}(n) = \begin{cases} 1, & |n| \le N_1 \\ 0, & \text{otherwise} \end{cases} \tag{5.5.35}$$

$$w_{\text{hp}}(n) = \begin{cases} 0, & |n| \le N_1 \\ 1, & |n| > N_1 \end{cases} \tag{5.5.36}$$

Once we have separated the cepstrum sequences $\{\hat{c}_h(n)\}$ and $\{\hat{c}_x(n)\}$ by windowing, the sequences $\{\hat{x}(n)\}$ and $\{\hat{h}(n)\}$ are obtained by passing $\{\hat{c}_h(n)\}$ and $\{\hat{c}_x(n)\}$ through the inverse homomorphic system, shown in Fig. 5.5.6.

In practice, a digital computer would be used to compute the cepstrum of the sequence $\{y(n)\}$, to perform the windowing functions, and to implement the inverse homomorphic system shown in Fig. 5.5.6. In place of the $z$-transform and inverse $z$-transform, we would substitute a special form of the Fourier transform and its inverse. This special form, called the discrete Fourier transform, is described in Chapter 7.

## 5.6 Summary and References

In this chapter we considered the frequency-domain characteristics of LTI systems. We showed that an LTI system is characterized in the frequency domain by its frequency response function $H(\omega)$, which is the Fourier transform of the impulse response of the system. We also observed that the frequency response function determines the effect of the system on any input signal. In fact, by transforming the input signal into the frequency domain, we observed that it is a simple matter to determine the effect of the system on the signal and to determine the system output. When viewed in the frequency domain, an LTI system performs spectral shaping or spectral filtering on the input signal.

The design of some simple IIR filters was also considered in this chapter from the viewpoint of pole–zero placement. By means of this method, we were able to design simple digital resonators, notch filters, comb filters, all-pass filters, and digital sinusoidal generators. The design of more complex IIR filters is treated in detail in Chapter 10, which also includes several references. Digital sinusoidal generators find use in frequency synthesis applications. A comprehensive treatment of frequency synthesis techniques is given in the text edited by Gorski-Popiel (1975).

Finally, we characterized LTI systems as either minimum-phase, maximum-phase, or mixed-phase, depending on the position of their poles and zeros in the frequency domain. Using these basic characteristics of LTI systems, we considered practical problems in inverse filtering, deconvolution, and system identification. We concluded with the description of a deconvolution method based on cepstral analysis of the output signal from a linear system.

A vast amount of technical literature exists on the topics of inverse filtering, deconvolution, and system identification. In the context of communications, system identification and inverse filtering as they relate to channel equalization are treated in the book by Proakis (2001). Deconvolution techniques are widely used in seismic signal processing. For reference, we suggest the papers by Wood and Treitel (1975), Peacock and Treitel (1969), and the books by Robinson and Treitel (1978, 1980). Homomorphic deconvolution and its applications to speech processing aretreated in the book by Oppenheim and Schafer (1989).

## Problems

**5.1** The following input–output pairs have been observed during the operation of various systems:

**(a)** $x(n) = (\frac{1}{2})^n \xrightarrow{\mathcal{T}_1} y(n) = (\frac{1}{8})^n$

**(b)** $x(n) = (\frac{1}{2})^n u(n) \xrightarrow{\mathcal{T}_2} y(n) = (\frac{1}{8})^n u(n)$

**(c)** $x(n) = e^{j\pi/5} \xrightarrow{\mathcal{T}_3} y(n) = 3e^{j\pi/5}$

**(d)** $x(n) = e^{j\pi/5} u(n) \xrightarrow{\mathcal{T}_4} y(n) = 3e^{j\pi/5 u(n)}$

**(e)** $x(n) = x(n + N_1) \xrightarrow{\mathcal{T}_5} y(n) = y(n + N_2), \qquad N_1 \neq N_2, \quad N_1, N_2$ prime

Determine their frequency response if each of the above systems is LTI.

**5.2** **(a)** Determine and sketch the Fourier transform $W_R(\omega)$ of the rectangular sequence

$$w_R(n) = \begin{cases} 1, & 0 \leq n \leq M \\ 0, & \text{otherwise} \end{cases}$$

**(b)** Consider the triangular sequence

$$w_I(n) = \begin{cases} n, & 0 \leq n \leq M/2 \\ M - n, & M/2 < 2 \leq M \\ 0, & \text{otherwise} \end{cases}$$

Determine and sketch the Fourier transform $W_I(\omega)$ of $w_I(n)$ by expressing it as the convolution of a rectangular sequence with itself.

**(c)** Consider the sequence

$$w_c(n) = \frac{1}{2}\left(1 + \cos\frac{2\pi n}{M}\right) w_R(n)$$

Determine and sketch $W_c(\omega)$ by using $W_R(\omega)$.

**5.3** Consider an LTI system with impulse response $h(n) = (\frac{1}{2})^n u(n)$.

**(a)** Determine and sketch the magnitude and phase response $|H(\omega)|$ and $\angle H(\omega)$, respectively.

**(b)** Determine and sketch the magnitude and phase spectra for the input and output signals for the following inputs:

**1.** $x(n) = \cos \dfrac{3\pi n}{10}, \quad -\infty < n < \infty$

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

**5.4** Determine and sketch the magnitude and phase response of the following systems:

**(a)** $y(n) = \frac{1}{2}[x(n) + x(n-1)]$

**(b)** $y(n) = \frac{1}{2}[x(n) - x(n-1)]$

**(c)** $y(n) = \frac{1}{2}[x(n+1) - x(n-1)]$

**(d)** $y(n) = \frac{1}{2}[x(n+1) + x(n-1)]$

**(e)** $y(n) = \frac{1}{2}[x(n) + x(n-2)]$

**(f)** $y(n) = \frac{1}{2}[x(n) - x(n-2)]$

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

**(h)** $y(n) = x(n) - x(n-8)$

**(i)** $y(n) = 2x(n-1) - x(n-2)$

**(j)** $y(n) = \frac{1}{4}[x(n) + x(n-1) + x(n-2) + x(n-3)]$

**(k)** $y(n) = \frac{1}{8}[x(n) + 3x(n-1) + 3x(n-2) + x(n-3)]$

**(l)** $y(n) = x(n-4)$

**(m)** $y(n) = x(n+4)$

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

**5.5** An FIR filter is described by the difference equation

$$y(n) = x(n) + x(n-10)$$

**(a)** Compute and sketch its magnitude and phase response.

**(b)** Determine its response to the inputs

**1.** $x(n) = \cos \dfrac{\pi}{10} n + 3 \sin\left(\dfrac{\pi}{3} n + \dfrac{\pi}{10}\right), \qquad -\infty < n < \infty$

**2.** $x(n) = 10 + 5 \cos\left(\dfrac{2\pi}{5} n + \dfrac{\pi}{2}\right), \qquad -\infty < n < \infty$

**5.6** Determine the transient and steady-state responses of the FIR *filter* shown in Fig. P5.6 to the input signal $x(n) = 10e^{j\pi n/2}u(n)$. Let $b = 2$ and $y(-1) = y(-2) = y(-3) = y(-4) = 0$.

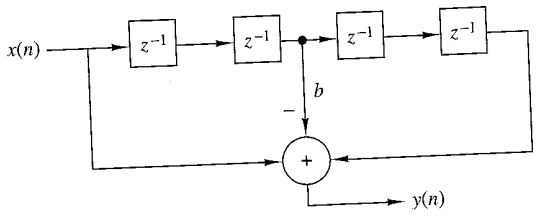



**Figure P5.6**

**5.7** Consider the FIR filter

$$y(n) = x(n) + x(n-4)$$

**(a)** Compute and sketch its magnitude and phase response.

**(b)** Compute its response to the input

$$x(n) = \cos\frac{\pi}{2}n + \cos\frac{\pi}{4}n, \qquad -\infty < n < \infty$$

**(c)** Explain the results obtained in part (b) in terms of the magnitude and phase responses obtained in part (a).

**5.8** Determine the steady-state and transient responses of the system

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

to the input signal

$$x(n) = 5 + 3\cos\left(\frac{\pi}{2}n + 60^\circ\right), \qquad -\infty < n < \infty$$

**5.9** From our discussions it is apparent that an LTI system cannot produce frequencies at its output that are different from those applied in its input. Thus, if a system creates "new" frequencies, it must be nonlinear and/or time varying. Determine the frequency content of the outputs of the following systems to the input signal

$$x(n) = A\cos\frac{\pi}{4}n$$

**(a)** $y(n) = x(2n)$

**(b)** $y(n) = x^2(n)$

**(c)** $y(n) = (\cos\pi n)x(n)$

**5.10** Determine and sketch the magnitude and phase response of the systems shown in Fig. P5.10(a) through (c).

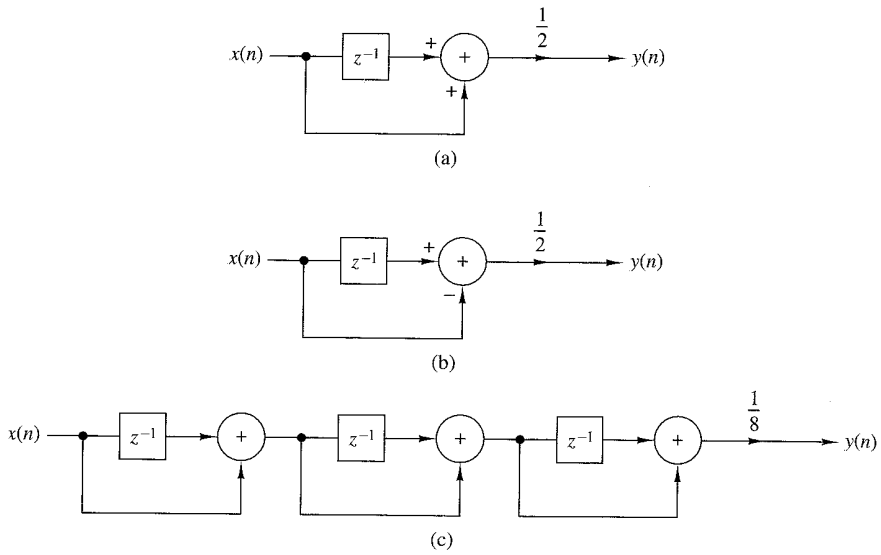



**Figure P5.10**

**5.11** Determine the magnitude and phase response of the multipath channel

$$y(n) = x(n) + x(n - M)$$

At what frequencies does $H(\omega) = 0$?

**5.12** Consider the filter

$$y(n) = 0.9y(n - 1) + bx(n)$$

**(a)** Determine $b$ so that $|H(0)| = 1$.

**(b)** Determine the frequency at which $|H(\omega)| = 1/\sqrt{2}$.

**(c)** Is this filter lowpass, bandpass, or highpass?

**(d)** Repeat parts (b) and (c) for the filter $y(n) = -0.9y(n - 1) + 0.1x(n)$.

**5.13** *Harmonic distortion in digital sinusoidal generators* An ideal sinusoidal generator produces the signal

$$x(n) = \cos 2\pi f_0 n, \qquad -\infty < n < \infty$$

which is periodic with fundamental period $N$ if $f_0 = k_0/N$ and $k_0$, $N$ are relatively prime numbers. The spectrum of such a "pure" sinusoid consist of two lines at $k = k_0$ and $k = N - k_0$ (we limit ourselves in the fundamental interval $0 \le k \le N - 1$). In practice, the approximations made in computing the samples of a sinusoid of relative frequency $f_0$ result in a certain amount of power falling into other frequencies. This spurious power results in distortion, which is referred to as *harmonic distortion*.

Harmonic distortion is usually measured in terms of the *total harmonic distortion* (THD), which is defined as the ratio

$$\text{THD} = \frac{\text{spurious harmonic power}}{\text{total power}}$$

**(a)** Show that

$$\text{THD} = 1 - 2\frac{|c_{k_0}|^2}{P_x}$$

where

$$c_{k_0} = \frac{1}{N}\sum_{n=0}^{N-1} x(n)e^{-j(2\pi/N)k_0 n}$$

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

**(b)** By using the Taylor approximation

$$\cos\phi = 1 - \frac{\phi^2}{2!} + \frac{\phi^4}{4!} - \frac{\phi^6}{6!} + \cdots$$

compute one period of $x(n)$ for $f_0 = 1/96, 1/32, 1/256$ by increasing the number of terms in the Taylor expansion from 2 to 8.

**(c)** Compute the THD and plot the power density spectrum for each sinusoid in part (b) as well as for the sinusoids obtained using the computer cosine function. Comment on the results.

**5.14** *Measurement of the total harmonic distortion in quantized sinusoids* Let $x(n)$ be a periodic sinusoidal signal with frequency $f_0 = k/N$, that is,

$$x(n) = \sin 2\pi f_0 n$$

**(a)** Write a computer program that quantizes the signal $x(n)$ into $b$ bits or equivalently into $L = 2^b$ levels by using rounding. The resulting signal is denoted by $x_q(n)$.

**(b)** For $f_0 = 1/50$ compute the THD of the quantized signals $x_q(n)$ obtained by using $b = 4, 6, 8$, and 16 bits.

**(c)** Repeat part (b) for $f_0 = 1/100$.

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

**5.15** Consider the discrete-time system

$$y(n) = ay(n-1) + (1-a)x(n), \qquad n \geq 0$$

where $a = 0.9$ and $y(-1) = 0$.

**(a)** Compute and sketch the output $y_i(n)$ of the system to the input signals

$$x_i(n) = \sin 2\pi f_i n, \qquad 0 \le n \le 100$$

where $f_1 = \frac{1}{4}$, $f_2 = \frac{1}{5}$, $f_3 = \frac{1}{10}$, $f_4 = \frac{1}{20}$.

**(b)** Compute and sketch the magnitude and phase response of the system and use these results to explain the response of the system to the signals given in part (a).

**5.16** Consider an LTI system with impulse response $h(n) = (\frac{1}{3})^{|n|}$.

**(a)** Determine and sketch the magnitude and phase response $H(\omega)|$ and $\measuredangle H(\omega)$, respectively.

**(b)** Determine and sketch the magnitude and phase spectra for the input and output signals for the following inputs:

1. $x(n) = \cos \dfrac{3\pi n}{8}, \; -\infty < n < \infty$
2. $x(n) = \{\ldots, -1, 1, -1, \underset{\uparrow}{1}, -1, 1, -1, 1, -1, 1, -1, 1, \ldots\}$

**5.17** Consider the digital filter shown in Fig. P5.17.

**(a)** Determine the input–output relation and the impulse response $h(n)$.

**(b)** Determine and sketch the magnitude $|H(\omega)|$ and the phase response $\measuredangle H(\omega)$ of the filter and find which frequencies are completely blocked by the filter.

**(c)** When $\omega_0 = \pi/2$, determine the output $y(n)$ to the input

$$x(n) = 3\cos\left(\frac{\pi}{3}n + 30^\circ\right), \qquad -\infty < n < \infty$$

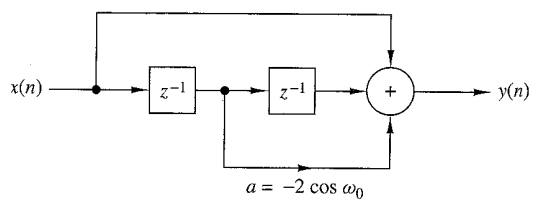



**Figure P5.17**

**5.18** Consider the FIR filter

$$y(n) = x(n) - x(n-4)$$

**(a)** Compute and sketch its magnitude and phase response.

**(b)** Compute its response to the input

$$x(n) = \cos\frac{\pi}{2}n + \cos\frac{\pi}{4}n, \qquad -\infty < n < \infty$$

**(c)** Explain the results obtained in part (b) in terms of the answer given in part (a).

**5.19** Determine the steady-state response of the system

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

to the input signal

$$x(n) = 5 + 3\cos\left(\frac{\pi}{2}n + 60^\circ\right) + 4\sin(\pi n + 45^\circ), \qquad -\infty < n < \infty$$

**5.20** Recall from Problem 5.9 that an LTI system cannot produce frequencies at its output that are different from those applied in its input. Thus if a system creates "new" frequencies, it must be nonlinear and/or time varying. Indicate whether the following systems are nonlinear and/or time varying and determine the output spectra when the input spectrum is

$$X(\omega) = \begin{cases} 1, & |\omega| \le \pi/4 \\ 0, & \pi/4 \le |\omega| \le \pi \end{cases}$$

**(a)** $y(n) = x(2n)$

**(b)** $y(n) = x^2(n)$

**(c)** $y(n) = (\cos \pi n)x(n)$

**5.21** Consider an LTI system with impulse response

$$h(n) = \left[\left(\frac{1}{4}\right)^n \cos\left(\frac{\pi}{4}n\right)\right]u(n)$$

**(a)** Determine its system function $H(z)$.

**(b)** Is it possible to implement this system using a finite number of adders, multipliers, and unit delays? If yes, how?

**(c)** Provide a rough sketch of $|H(\omega)|$ using the pole–zero plot.

**(d)** Determine the response of the system to the input

$$x(n) = (\frac{1}{4})^n u(n)$$

**5.22** An FIR filter is described by the difference equation

$$y(n) = x(n) - x(n-6)$$

**(a)** Compute and sketch its magnitude and phase response.

**(b)** Determine its response to the inputs

1. $x(n) = \cos\frac{\pi}{10}n + 3\sin\left(\frac{\pi}{3}n + \frac{\pi}{10}\right), \qquad -\infty < n < \infty$
2. $x(n) = 5 + 6\cos\left(\frac{2\pi}{5}n + \frac{\pi}{2}\right), \qquad -\infty < n < \infty$

**5.23** The frequency response of an ideal bandpass filter is given by

$$H(\omega) = \begin{cases} 0, & |\omega| \le \dfrac{\pi}{8} \\ 1, & \dfrac{\pi}{8} < |\omega| < \dfrac{3\pi}{8} \\ 0, & \dfrac{3\pi}{8} \le |\omega| \le \pi \end{cases}$$

**(a)** Determine its impulse response

**(b)** Show that this impulse response can be expressed as the product of $\cos(n\pi/4)$ and the impulse response of a lowpass filter.

**5.24** Consider the system described by the difference equation

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

**(a)** Determine its impulse response.

**(b)** Determine its frequency response:

1. From the impulse response
2. From the difference equation

**(c)** Determine its response to the input

$$x(n) = \cos\left(\frac{\pi}{2}n + \frac{\pi}{4}\right), \qquad -\infty < n < \infty$$

**5.25** Sketch roughly the magnitude $|H(\omega)|$ of the Fourier transforms corresponding to the pole–zero patterns of systems given in Fig. P5.25.

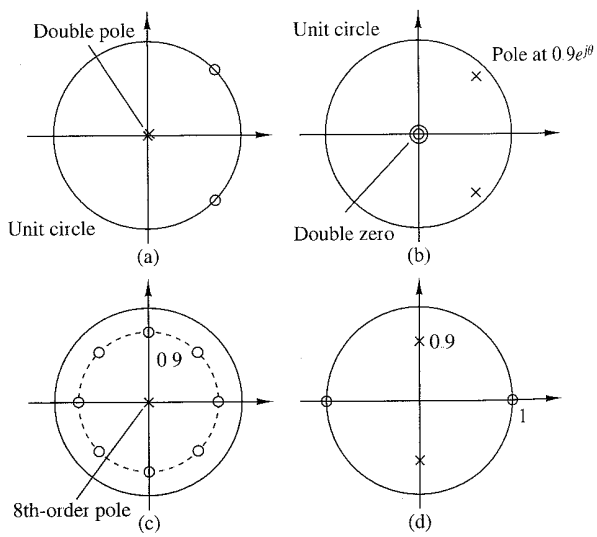



**Figure P5.25**

**5.26** Design an FIR filter that completely blocks the frequency $\omega_0 = \pi/4$ and then compute its output if the input is

$$x(n) = \left(\sin \frac{\pi}{4} n\right) u(n)$$

for $n = 0, 1, 2, 3, 4$. Does the filter fulfill your expectations? Explain.

**5.27** A digital filter is characterized by the following properties:

1. It is highpass and has one pole and one zero.
2. The pole is at a distance $r = 0.9$ from the origin of the $z$-plane.
3. Constant signals do not pass through the system.

**(a)** Plot the pole–zero pattern of the filter and determine its system function $H(z)$.

**(b)** Compute the magnitude response $|H(\omega)|$ and the phase response $\measuredangle H(\omega)$ of the filter.

**(c)** Normalize the frequency response $H(\omega)$ so that $|H(\pi)| = 1$.

**(d)** Determine the input–output relation (difference equation) of the filter in the time domain.

**(e)** Compute the output of the system if the input is

$$x(n) = 2\cos\left(\frac{\pi}{6} n + 45^\circ\right), \qquad -\infty < n < \infty$$

(You can use either algebraic or geometrical arguments.)

**5.28** A causal first-order digital filter is described by the system function

$$H(z) = b_0 \frac{1 + bz^{-1}}{1 + az^{-1}}$$

**(a)** Sketch the direct form I and direct form II realizations of this filter and find the corresponding difference equations.

**(b)** For $a = 0.5$ and $b = -0.6$, sketch the pole–zero pattern. Is the system stable? Why?

**(c)** For $a = -0.5$ and $b = 0.5$, determine $b_0$, so that the maximum value of $|H(\omega)|$ is equal to 1.

**(d)** Sketch the magnitude response $|H(\omega)|$ and the phase response $\measuredangle H(\omega)$ of the filter obtained in part (c).

**(e)** In a specific application it is known that $a = 0.8$. Does the resulting filter amplify high frequencies or low frequencies in the input? Choose the value of $b$ so as to improve the characteristics of this filter (i.e., make it a better lowpass or a better highpass filter).

**5.29** Derive the expression for the resonant frequency of a two-pole filter with poles at $p_1 = re^{j\theta}$ and $p_2 = p_1^*$, given by (5.4.25).

**5.30** Determine and sketch the magnitude and phase responses of the Hanning filter characterized by the (moving average) difference equation

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

**5.31** A causal LTI system excited by the input

$$x(n) = (\frac{1}{4})^n u(n) + u(-n-1)$$

produces an output $y(n)$ with $z$-transform

$$Y(z) = \frac{-\frac{3}{4}z^{-1}}{(1-\frac{1}{4}z^{-1})(1+z^{-1})}$$

**(a)** Determine the system function $H(z)$ and its ROC.

**(b)** Determine the output $y(n)$ of the system.
(*Hint:* Pole cancellation increases the original ROC.)

**5.32** Determine the coefficients of a linear-phase FIR filter

$$y(n) = b_0x(n) + b_1x(n-1) + b_2x(n-2)$$

such that:

**(a)** It rejects completely a frequency component at $\omega_0 = 2\pi/3$.

**(b)** Its frequency response is normalized so that $H(0) = 1$.

**(c)** Compute and sketch the magnitude and phase response of the filter to check if it satisfies the requirements.

**5.33** Determine the frequency response $H(\omega)$ of the following moving average filters.

**(a)** $y(n) = \dfrac{1}{2M+1}\displaystyle\sum_{k=-M}^{M} x(n-k)$

**(b)** $y(n) = \dfrac{1}{4M}x(n+M) + \dfrac{1}{2M}\displaystyle\sum_{k=-M+1}^{M-1} x(n-k) + \dfrac{1}{4M}x(n-M)$

Which filter provides better smoothing? Why?

**5.34** Compute the magnitude and phase response of a filter with system function

$$H(z) = 1 + z^{-1} + z^{-2} + \cdots + z^{-8}$$

If the sampling frequency is $F_s = 1$ kHz, determine the frequencies of the analog sinusoids that cannot pass through the filter.

**5.35** A second-order system has a double pole at $p_{1,2} = 0.5$ and two zeros at

$$z_{1,2} = e^{\pm j3\pi/4}$$

Using geometric arguments, choose the gain $G$ of the filter so that $|H(0)| = 1$.

**5.36** In this problem we consider the effect of a single zero on the frequency response of a system. Let $z = re^{j\theta}$ be a zero inside the unit circle ($r < 1$). Then

$$\begin{aligned} H_z(\omega) &= 1 - re^{j\theta}e^{-j\omega} \\ &= 1 - r\cos(\omega - \theta) + jr\sin(\omega - \theta) \end{aligned}$$

**(a)** Show that the magnitude response is

$$|H_z(\omega)| = [1 - 2r\cos(\omega - \theta) + r^2]^{1/2}$$

or, equivalently,

$$20\log_{10}|H_z(\omega)| = 10\log_{10}[1 - 2r\cos(\omega - \theta) + r^2]$$

**(b)** Show that the phase response is given as

$$\Theta_z(\omega) = \tan^{-1}\frac{r\sin(\omega - \theta)}{1 - r\cos(\omega - \theta)}$$

**(c)** Show that the group delay is given as

$$\tau_g(\omega) = \frac{r^2 - r\cos(\omega - \theta)}{1 + r^2 - 2r\cos(\omega - \theta)}$$

**(d)** Plot the magnitude $|H(\omega)|_{\text{dB}}$, the phase $\Theta(\omega)$ and the group delay $\tau_g(\omega)$ for $r = 0.7$ and $\theta = 0, \pi/2$, and $\pi$.

**5.37** In this problem we consider the effect of a single pole on the frequency response of a system. Hence, we let

$$H_p(\omega) = \frac{1}{1 - re^{j\theta}e^{-j\omega}}, \qquad r < 1$$

Show that

$$\begin{aligned} |H_p(\omega)|_{\text{dB}} &= -|H_z(\omega)|_{\text{dB}} \\ \measuredangle H_p(\omega) &= -\measuredangle H_z(\omega) \\ \tau_g^p(\omega) &= -\tau_g^z(\omega) \end{aligned}$$

where $H_z(\omega)$ and $\tau_g^z(\omega)$ are defined in Problem 5.36.

**5.38** In this problem we consider the effect of complex-conjugate pairs of poles and zeros on the frequency response of a system. Let

$$H_z(\omega) = (1 - re^{j\theta}e^{-j\omega})(1 - re^{-j\theta}e^{-j\omega})$$

**(a)** Show that the magnitude response in decibels is

$$|H_z(\omega)|_{\text{dB}} = 10\log_{10}[1 + r^2 - 2r\cos(\omega - \theta)] + 10\log_{10}[1 + r^2 - 2r\cos(\omega + \theta)]$$

**(b)** Show that the phase response is given as

$$\Theta_z(\omega) = \tan^{-1}\frac{r\sin(\omega - \theta)}{1 - r\cos(\omega - \theta)} + \tan^{-1}\frac{r\sin(\omega + \theta)}{1 - r\cos(\omega + \theta)}$$

**(c)** Show that the group delay is given as

$$\tau_g^z(\omega) = \frac{r^2 - r\cos(\omega - \theta)}{1 + r^2 - 2r\cos(\omega - \theta)} + \frac{r^2 - r\cos(\omega + \theta)}{1 + r^2 - 2r\cos(\omega + \theta)}$$

**(d)** If $H_p(\omega) = 1/H_z(\omega)$, show that

$$|H_p(\omega)|_{\text{dB}} = -|H_z(\omega)|_{\text{dB}}$$

$$\Theta_p(\omega) = -\Theta_z(\omega)$$

$$\tau_g^p(\omega) = -\tau_g^z(\omega)$$

**(e)** Plot $|H_p(\omega)|$, $\Theta_p(\omega)$ and $\tau_g^p(\omega)$ for $\tau = 0.9$, and $\theta = 0, \pi/2$.

**5.39** Determine the 3-dB bandwidth of the filters $(0 < a < 1)$

$$H_1(z) = \frac{1 - a}{1 - az^{-1}}$$

$$H_2(z) = \frac{1 - a}{2}\frac{1 + z^{-1}}{1 - az^{-1}}$$

Which is a better lowpass filter?

**5.40** Design a digital oscillator with adjustable phase, that is, a digital filter which produces the signal

$$y(n) = \cos(\omega_0 n + \theta)u(n)$$

**5.41** This problem provides another derivation of the structure for the coupled-form oscillator by considering the system

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

for $a = e^{j\omega_0}$.

Let $x(n)$ be real. Then $y(n)$ is complex. Thus

$$y(n) = y_R(n) + jy_I(n)$$

**(a)** Determine the equations describing a system with one input $x(n)$ and the two outputs $y_R(n)$ and $y_I(n)$.

**(b)** Determine a block diagram realization

**(c)** Show that if $x(n) = \delta(n)$, then

$$y_R(n) = (\cos \omega_0 n)u(n)$$

$$y_I(n) = (\sin \omega_0 n)u(n)$$

**(d)** Compute $y_R(n)$, $y_I(n)$, $n = 0, 1, \ldots, 9$ for $\omega_0 = \pi/6$. Compare these with the true values of the sine and cosine.

**5.42** Consider a filter with system function

$$H(z) = b_0 \frac{(1 - e^{j\omega_0} z^{-1})(1 - e^{-j\omega_0} z^{-1})}{(1 - re^{j\omega_0} z^{-1})(1 - re^{-j\omega_0} z^{-1})}$$

**(a)** Sketch the pole–zero pattern.

**(b)** Using geometric arguments, show that for $r \simeq 1$, the system is a notch filter and provide a rough sketch of its magnitude response if $\omega_0 = 60°$.

**(c)** For $\omega_0 = 60°$, choose $b_0$ so that the maximum value of $|H(\omega)|$ is 1.

**(d)** Draw a direct form II realization of the system.

**(e)** Determine the approximate 3-dB bandwidth of the system.

**5.43** Design an FIR digital filter that will reject a very strong 60-Hz sinusoidal interference contaminating a 200-Hz useful sinusoidal signal. Determine the gain of the filter so that the useful signal does not change amplitude. The filter works at a sampling frequency $F_s = 500$ samples/s. Compute the output of the filter if the input is a 60-Hz sinusoid or a 200-Hz sinusoid with unit amplitude. How does the performance of the filter compare with your requirements?

**5.44** Determine the gain $b_0$ for the digital resonator described by (5.4.28) so that $|H(\omega_0)| = 1$.

**5.45** Demonstrate that the difference equation given in (5.4.52) can be obtained by applying the trigonometric identity

$$\cos \alpha + \cos \beta = 2 \cos \frac{\alpha + \beta}{2} \cos \frac{\alpha - \beta}{2}$$

where $\alpha = (n + 1)\omega_0$, $\beta = (n - 1)\omega_0$, and $y(n) = \cos \omega_0 n$. Thus show that the sinusoidal signal $y(n) = A \cos \omega_0 n$ can be generated from (5.4.52) by use of the initial conditions $y(-1) = A \cos \omega_0$ and $y(-2) = A \cos 2\omega_0$.

**5.46** Use the trigonometric identity in (5.4.53) with $\alpha = n\omega_0$ and $\beta = (n - 2)\omega_0$ to derive the difference equation for generating the sinusoidal signal $y(n) = A \sin n\omega_0$. Determine the corresponding initial conditions.

**5.47** Using the $z$-transform pairs 8 and 9 in Table 3.3, determine the difference equations for the digital oscillators that have impulse responses $h(n) = A \cos n\omega_0 u(n)$ and $h(n) = A \sin n\omega_0 u(n)$, respectively.

**5.48** Determine the structure for the coupled-form oscillator by combining the structure for the digital oscillators obtained in Problem 5.47.

**5.49** Convert the highpass filter with system function

$$H(z) = \frac{1 - z^{-1}}{1 - az^{-1}}, \qquad a < 1$$

into a notch filter that rejects the frequency $\omega_0 = \pi/4$ and its harmonics.

**(a)** Determine the difference equation.

**(b)** Sketch the pole–zero pattern.

**(c)** Sketch the magnitude response for both filters.

**5.50** Choose $L$ and $M$ for a lunar filter that must have narrow passbands at $(k \pm \Delta F)$ cycles/day, where $k = 1, 2, 3, \ldots$ and $\Delta F = 0.067726$.

**5.51** **(a)** Show that the systems corresponding to the pole–zero patterns of Fig. 5.4.16 are all-pass.

**(b)** What is the number of delays and multipliers required for the efficient implementation of a second-order all-pass system?

**5.52** A digital notch filter is required to remove an undesirable 60-Hz hum associated with a power supply in an ECG recording application. The sampling frequency used is $F_s = 500$ samples/s. **(a)** Design a second-order FIR notch filter and **(b)** a second-order pole–zero notch filter for this purpose. In both cases choose the gain $b_0$ so that $|H(\omega)| = 1$ for $\omega = 0$.

**5.53** Determine the coefficients $\{h(n)\}$ of a highpass linear phase FIR filter of length $M = 4$ which has an antisymmetric unit sample response $h(n) = -h(M - 1 - n)$ and a frequency response that satisfies the condition

$$\left|H\left(\frac{\pi}{4}\right)\right| = \frac{1}{2}, \qquad \left|H\left(\frac{3\pi}{4}\right)\right| = 1$$

**5.54** In an attempt to design a four-pole bandpass digital filter with desired magnitude response

$$|H_d(\omega)| = \begin{cases} 1, & \dfrac{\pi}{6} \le \omega \le \dfrac{\pi}{2} \\ 0, & \text{elsewhere} \end{cases}$$

we select the four poles at

$$p_{1,2} = 0.8e^{\pm j^2\pi/9}$$

$$p_{3,4} = 0.8e^{\pm j^4\pi/9}$$

and four zeros at

$$z_1 = 1, \qquad z_2 = -1, \qquad z_{3,4} = e^{\pm 3\pi/4}$$

**(a)** Determine the value of the gain so that

$$\left|H\left(\frac{5\pi}{12}\right)\right| = 1$$

**(b)** Determine the system function $H(z)$.

**(c)** Determine the magnitude of the frequency response $H(\omega)$ for $0 \le \omega \le \pi$ and compare it with the desired response $|H_d(\omega)|$.

**5.55** A discrete-time system with input $x(n)$ and output $y(n)$ is described in the frequency domain by the relation

$$Y(\omega) = e^{-j2\pi\omega}X(\omega) + \frac{dX(\omega)}{d\omega}$$

**(a)** Compute the response of the system to the input $x(n) = \delta(n)$.

**(b)** Check if the system is LTI and stable.

**5.56** Consider an ideal lowpass filter with impulse response $h(n)$ and frequency response

$$H(\omega) = \begin{cases} 1, & |\omega| \le \omega_c \\ 0, & \omega_c < |\omega| < \pi \end{cases}$$

What is the frequency response of the filter defined by

$$g(n) = \begin{cases} h\left(\frac{n}{2}\right), & n \text{ even} \\ 0, & n \text{ odd} \end{cases}$$

**5.57** Consider the system shown in Fig. P5.57. Determine its impulse response and its frequency response if the system $H(\omega)$ is:

**(a)** Lowpass with cutoff frequency $\omega_c$.

**(b)** Highpass with cutoff frequency $\omega_c$.

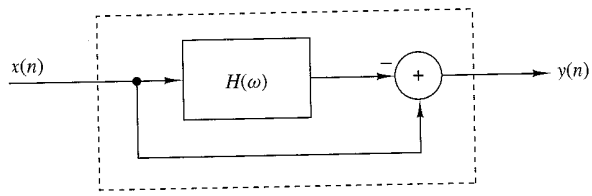



**Figure P5.57**

**5.58** Frequency inverters have been used for many years for speech scrambling. Indeed, a voice signal $x(n)$ becomes unintelligible if we invert its spectrum as shown in Fig. P5.58.

**(a)** Determine how frequency inversion can be performed in the time domain.

**(b)** Design an unscrambler. (*Hint:* The required operations are very simple and can easily be done in real time.)

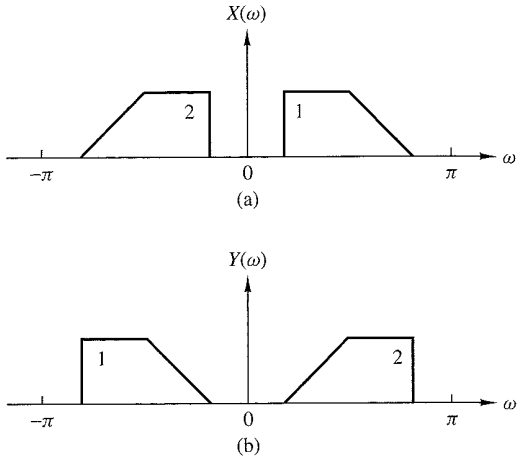



**Figure P5.58**
(a) Original spectrum;
(b) frequency-inverted spectrum.

**5.59** A lowpass filter is described by the difference equation

$$y(n) = 0.9y(n-1) + 0.1x(n)$$

**(a)** By performing a frequency translation of $\pi/2$, transform the filter into a bandpass filter.

**(b)** What is the impulse response of the bandpass filter?

**(c)** What is the major problem with the frequency translation method for transforming a prototype lowpass filter into a bandpass filter?

**5.60** Consider a system with a real-valued impulse response $h(n)$ and frequency response

$$H(\omega) = |H(\omega)|e^{j\theta(\omega)}$$

The quantity

$$D = \sum_{n=-\infty}^{\infty} n^2 h^2(n)$$

provides a measure of the "effective duration" of $h(n)$.

**(a)** Express $D$ in terms of $H(\omega)$.

**(b)** Show that $D$ is minimized for $\theta(\omega) = 0$.

**5.61** Consider the lowpass filter

$$y(n) = ay(n-1) + bx(n), \qquad 0 < a < 1$$

**(a)** Determine $b$ so that $|H(0)| = 1$.

**(b)** Determine the 3-dB bandwidth $\omega_3$ for the normalized filter in part (a).

**(c)** How does the choice of the parameter $a$ affect $\omega_3$?

**(d)** Repeat parts (a) through (c) for the highpass filter obtained by choosing $-1 < a < 0$.

**5.62** Sketch the magnitude and phase response of the multipath channel

$$y(n) = x(n) + \alpha x(n - M), \qquad \alpha > 0$$

for $\alpha << 1$.

**5.63** Determine the system functions and the pole-zero locations for the systems shown in Fig. P5.63(a) through (c), and indicate whether or not the systems are stable.

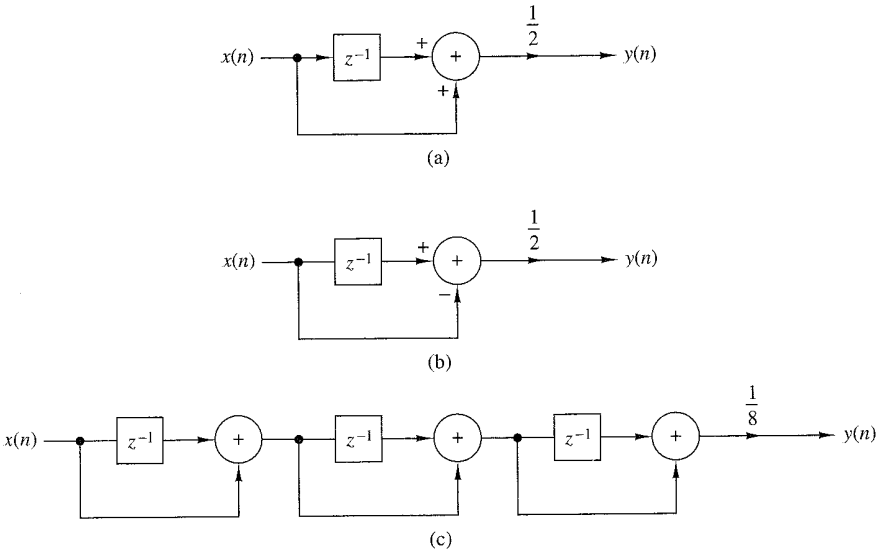



**Figure P5.63**

**5.64** Determine and sketch the impulse response and the magnitude and phase responses of the FIR filter shown in Fig. P5.64 for $b = 1$ and $b = -1$.

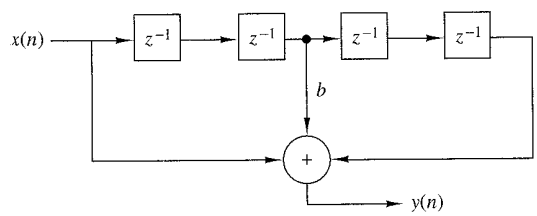



**Figure P5.64**

**5.65** Consider the system

$$y(n) = x(n) - 0.95x(n - 6)$$

**(a)** Sketch its pole–zero pattern.

**(b)** Sketch its magnitude response using the pole–zero plot.

**(c)** Determine the system function of its causal inverse system.

**(d)** Sketch the magnitude response of the inverse system using the pole–zero plot.

**5.66** Determine the impulse response and the difference equation for all possible systems specified by the system functions

**(a)** $H(z) = \dfrac{z^{-1}}{1 - z^{-1} - z^{-2}}$

**(b)** $H(z) = \dfrac{1}{1 - e^{-4a} z^{-4}}, \qquad 0 < a < 1$

**5.67** Determine the impulse response of a causal LTI system which produces the response

$$y(n) = \{\underset{\uparrow}{1}, -1, 3, -1, 6\}$$

when excited by the input signal

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

**5.68** The system

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

is excited with the input

$$x(n) = (\frac{1}{4})^n u(n)$$

Determine the sequences $r_{xx}(l)$, $r_{hh}(l)$, $r_{xy}(l)$, and $r_{yy}(l)$.

**5.69** Determine if the following FIR systems are minimum phase.

**(a)** $h(n) = \{\underset{\uparrow}{10}, 9, -7, -8, 0, 5, 3\}$

**(b)** $h(n) = \{\underset{\uparrow}{5}, 4, -3, -4, 0, 2, 1\}$

**5.70** Can you determine the coefficients of the all-pole system

$$H(z) = \frac{1}{1 + \sum_{k=1}^{N} a_k z^{-k}}$$

if you know its order $N$ and the values $h(0), h(1), \ldots, h(L-1)$ of its impulse response? How? What happens if you do not know $N$?

**5.71** Consider a system with impulse response

$$h(n) = b_0 \delta(n) + b_1 \delta(n - D) + b_2 \delta(n - 2D)$$

**(a)** Explain why the system generates echoes spaced $D$ samples apart.

**(b)** Determine the magnitude and phase response of the system.

**(c)** Show that for $|b_0 + b_2| << |b_1|$, the locations of maxima and minima of $|H(\omega)|^2$ are at

$$\omega = \pm \frac{k}{D} \pi, \qquad k = 0, 1, 2, \ldots$$

**(d)** Plot $|H(\omega)|$ and $\measuredangle H(\omega)$ for $b_0 = 0.1$, $b_1 = 1$, and $b_2 = 0.05$ and discuss the results.

**5.72** Consider the pole–zero system

$$H(z) = \frac{B(z)}{A(z)} = \frac{1 + bz^{-1}}{1 + az^{-1}} = \sum_{n=0}^{\infty} h(n)z^{-n}$$

**(a)** Determine $h(0)$, $h(1)$, $h(2)$, and $h(3)$ in terms of $a$ and $b$.

**(b)** Let $r_{hh}(l)$ be the autocorrelation sequence of $h(n)$. Determine $r_{hh}(0)$, $r_{hh}(1)$, $r_{hh}(2)$, and $r_{hh}(3)$ in terms of $a$ and $b$.

**5.73** Let $x(n)$ be a real-valued minimum-phase sequence. Modify $x(n)$ to obtain another real-valued minimum-phase sequence $y(n)$ such that $y(0) = x(0)$ and $y(n) = |x(n)|$.

**5.74** The frequency response of a stable LTI system is known to be real and even. Is the inverse system stable?

**5.75** Let $h(n)$ be a real filter with nonzero linear or nonlinear phase response. Show that the following operations are equivalent to filtering the signal $x(n)$ with a zero-phase filter.

**(a)** $g(n) = h(n) * x(n)$

$f(n) = h(n) * g(-n)$

$y(n) = f(-n)$

**(b)** $g(n) = h(n) * x(n)$

$f(n) = h(n) * x(-n)$

$y(n) = g(n) + f(-n)$

(*Hint:* Determine the frequency response of the composite system $y(n) = H[x(n)]$.)

**5.76** Check the validity of the following statements:

**(a)** The convolution of two minimum-phase sequences is always a minimum-phase sequence.

**(b)** The sum of two minimum-phase sequences is always minimum phase.

**5.77** Determine the minimum-phase system whose squared magnitude response is given by:

**(a)** $$|H(\omega)|^2 = \frac{\frac{5}{4} - \cos\omega}{\frac{10}{9} - \frac{2}{3}\cos\omega}$$

**(b)** $$|H(\omega)|^2 = \frac{2(1 - a^2)}{(1 + a^2) - 2a\cos\omega}, \qquad |a| < 1$$

**5.78** Consider an FIR system with the following system function:

$$H(z) = (1 - 0.8e^{j\pi/2}z^{-1})(1 - 0.8e^{-j\pi/2}z^{-1})(1 - 1.5e^{j\pi/4}z^{-1})(1 - 1.5e^{-j\pi/4}z^{-1})$$

**(a)** Determine all systems that have the same magnitude response. Which is the minimum-phase system?

**(b)** Determine the impulse response of all systems in part (a).

**(c)** Plot the partial energy

$$E(n) = \sum_{k=0}^{n} h^2(n)$$

for every system and use it to identify the minimum- and maximum-phase systems.

**5.79** The causal system

$$H(z) = \frac{1}{1 + \sum_{k=1}^{N} a_k z^{-k}}$$

is known to be unstable.

We modify this system by changing its impulse response $h(n)$ to

$$h'(n) = \lambda^n h(n) u(n)$$

**(a)** Show that by properly choosing $\lambda$ we can obtain a new stable system.

**(b)** What is the difference equation describing the new system?

**5.80** Given a signal $x(n)$, we can create echoes and reverberations by delaying and scaling the signal as follows

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

where $D$ is positive integer and $g_k > g_{k-1} > 0$.

**(a)** Explain why the comb filter

$$H(z) = \frac{1}{1 - az^{-D}}$$

can be used as a reverberator (i.e., as a device to produce artificial reverberations). (*Hint:* Determine and sketch its impulse response.)

**(b)** The all-pass comb filter

$$H(z) = \frac{z^{-D} - a}{1 - az^{-D}}$$

is used in practice to build digital reverberators by cascading three to five such filters and properly choosing the parameters $a$ and $D$. Compute and plot the impulse response of two such reverberators each obtained by cascading three sections with the following parameters.

UNIT 1

| Section | $D$ | $a$ |
|---|---|---|
| 1 | 50 | 0.7 |
| 2 | 40 | 0.665 |
| 3 | 32 | 0.63175 |

UNIT 2

| Section | $D$ | $a$ |
|---|---|---|
| 1 | 50 | 0.7 |
| 2 | 17 | 0.77 |
| 3 | 6 | 0.847 |

**(c)** The difference between echo and reverberation is that with pure echo there are clear repetitions of the signal, but with reverberations, there are not. How is this reflected in the shape of the impulse response of the reverberator? Which unit in part (b) is a better reverberator?

**(d)** If the delays $D_1, D_2, D_3$ in a certain unit are prime numbers, the impulse response of the unit is more "dense." Explain why.

**(e)** Plot the phase response of units 1 and 2 and comment on them.

**(f)** Plot $h(n)$ for $D_1$, $D_2$, and $D_3$ being nonprime. What do you notice?

More details about this application can be found in a paper by J. A. Moorer, "Signal Processing Aspects of Computer Music: A Survey," *Proc. IEEE,* Vol. 65, No. 8, Aug. 1977, pp. 1108–1137.

**5.81** By trial-and-error design a third-order lowpass filter with cutoff frequency at $\omega_c = \pi/9$ radians/sample interval. Start your search with

**(a)** $z_1 = z_2 = z_3 = 0$, $p_1 = r$, $p_{2,3} = re^{\pm j\omega_c}$, $r = 0.8$

**(b)** $r = 0.9$, $z_1 = z_2 = z_3 = -1$

**5.82** A speech signal with bandwidth $B = 10$ kHz is sampled at $F_2 = 20$ kHz. Suppose that the signal is corrupted by four sinusoids with frequencies

$$F_1 = 10{,}000 \text{ Hz}, \quad F_3 = 7778 \text{ Hz}$$
$$F_2 = 8889 \text{ Hz}, \quad F_4 = 6667 \text{ Hz}$$

**(a)** Design a FIR filter that eliminates these frequency components.

**(b)** Choose the gain of the filter so that $|H(0)| = 1$ and then plot the log magnitude response and the phase response of the filter.

**(c)** Does the filter fulfill your objectives? Do you recommend the use of this filter in a practical application?

**5.83** Compute and sketch the frequency response of a digital resonator with $\omega = \pi/6$ and $r = 0.6, 0.9, 0.99$. In each case, compute the bandwidth and the resonance frequency from the graph, and check if they are in agreement with the theoretical results.

**5.84** The system function of a communication channel is given by

$$H(z) = (1 - 0.9e^{j0.4\pi}z^{-1})(1 - 0.9e^{-j0.4\pi}z^{-1})(1 - 1.5e^{j0.6\pi}z^{-1})(1 - 1.5e^{-j0.6\pi}z^{-1})$$

Determine the system function $H_c(z)$ of a causal and stable compensating system so that the cascade interconnection of the two systems has a flat magnitude response. Sketch the pole–zero plots and the magnitude and phase responses of all systems involved into the analysis process. [*Hint:* Use the decomposition $H(z) = H_{\text{ap}}(z)H_{\text{min}}(z)$.]

ElCoM
Electrical | Computer | Mechatronics
ELCOM-HU.COM

CHAPTER

6

# Sampling and Reconstruction of Signals

In Chapter 1 we treated the sampling of continuous-time signals and demonstrated that if the signals are bandlimited, it is possible to reconstruct the original signal from the samples, provided that the sampling rate is at least twice the highest frequency contained in the signal. We also briefly described the subsequent operations of quantization and coding that are necessary to convert an analog signal to a digital signal appropriate for digital processing.

In this chapter we consider time-domain sampling, analog-to-digital (A/D) conversion (quantization and coding), and digital-to-analog (D/A) conversion (signal reconstruction) in greater depth. We also consider the sampling of signals that are characterized as bandpass signals. The final topic deals with the use of oversampling and sigma-delta modulation in the design of high precision A/D converters.

## 6.1 Ideal Sampling and Reconstruction of Continuous-Time Signals

To process a continuous-time signal using digital signal processing techniques, it is necessary to convert the signal into a sequence of numbers. As was discussed in Section 1.4, this is usually done by sampling the analog signal, say $x_a(t)$, periodically every $T$ seconds to produce a discrete-time signal $x(n)$ given by

$$x(n) = x_a(nT), \qquad -\infty < n < \infty \tag{6.1.1}$$

The relationship (6.1.1) describes the sampling process in the time domain. As discussed in Chapter 1, the sampling frequency $F_s = 1/T$ must be selected large

enough such that the sampling does not cause any loss of spectral information (no aliasing). Indeed, if the spectrum of the analog signal can be recovered from the spectrum of the discrete-time signal, there is no loss of information. Consequently, we investigate the sampling process by finding the relationship between the spectra of signals $x_a(t)$ and $x(n)$.

If $x_a(t)$ is an aperiodic signal with finite energy, its (voltage) spectrum is given by the Fourier transform relation

$$X_a(F) = \int_{-\infty}^{\infty} x_a(t)e^{-j2\pi Ft}\, dt \tag{6.1.2}$$

whereas the signal $x_a(t)$ can be recovered from its spectrum by the inverse Fourier transform

$$x_a(t) = \int_{-\infty}^{\infty} X_a(F)e^{j2\pi Ft}\, dF \tag{6.1.3}$$

Note that utilization of all frequency components in the infinite frequency range $-\infty < F < \infty$ is necessary to recover the signal $x_a(t)$ if the signal $x_a(t)$ is not bandlimited.

The spectrum of a discrete-time signal $x(n)$, obtained by sampling $x_a(t)$, is given by the Fourier transform relation

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} \tag{6.1.4}$$

or, equivalently,

$$X(f) = \sum_{n=-\infty}^{\infty} x(n)e^{-j2\pi fn} \tag{6.1.5}$$

The sequence $x(n)$ can be recovered from its spectrum $X(\omega)$ or $X(f)$ by the inverse transform

$$\begin{aligned} x(n) &= \frac{1}{2\pi}\int_{-\pi}^{\pi} X(\omega)e^{j\omega n}d\omega \\ &= \int_{-1/2}^{1/2} X(f)e^{j2\pi fn}\, df \end{aligned} \tag{6.1.6}$$

In order to determine the relationship between the spectra of the discrete-time signal and the analog signal, we note that periodic sampling imposes a relationship between the independent variables $t$ and $n$ in the signals $x_a(t)$ and $x(n)$, respectively. That is,

$$t = nT = \frac{n}{F_s} \tag{6.1.7}$$

This relationship in the time domain implies a corresponding relationship between the frequency variables $F$ and $f$ in $X_a(F)$ and $X(f)$, respectively.

Indeed, substitution of (6.1.7) into (6.1.3) yields

$$x(n) \equiv x_a(nT) = \int_{-\infty}^{\infty} X_a(F) e^{j2\pi nF/F_s}\, dF \tag{6.1.8}$$

If we compare (6.1.6) with (6.1.8), we conclude that

$$\int_{-1/2}^{1/2} X(f) e^{j2\pi fn}\, df = \int_{-\infty}^{\infty} X_a(F) e^{j2\pi nF/F_s}\, dF \tag{6.1.9}$$

From the development in Chapter 1 we know that periodic sampling imposes a relationship between the frequency variables $F$ and $f$ of the corresponding analog and discrete-time signals, respectively. That is,

$$f = \frac{F}{F_s} \tag{6.1.10}$$

With the aid of (6.1.10), we can make a simple change in variable in (6.1.9), and obtain the result

$$\frac{1}{F_s} \int_{-F_s/2}^{F_s/2} X(F) e^{j2\pi nF/F_s}\, dF = \int_{-\infty}^{\infty} X_a(F) e^{j2\pi nF/F_s}\, dF \tag{6.1.11}$$

We now turn our attention to the integral on the right-hand side of (6.1.11). The integration range of this integral can be divided into an infinite number of intervals of width $F_s$. Thus the integral over the infinite range can be expressed as a sum of integrals, that is,

$$\int_{-\infty}^{\infty} X_a(F) e^{j2\pi nF/F_s}\, dF = \sum_{k=-\infty}^{\infty} \int_{(k-1/2)F_s}^{(k+1/2)F_s} X_a(F) e^{j2\pi nF/F_s}\, dF \tag{6.1.12}$$

We observe that $X_a(F)$ in the frequency interval $(k - \frac{1}{2})F_s$ to $(k + \frac{1}{2})F_s$ is identical to $X_a(F - kF_s)$ in the interval $-F_s/2$ to $F_s/2$. Consequently,

$$\begin{aligned}\sum_{k=-\infty}^{\infty} \int_{(k-1/2)F_s}^{(k+1/2)F_s} X_a(F) e^{j2\pi nF/F_s}\, dF &= \sum_{k=-\infty}^{\infty} \int_{-F_s/2}^{F_s/2} X_a(F - kF_s) e^{j2\pi nF/F_s}\, dF \\ &= \int_{-F_s/2}^{F_s/2} \left[ \sum_{k=-\infty}^{\infty} X_a(F - kF_s) \right] e^{j2\pi nF/F_s}\, dF\end{aligned} \tag{6.1.13}$$

where we have used the periodicity of the complex exponential, namely,

$$e^{j2\pi n(F+kF_s)/F_s} = e^{j2\pi nF/F_s}$$

Comparing (6.1.13), (6.1.12), and (6.1.11), we conclude that

$$X(F) = F_s \sum_{k=-\infty}^{\infty} X_a(F - kF_s) \tag{6.1.14}$$

or, equivalently,

$$X(f) = F_s \sum_{k=-\infty}^{\infty} X_a[(f - k)F_s] \tag{6.1.15}$$

This is the desired relationship between the spectrum $X(F)$ or $X(f)$ of the discrete-time signal and the spectrum $X_a(F)$ of the analog signal. The right-hand side of (6.1.14) or (6.1.15) consists of a periodic repetition of the scaled spectrum $F_s X_a(F)$ with period $F_s$. This periodicity is necessary because the spectrum $X(f)$ of the discrete-time signal is periodic with period $f_p = 1$ or $F_p = F_s$.

For example, suppose that the spectrum of a band-limited analog signal is as shown in Fig. 6.1.1(a). The spectrum is zero for $|F| \geq B$. Now, if the sampling frequency $F_s$ is selected to be greater than $2B$, the spectrum $X(F)$ of the discrete-time signal will appear as shown in Fig. 6.1.1(b). Thus, if the sampling frequency $F_s$ is selected such that $F_s \geq 2B$, where $2B$ is the Nyquist rate, then

$$X(F) = F_s X_a(F), \qquad |F| \leq F_s/2 \tag{6.1.16}$$

In this case there is no aliasing and therefore the spectrum of the discrete-time signal is identical (within the scale factor $F_s$) to the spectrum of the analog signal, within the fundamental frequency range $|F| \leq F_s/2$ or $|f| \leq \frac{1}{2}$.

On the other hand, if the sampling frequency $F_s$ is selected such that $F_s < 2B$, the periodic continuation of $X_a(F)$ results in spectral overlap, as illustrated in Fig. 6.1.1(c) and (d). Thus the spectrum $X(F)$ of the discrete-time signal contains aliased frequency components of the analog signal spectrum $X_a(F)$. The end result is that the aliasing which occurs prevents us from recovering the original signal $x_a(t)$ from the samples.

Given the discrete-time signal $x(n)$ with the spectrum $X(F)$, as illustrated in Fig. 6.1.1(b), with no aliasing, it is now possible to reconstruct the original analog signal from the samples $x(n)$. Since in the absence of aliasing

$$X_a(F) = \begin{cases} \dfrac{1}{F_s} X(F), & |F| \leq F_s/2 \\ 0, & |F| > F_s/2 \end{cases} \tag{6.1.17}$$

and by the Fourier transform relationship (6.1.5),

$$X(F) = \sum_{n=-\infty}^{\infty} x(n) e^{-j2\pi F n/F_s} \tag{6.1.18}$$

the inverse Fourier transform of $X_a(F)$ is

$$x_a(t) = \int_{-F_s/2}^{F_s/2} X_a(F) e^{j2\pi F t}\, dF \tag{6.1.19}$$

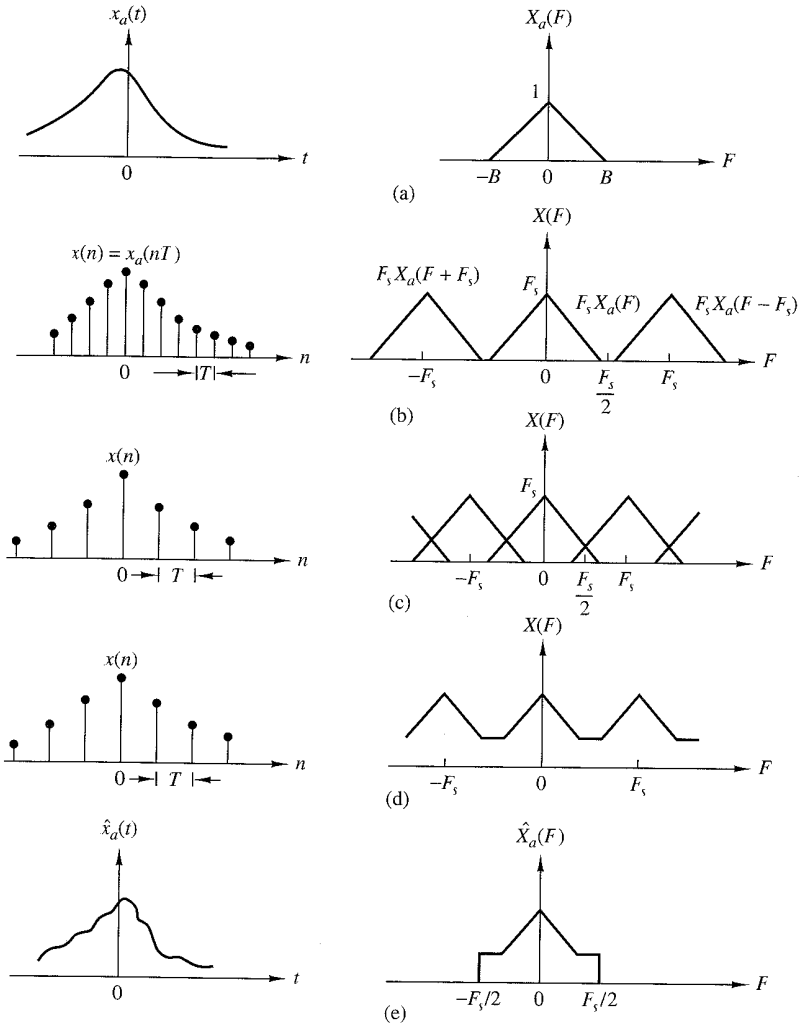



**Figure 6.1.1** Sampling of an analog bandlimited signal and aliasing of spectral components.

Let us assume that $F_s \geq 2B$. With the substitution of (6.1.17) into (6.1.19), we have

$$
\begin{aligned}
x_a(t) &= \frac{1}{F_s} \int_{-F_s/2}^{F_s/2} \left[ \sum_{n=-\infty}^{\infty} x(n) e^{-j2\pi F n/F_s} \right] e^{j2\pi F t}\, dF \\
&= \frac{1}{F_s} \sum_{n=-\infty}^{\infty} x(n) \int_{-F_s/2}^{F_s/2} e^{j2\pi F(t - n/F_s)}\, dF \qquad (6.1.20) \\
&= \sum_{n=-\infty}^{\infty} x_a(nT) \frac{\sin(\pi/T)(t - nT)}{(\pi/T)(t - nT)}
\end{aligned}
$$

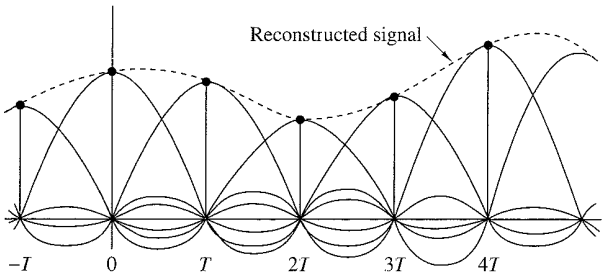



**Figure 6.1.2**
Reconstruction of a continuous-time signal using ideal interpolation.

where $x(n) = x_a(nT)$ and where $T = 1/F_s$ is the sampling interval. This is the reconstruction formula given by (1.4.24) in our discussion of the sampling theorem.

The reconstruction formula in (6.1.20) involves the function

$$g(t) = \frac{\sin(\pi/T)t}{(\pi/T)t} \tag{6.1.21}$$

appropriately shifted by $nT$, $n = 0, \pm 1, \pm 2, \ldots$, and multiplied or weighted by the corresponding samples $x_a(nT)$ of the signal. We call (6.1.20) an interpolation formula for reconstructing $x_a(t)$ from its samples, and $g(t)$, given in (6.1.21), is the interpolation function. We note that at $t = kT$, the interpolation function $g(t - nT)$ is zero except at $k = n$. Consequently, $x_a(t)$ evaluated at $t = kT$ is simply the sample $x_a(kT)$. At all other times the weighted sums of the time-shifted versions of the interpolation function combine to yield exactly $x_a(t)$. This combination is illustrated in Fig. 6.1.2.

The formula in (6.1.20) for reconstructing the analog signal $x_a(t)$ from its samples is called the *ideal interpolation formula*. It forms the basis for the *sampling theorem*, which can be stated as follows.

**Sampling Theorem.** A bandlimited continuous-time signal, with highest frequency (bandwidth) $B$ hertz, can be uniquely recovered from its samples provided that the sampling rate $F_s \geq 2B$ samples per second.

According to the sampling theorem and the reconstruction formula in (6.1.20), the recovery of $x_a(t)$ from its samples $x(n)$ requires an infinite number of samples. However, in practice we use a finite number of samples of the signal and deal with finite-duration signals. As a consequence, we are concerned only with reconstructing a finite-duration signal from a finite number of samples.

When aliasing occurs due to too low a sampling rate, the effect can be described by a multiple folding of the frequency axis of the frequency variable $F$ for the analog signal. Figure 6.1.3(a) shows the spectrum $X_a(F)$ of an analog signal. According to (6.1.14), sampling of the signal with a sampling frequency $F_s$ results in a periodic repetition of $X_a(F)$ with period $F_s$. If $F_s < 2B$, the shifted replicas of $X_a(F)$ overlap. The overlap that occurs within the fundamental frequency range $-F_s/2 \leq F \leq F_s/2$ is illustrated in Fig. 6.1.3(b). The corresponding spectrum of the discrete-time signal within the fundamental frequency range is obtained by adding all the shifted portions within the range $|f| \leq \frac{1}{2}$, to yield the spectrum shown in Fig. 6.1.3(c).

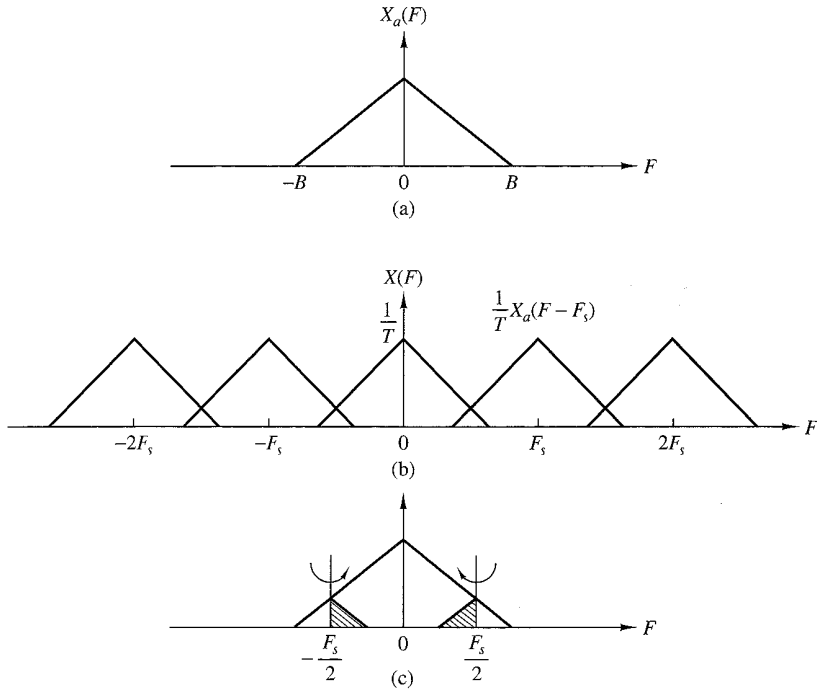



**Figure 6.1.3** Illustration of aliasing around the folding frequency.

A careful inspection of Fig. 6.1.3(a) and (b) reveals that the aliased spectrum in Fig. 6.1.3(c) can be obtained by folding the original spectrum like an accordion with pleats at every odd multiple of $F_s/2$. Consequently, the frequency $F_s/2$ is called the *folding frequency*, as indicated in Chapter 1. Clearly, then, periodic sampling automatically forces a folding of the frequency axis of an analog signal at odd multiples of $F_s/2$, and this results in the relationship $F = fF_s$ between the frequencies for continuous-time signals and discrete-time signals. Due to the folding of the frequency axis, the relationship $F = fF_s$ is not truly linear, but piecewise linear, to accommodate for the aliasing effect. This relationship is illustrated in Fig. 6.1.4.

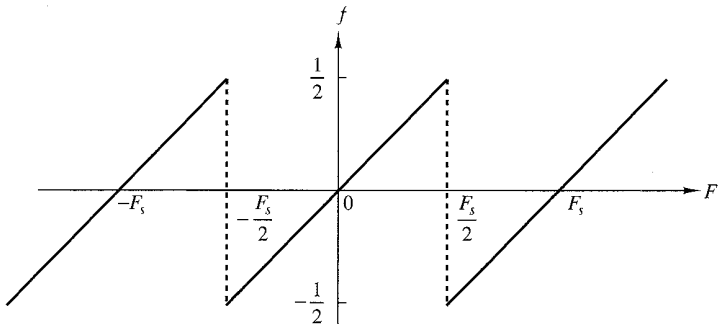



**Figure 6.1.4** Relationship between frequency variables $F$ and $f$.

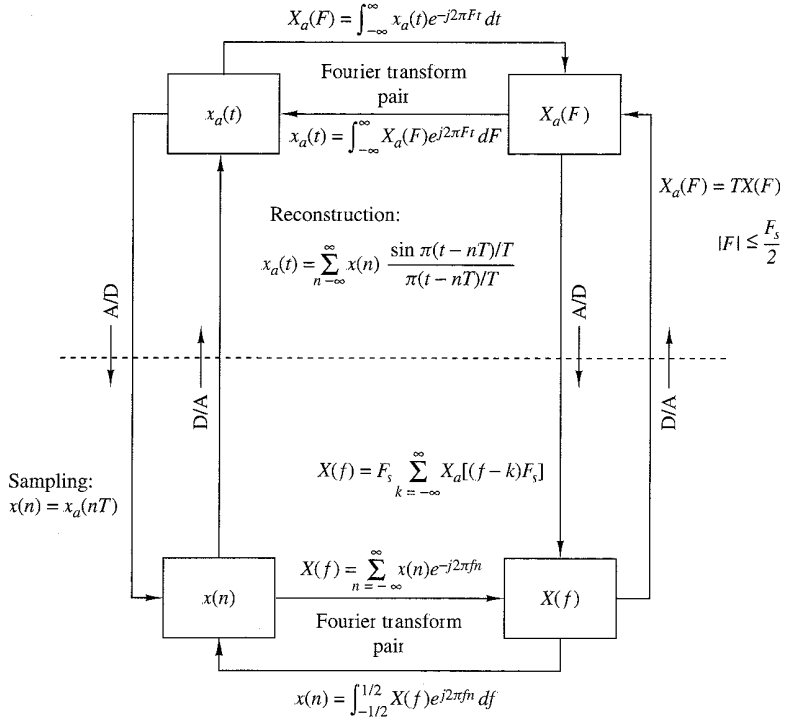



**Figure 6.1.5** Time-domain and frequency-domain relationships for sampled signals.

If the analog signal is bandlimited to $B \le F_s/2$, the relationship between $f$ and $F$ is linear and one-to-one. In other words, there is no aliasing. In practice, prefiltering with an antialiasing filter is usually employed prior to sampling. This ensures that frequency components of the signal above $F \ge B$ are sufficiently attenuated so that, if aliased, they cause negligible distortion on the desired signal.

The relationships among the time-domain and frequency-domain functions $x_a(t)$, $x(n)$, $X_a(F)$, and $X(f)$ are summarized in Fig. 6.1.5. The relationships for recovering the continuous-time functions, $x_a(t)$ and $X_a(F)$, from the discrete-time quantities $x(n)$ and $X(f)$, assume that the analog signal is bandlimited and that it is sampled at the Nyquist rate (or faster).

The following examples serve to illustrate the problem of the aliasing of frequency components.

### EXAMPLE 6.1.1 Aliasing in Sinusoidal Signals

The continuous-time signal

$$x_a(t) = \cos 2\pi F_0 t = \frac{1}{2}e^{j2\pi F_0 t} + \frac{1}{2}e^{-j2\pi F_0 t}$$

has a discrete spectrum with spectral lines at $F = \pm F_0$, as shown in Fig. 6.1.6(a). The process

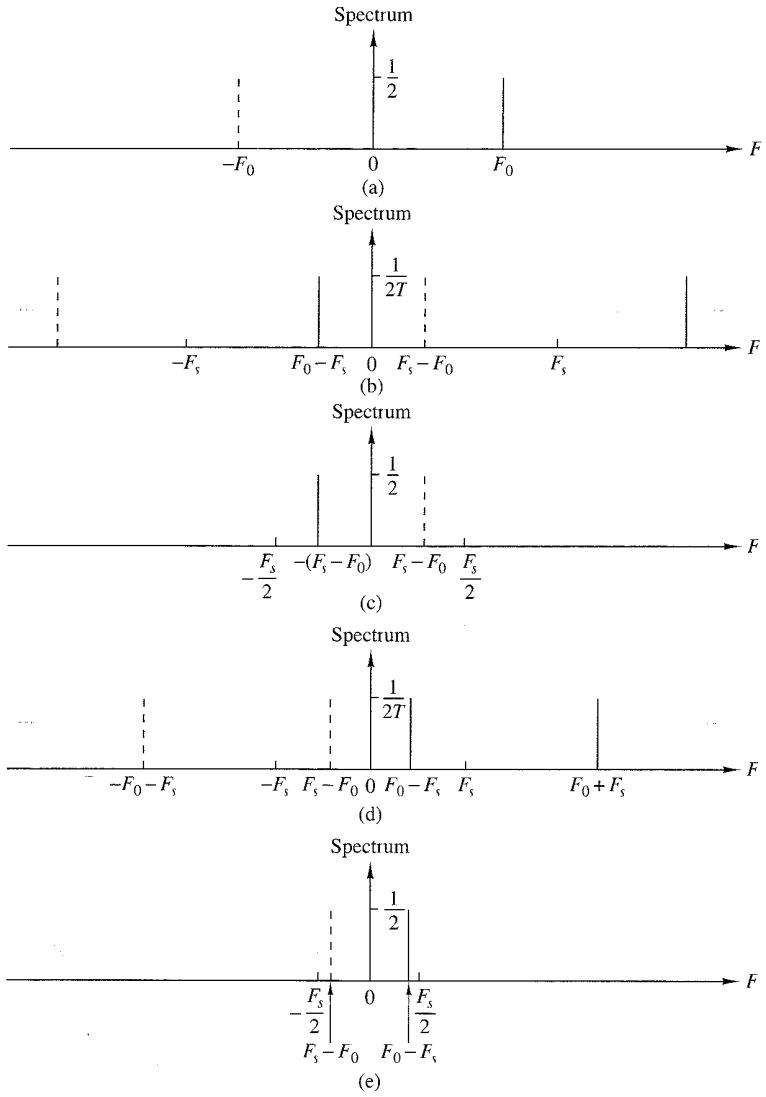



**Figure 6.1.6** Aliasing of sinusoidal signals.

of sampling this signal with a sampling frequency $F_s$ introduces replicas of the spectrum about multiples of $F_s$. This is illustrated in Fig. 6.1.6(b) for $F_s/2 < F_0 < F_s$.

To reconstruct the continuous-time signal, we should select the frequency components inside the fundamental frequency range $|F| \leq F_s/2$. The resulting spectrum is shown in Fig. 6.1.6(c). The reconstructed signal is

$$\hat{x}_a(t) = \cos 2\pi(F_s - F_0)t$$

Now, if $F_s$ is selected such that $F_s < F_0 < 3F_s/2$, the spectrum of the sampled signal is shown in Fig. 6.1.6(d). The reconstructed signal, shown in Fig. 6.1.6(e), is

$$x_a(t) = \cos 2\pi(F_0 - F_s)t$$

In both cases, aliasing has occurred, so that the frequency of the reconstructed signal is an aliased version of the frequency of the original signal.

---

### EXAMPLE 6.1.2 Sampling and Reconstruction of a Nonbandlimited Signal

Consider the following continuous-time two-sided exponential signal:

$$x_a(t) = e^{-A|t|} \xleftrightarrow{\mathcal{F}} X_a(F) = \frac{2A}{A^2 + (2\pi F)^2}, \quad A > 0$$

**(a)** Determine the spectrum of the sampled signal $x(n) = x_a(nT)$. **(b)** Plot the signals $x_a(t)$ and $x(n) = x_a(nT)$, for $T = 1/3$ sec and $T = 1$ sec, and their spectra. **(c)** Plot the continuous-time signal $\hat{x}_a(t)$ after reconstruction with ideal bandlimited interpolation.

**Solution.**

**(a)** If we sample $x_a(nT)$ with a sampling frequency $F_s = 1/T$, we have

$$x(n) = x_a(nT) = e^{-AT|n|} = (e^{-AT})^{|n|}, \quad -\infty < n < \infty$$

The spectrum of $x(n)$ can be found easily if we use a direct computation of the discrete-time Fourier transform. We find that

$$X(F) = \frac{1 - a^2}{1 - 2a\cos 2\pi(F/F_s) + a^2}, \quad a = e^{-AT}$$

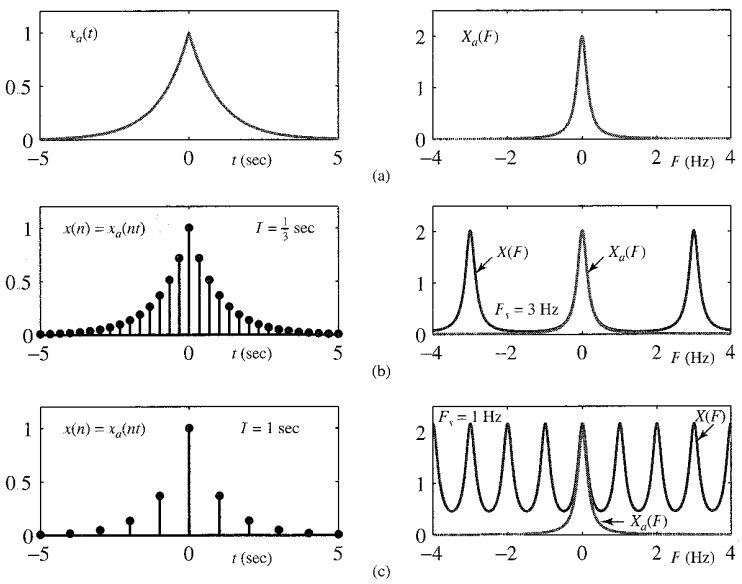



**Figure 6.1.7** (a) Analog signal $x_a(t)$ and its spectrum $X_a(F)$; (b) $x(n) = x_a(nT)$ and its spectrum for $F_s = 3$ Hz; and (c) $x(n) = x_a(nT)$ and its spectrum for $F_s = 1$ Hz.

Clearly, since $\cos 2\pi(F/F_s)$ is periodic with period $F_s$, so is the spectrum $X(F)$.

**(b)** Since $X_a(F)$ is not bandlimited, aliasing cannot be avoided. Figure 6.1.7(a) shows the original signal $x_a(t)$ and its spectrum $X_a(F)$ for $A = 1$. The sampled signal $x(n)$ and its spectrum $X(F)$ are shown for $F_s = 3$ Hz and $F_s = 1$ Hz in Figures 6.1.7(b) and 6.1.7(c). The aliasing distortion is clearly noticeable in the frequency domain when $F_s = 1$ Hz and almost unnoticeable when $F_s = 3$ Hz.

**(c)** The spectrum $\hat{X}_a(F)$ of the reconstructed signal $\hat{x}_a(t)$ is given by

$$\hat{X}_a(F) = \begin{cases} TX(F), & |F| \le F_s/2 \\ 0, & \text{otherwise} \end{cases}$$

The values of $\hat{x}_a(t)$ can be evaluated for plotting purposes using the ideal bandlimited interpolation formula (6.1.20) for all significant values of $x(n)$ and $\sin(\pi t/T)/(\pi t/T)$. Figure 6.1.8 illustrates the reconstructed signal and its spectrum for $F_s = 3$ Hz and $F_s = 1$ Hz. It is interesting to note that in every case $\hat{x}_a(nT) = x_a(nt)$, but $\hat{x}_a(nt) \neq x_a(nT)$ for $t \neq nT$. The results of aliasing are clearly evident in the spectrum of $\hat{x}_a(t)$ for $F_s = 1$ Hz, where we note how the folding of the spectrum about $F = \pm 0.5$ Hz increases the high frequency content of $\hat{X}_a(F)$.

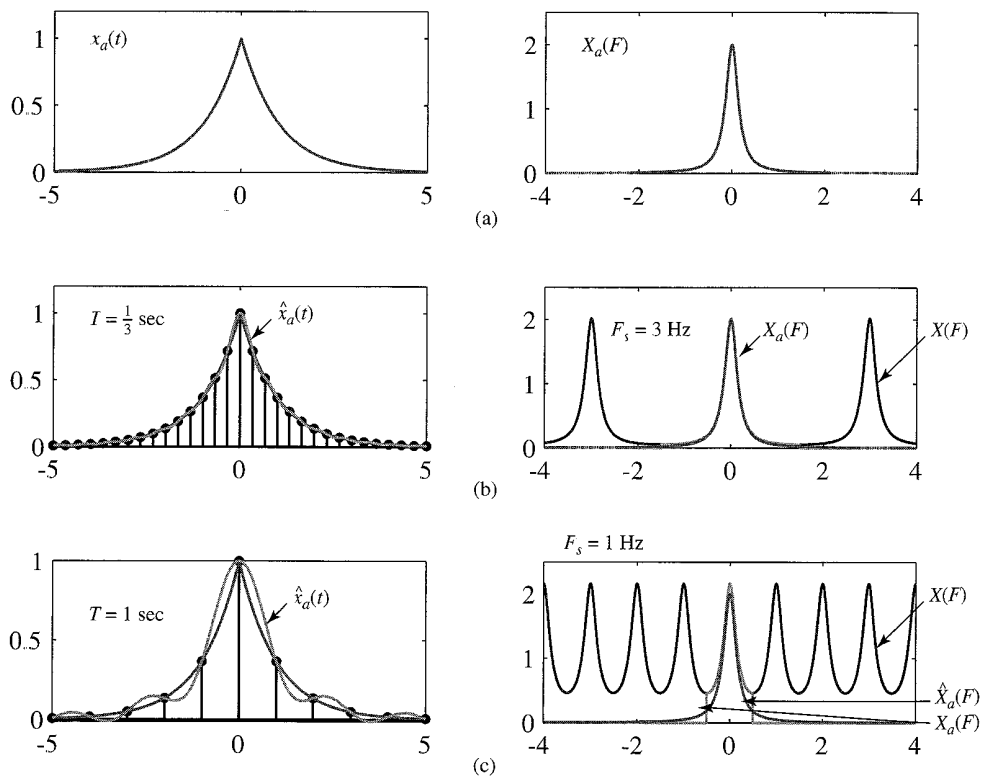



**Figure 6.1.8** (a) Analog signal $x_a(t)$ and its spectrum $X_a(F)$; (b) reconstructed signal $\hat{x}_a(t)$ and its spectrum for $F_s = 3$ Hz; and (c) reconstructed signal $\hat{x}_a(t)$ and its spectrum for $F_s = 1$ Hz

## 6.2 Discrete-Time Processing of Continuous-Time Signals

Many practical applications require the discrete-time processing of continuous-time signals. Figure 6.2.1 shows the configuration of a general system used to achieve this objective. In designing the processing to be performed, we must first select the bandwidth of the signal to be processed, since the signal bandwidth determines the minimum sampling rate. For example, a speech signal, which is to be transmitted digitally, can contain frequency components above 3000 Hz, but for purposes of speech intelligibility and speaker identification, the preservation of frequency components bellow 3000 Hz is sufficient. Consequently, it would be inefficient from a processing viewpoint to preserve the higher-frequency components and wasteful of channel bandwidth to transmit the extra bits needed to represent these higher-frequency components in the speech signal. Once the desired frequency band is selected we can specify the sampling rate and the characteristics of the prefilter, which is also called an antialiasing filter.

The prefilter is an analog filter which has a twofold purpose. First, it ensures that the bandwidth of the signal to be sampled is limited to the desired frequency range. Thus any frequency components of the signal above $F_s/2$ are sufficiently attenuated so that the amount of signal distortion due to aliasing is negligible. For example, the speech signal to be transmitted digitally over a telephone channel would be filtered by a lowpass filter having a passband extending to 3000 Hz, a transition band of approximately 400 to 500 Hz, and a stopband above 3400 to 3500 Hz. The speech signal may be sampled at 8000 Hz and hence the folding frequency would be 4000 Hz. Thus aliasing would be negligible. Another reason for using a prefilter is to limit the additive noise spectrum and other interference, which often corrupts the desired signal. Usually, additive noise is wideband and exceeds the bandwidth of the desired signal. By prefiltering we reduce the additive noise power to that which falls within the bandwidth of the desired signal and we reject the out-of-band noise.

Once we have specified the prefilter requirements and have selected the desired sampling rate, we can proceed with the design of the digital signal processing operations to be performed on the discrete-time signal. The selection of the sampling rate $F_s = 1/T$, where $T$ is the sampling interval, not only determines the highest frequency $(F_s/2)$ that is preserved in the analog signal, but also serves as a scale factor that influences the design specifications for digital filters and any other discrete-time systems through which the signal is processed.

The ideal A/D converter and the ideal D/A converter provide the interface between the continuous-time and discrete-time domains. The overall system is equivalent to a continuous-time system, which may or may not be linear and time-invariant

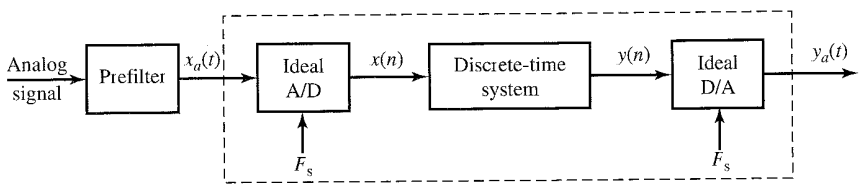



**Figure 6.2.1** System for the discrete-time processing of continuous-time signals.

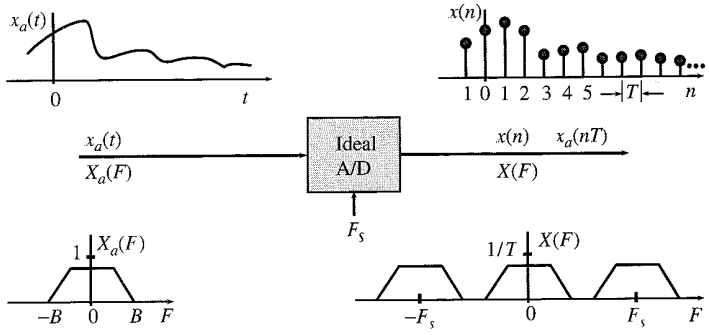



**Figure 6.2.2** Characteristics of an ideal A/D converter in the time and frequency domains.

(even if the discrete-time system is linear and time invariant) because ideal A/D and D/A converters are time-varying operations.

Figure 6.2.2 summarizes the input-output characteristics of an ideal A/D converter in the time and frequency domains. We recall that if $x_a(t)$ is the input signal and $x(n)$ is the output signal, we have

$$x(n) = x_a(t)|_{t=nT} = x_a(nT) \qquad \text{(Time domain)} \tag{6.2.1}$$

$$X(F) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_a(F - kF_s) \qquad \text{(Frequency domain)} \tag{6.2.2}$$

Basically, the ideal A/D converter is a linear time-varying system that (a) scales the analog spectrum by a factor $F_s = 1/T$ and (b) creates a periodic repetition of the scaled spectrum with period $F_s$.

The input–output characteristics of the ideal D/A converter are illustrated in Figure 6.2.3. In the time domain the input and output signals are related by

$$y_a(t) = \sum_{n=-\infty}^{\infty} y(n) g_a(t - nT), \qquad \text{(Time domain)} \tag{6.2.3}$$

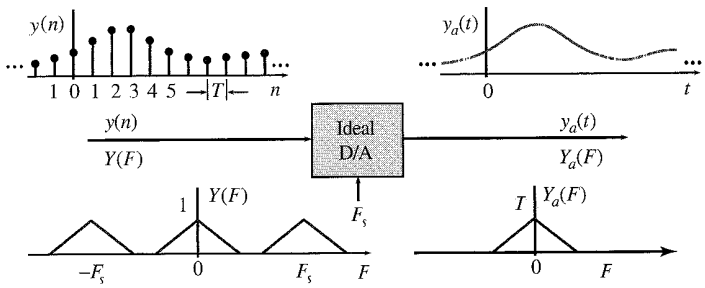



**Figure 6.2.3** Characteristics of an ideal D/A converter in the time and frequency domains.

where

$$g_a(t) = \frac{\sin(\pi t/T)}{\pi t/T} \overset{\mathcal{F}}{\longleftrightarrow} G_a(F) = \begin{cases} T, & |F| \le F_s/2 \\ 0, & \text{otherwise} \end{cases} \tag{6.2.4}$$

is the interpolation function of the ideal D/A. We emphasize that (6.2.3) looks like, but it is *not*, a convolution because the D/A is a linear time-varying system with a discrete-time input and a continuous-time output. To obtain a frequency-domain description, we evaluate the Fourier transform of the output signal

$$\begin{aligned} Y_a(F) &= \sum_{n=-\infty}^{\infty} y(n) \int_{-\infty}^{\infty} g_a(t - nT) e^{-j2\pi Ft}\, dt \\ &= \sum_{n=-\infty}^{\infty} y(n) G_a(F) e^{-j2\pi FnT} \end{aligned}$$

After taking $G_a(F)$ outside the summation, we obtain

$$Y_a(F) = G_a(F)Y(F) \qquad \text{(Frequency domain)} \tag{6.2.5}$$

where $Y(F)$ is the discrete-time Fourier transform of $y(n)$. We note that the ideal D/A converter (a) scales the input spectrum by a factor $T = 1/F_s$ and (b) removes the frequency components for $|F| > F_s/2$. Basically, the ideal D/A acts as a frequency window that "removes" the discrete-time spectral periodicity to generate an aperiodic continuous-time signal spectrum. We stress once again that the ideal D/A is *not* a continuous-time ideal lowpass filter because (6.2.3) is not a continuous-time convolution integral.

Suppose now that we are given a continuous-time LTI system defined by

$$\tilde{y}_a(t) = \int_{-\infty}^{\infty} h_a(\tau) x_a(t - \tau)\, dt \tag{6.2.6}$$

$$\tilde{Y}_a(F) = H_a(F) X_a(F) \tag{6.2.7}$$

and we wish to determine whether there is a discrete-time system $H(F)$ such that the entire system in Figure 6.2.1 is equivalent to the continuous-time system $H_a(F)$. If $x_a(t)$ is not bandlimited or it is bandlimited but $F_s < 2B$, it is impossible to find such a system due to the presence of aliasing. However, if $x_a(t)$ is bandlimited and $F_s > 2B$, we have $X(F) = X_a(F)/T$ for $|F| \le F_s/2$. Therefore, the output of the system in Figure 6.2.1 is given by

$$Y_a(F) = H(F)X(F)G_a(F) = \begin{cases} H(F)X_a(F), & |F| \le F_s/2 \\ 0, & |F| > F_s/2 \end{cases} \tag{6.2.8}$$

To assure that $y_a(t) = \hat{y}_a(t)$, we should choose the discrete-time system so that

$$H(F) = \begin{cases} H_a(F), & |F| \le F_s/2 \\ 0, & |F| > F_s/2 \end{cases} \tag{6.2.9}$$

We note that, in this special case, the cascade connection of the A/D converter (linear time-varying system), an LTI system, and the D/A converter (linear time-varying system) is equivalent to a continuous-time LTI system. This important result provides the theoretical basis for the discrete-time filtering of continuous-time signals. These concepts are illustrated in the following examples.

### EXAMPLE 6.2.1 Simulation of an analog integrator

Consider the analog integrator circuit shown in Figure 6.2.4(a). Its input–output relation is given by

$$RC\frac{dy_a(t)}{dt} + y_a(t) = x_a(t)$$

Taking the Fourier transform of both sides, we can show that the frequency response of the integrator is

$$H_a(F) = \frac{Y_a(F)}{X_a(F)} = \frac{1}{1 + jF/F_c}, \quad F_c = \frac{1}{2\pi RC}$$

Evaluating the inverse Fourier transform yields the impulse response

$$h_a(t) = Ae^{-At}u(t), \quad A = \frac{1}{RC}$$

Clearly the impulse response $h_a(t)$ is a nonbandlimited signal. We now define a discrete-time system by sampling the continuous-time impulse response as follows:

$$h(n) = h_a(nT) = A(e^{-AT})^n u(n)$$

We say that the discrete-time system is obtained from the continuous-time system through an *impulse-invariance* transformation (see Section 10.3.2). The system function and the difference equation of the discrete-time system are

$$H(z) = \sum_{n=0}^{\infty} A(e^{-AT})^n z^{-n} = \frac{1}{1 - e^{-AT}z^{-1}}$$

$$y(n) = e^{-AT}y(n-1) + Ax(n)$$

The system is causal and has a pole $p = e^{-AT}$. Since $A > 0$, $|p| < 1$ and the system is always stable. The frequency response of the system is obtained by evaluating $H(z)$ for $z = e^{j2\pi F/F_s}$. Figure 6.2.4(b) shows the magnitude frequency responses of the analog integrator and the discrete-time simulator for $F_s$ =50, 100, 200, and 1000 Hz. We note that the effects of aliasing, caused by the sampling of $h_a(t)$, become negligible only for sampling frequencies larger than 1 kHz. The discrete-time implementation is accurate for input signals with bandwidth much less than the sampling frequency.

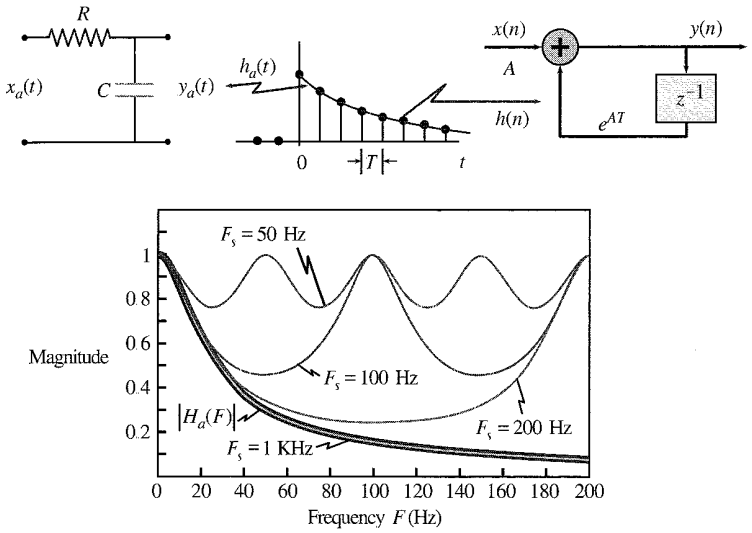



**Figure 6.2.4** Discrete-time implementation of an analog integrator using impulse response sampling. The approximation is satisfactory when the bandwidth of the input signal is much less than the sampling frequency.

---

### EXAMPLE 6.2.2 Ideal bandlimited differentiator

The ideal continuous-time differentiator is defined by

$$y_a(t) = \frac{dx_a(t)}{dt} \tag{6.2.10}$$

and has frequency response function

$$H_a(F) = \frac{Y_a(F)}{X_a(F)} = j2\pi F \tag{6.2.11}$$

For processing bandlimited signals, it is sufficient to use the ideal bandlimited differentiator defined by

$$H_a(F) = \begin{cases} j2\pi F, & |F| \le F_c \\ 0, & |F| > F_c \end{cases} \tag{6.2.12}$$

If we choose $F_s = 2F_c$, we can define an ideal discrete time differentiator by

$$H(F) = H_a(F) = j2\pi F, \quad |F| \le F_s/2 \tag{6.2.13}$$

Since by definition $H(F) = \sum_k H_a(F - kF_s)$, we have $h(n) = h_a(nT)$. In terms of $\omega = 2\pi F/F_s$, $H(\omega)$ is periodic with period $2\pi$. Therefore, the discrete-time impulse response is given by

$$h(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H(\omega)e^{j\omega n}d\omega = \frac{\pi n \cos \pi n - \sin \pi n}{\pi n^2 T} \tag{6.2.14}$$

or in a more compact form

$$h(n) = \begin{cases} 0, & n = 0 \\ \frac{\cos \pi n}{nT}, & n \neq 0 \end{cases} \tag{6.2.15}$$

The magnitude and phase responses of the continuous-time and discrete-time ideal differentiators are shown in Figure 6.2.5.

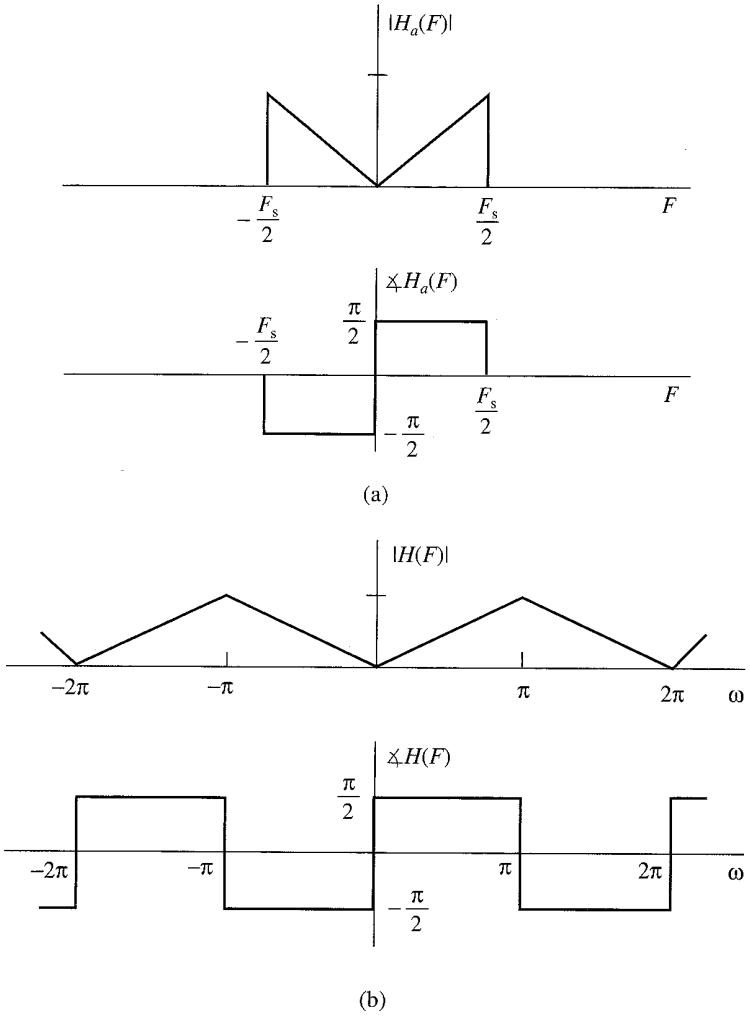



**Figure 6.2.5** Frequency responses of the ideal bandlimited continuous-time differentiator (a) and its discrete-time counterpart (b).

EXAMPLE 6.2.3 **Fractional delay**

A continuous-time delay system is defined by

$$y_a(t) = x_a(t - t_d) \tag{6.2.16}$$

for any $t_d > 0$. Although the concept is simple, its practical implementation is quite complicated. If $x_a(t)$ is bandlimited and sampled at the Nyquist rate, we obtain

$$y(n) = y_a(nT) = x_a(nT - t_d) = x_a[(n - \Delta)T] = x(n - \Delta) \tag{6.2.17}$$

where $\Delta = t_d/T$. If $\Delta$ is an integer, delaying the sequence $x(n)$ is a simple process. For noninteger values of $\Delta$, the delayed value of $x(n)$ would lie somewhere between two samples. However, this value is unavailable and the only way to generate an appropriate value is by ideal bandlimited interpolation. One way to approach this problem is by considering the frequency response

$$H_{\text{id}}(\omega) = e^{-j\omega\Delta} \tag{6.2.18}$$

of the delay system in (6.2.17) and its impulse response

$$h_{\text{id}}(n) = \frac{1}{2\pi}\int_{-\pi}^{\pi} H(\omega)e^{j\omega n}d\omega = \frac{\sin \pi(n - \Delta)}{\pi(n - \Delta)} \tag{6.2.19}$$

When the delay $\Delta$ assumes integer values, $h_{\text{id}}(n)$ reduces to $\delta(n - \Delta)$ because the sin function is sampled at the zero crossings. When $\Delta$ is noninteger, $h_{\text{id}}(n)$ is infinitely long because the sampling times fall between the zero crossings. Unfortunately, the ideal impulse response for fractional delay systems is noncausal and has infinite duration. Therefore, the frequency response (6.2.18) has to be approximated with realizable FIR or IIR filters. More details about fractional delay filter design can be found in Laakso et al. (1996). Implementation of fractional delay using sampling rate conversion techniques is discussed in Section 11.8.

## 6.3 Analog-to-Digital and Digital-to-Analog Converters

In the previous section we assumed that the A/D and D/A converters in the processing of continuous-time signals are ideal. The one implicit assumption that we have made in the discussion on the equivalence of continuous-time and discrete-time signal processing is that the quantization error in analog-to-digital conversion and round-off errors in digital signal processing are negligible. These issues are further discussed in this section. However, we should emphasize that analog signal processing operations cannot be done very precisely either, since electronic components in analog systems have tolerances and they introduce noise during their operation. In general, a digital system designer has better control of tolerances in a digital signal processing system than an analog system designer who is designing an equivalent analog system.

The discussion in Section 6.1 focused on the conversion of continuous-time signals to discrete-time signals using an ideal sampler and ideal interpolation. In this section we deal with the devices for performing these conversions from analog to digital.

### 6.3.1 Analog-to-Digital Converters

Recall that the process of converting a continuous-time (analog) signal to a digital sequence that can be processed by a digital system requires that we quantize the sampled values to a finite number of levels and represent each level by a number of bits.

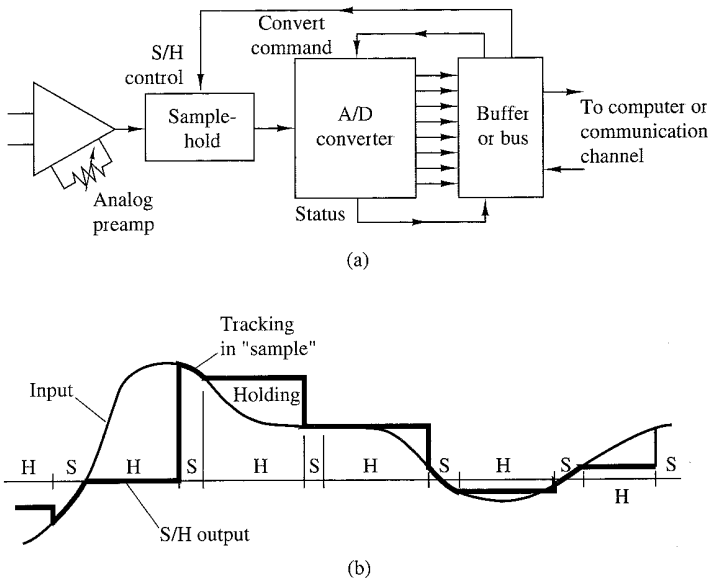



**Figure 6.3.1** (a) Block diagram of basic elements of an A/D converter; (b) time-domain response of an ideal S/H circuit.

Figure 6.3.1(a) shows a block diagram of the basic elements of an A/D converter. In this section we consider the performance requirements for these elements. Although we focus mainly on ideal system characteristics, we shall also mention some key imperfections encountered in practical devices and indicate how they affect the performance of the converter. We concentrate on those aspects that are more relevant to signal processing applications. The practical aspects of A/D converters and related circuitry can be found in the manufacturers' specifications and data sheets.

In practice, the sampling of an analog signal is performed by a sample-and-hold (S/H) circuit. The sampled signal is then quantized and converted to digital form. Usually, the S/H is integrated into the A/D converter. The S/H is a digitally controlled analog circuit that tracks the analog input signal during the sample mode, and then holds it fixed during the hold mode to the instantaneous value of the signal at the time the system is switched from the sample mode to the hold mode. Figure 6.3.1(b) shows the time-domain response of an ideal S/H circuit (i.e., an S/H that responds instantaneously and accurately).

The goal of the S/H is to continuously sample the input signal and then to hold that value constant as long as it takes for the A/D converter to obtain its digital representation. The use of an S/H allows the A/D converter to operate more slowly compared to the time actually used to acquire the sample. In the absence of an S/H, the input signal must not change by more than one-half of the quantization step during the conversion, which may be an impractical constraint. Consequently, the S/H is crucial in high-resolution (12 bits per sample or higher) digital conversion of signals that have large bandwidths (i.e., they change very rapidly).

An ideal S/H introduces no distortion in the conversion process and is accurately modeled as an ideal sampler. However, time-related degradations such as errors in the periodicity of the sampling process ("jitter"), nonlinear variations in the duration of the sampling aperture, and changes in the voltage held during conversion ("droop") do occur in practical devices.

The A/D converter begins the conversion after it receives a convert command. The time required to complete the conversion should be less than the duration of the hold mode of the S/H. Furthermore, the sampling period $T$ should be larger than the duration of the sample mode and the hold mode.

In the following sections we assume that the S/H introduces negligible errors and we focus on the digital conversion of the analog samples.

### 6.3.2 Quantization and Coding

The basic task of the A/D converter is to convert a continuous range of input amplitudes into a discrete set of digital code words. This conversion involves the processes of *quantization* and *coding*. Quantization is a nonlinear and noninvertible process that maps a given amplitude $x(n) \equiv x(nT)$ at time $t = nT$ into an amplitude $x_k$, taken from a finite set of values. The procedure is illustrated in Fig. 6.3.2(a), where the signal amplitude range is divided into $L$ intervals

$$I_k = \{x_k < x(n) \leq x_{k+1}\}, \qquad k = 1, 2, \ldots, L \tag{6.3.1}$$

by the $L + 1$ *decision levels* $x_1, x_2, \ldots, x_{L+1}$. The possible outputs of the quantizer (i.e., the quantization levels) are denoted as $\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_L$. The operation of the quantizer is defined by the relation

$$x_q(n) \equiv Q[x(n)] = \hat{x}_k, \qquad \text{if } x(n) \in I_k \tag{6.3.2}$$

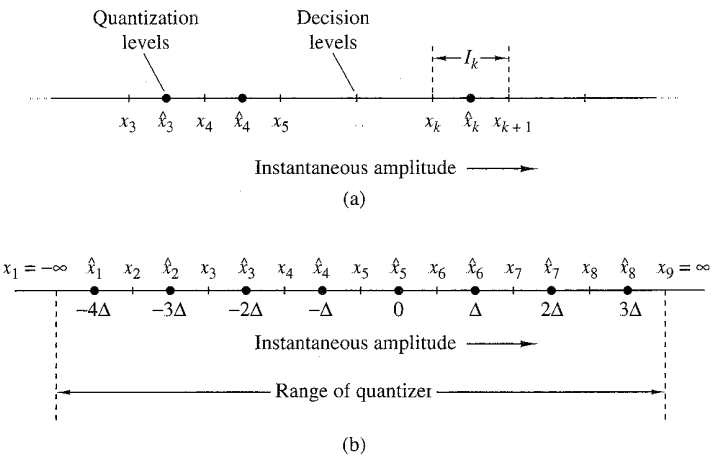



**Figure 6.3.2** Quantization process and an example of a midtread quantizer.

In most digital signal processing operations the mapping in (6.3.2) is independent of $n$ (i.e., the quantization is memoryless and is simply denoted as $x_q = Q[x]$). Furthermore, in signal processing we often use *uniform* or *linear quantizers* defined by

$$\begin{aligned} \hat{x}_{k+1} - \hat{x}_k &= \Delta, \qquad k = 1, 2, \ldots, L-1 \\ x_{k+1} - x_k &= \Delta, \qquad \text{for finite } x_k, x_{k+1} \end{aligned} \tag{6.3.3}$$

where $\Delta$ is the *quantizer step size*. Uniform quantization is usually a requirement if the resulting digital signal is to be processed by a digital system. However, in transmission and storage applications of signals such as speech, nonlinear and time-variant quantizers are frequently used.

If zero is assigned a quantization level, the quantizer is of the *midtread* type. If zero is assigned a decision level, the quantizer is called a *midrise* type. Figure 6.3.2(b) illustrates a midtread quantizer with $L = 8$ levels. In theory, the extreme decision levels are taken as $x_1 = -\infty$ and $x_{L+1} = \infty$, to cover the total *dynamic range* of the input signal. However, practical A/D converters can handle only a finite range. Hence we define the *range* $R$ of the quantizer by assuming that $I_1 = I_L = \Delta$. For example, the range of the quantizer shown in Fig. 6.3.2(b) is equal to $8\Delta$. In practice, the term *full-scale range* (FSR) is used to describe the range of an A/D converter for bipolar signals (i.e., signals with both positive and negative amplitudes). The term *full scale* (FS) is used for unipolar signals.

It can be easily seen that the quantization error $e_q(n)$ is always in the range $-\Delta/2$ to $\Delta/2$:

$$-\frac{\Delta}{2} < e_q(n) \leq \frac{\Delta}{2} \tag{6.3.4}$$

In other words, the instantaneous quantization error cannot exceed half of the quantization step. If the dynamic range of the signal, defined as $x_{\max} - x_{\min}$, is larger than the range of the quantizer, the samples that exceed the quantizer range are clipped, resulting in a large (greater than $\Delta/2$) quantization error.

The operation of the quantizer is better described by the quantization characteristic function, illustrated in Fig. 6.3.3 for a midtread quantizer with eight quantization levels. This characteristic is preferred in practice over the midriser because it provides an output that is insensitive to infinitesimal changes of the input signal about zero. Note that the input amplitudes of a midtread quantizer are rounded to the nearest quantization levels.

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 $b + 1$ bits we can represent $2^{b+1}$ distinct binary numbers. Hence we should have $2^{b+1} \geq L$ or, equivalently, $b + 1 \geq \log_2 L$. Then the step size or the *resolution* of the A/D converter is given by

$$\Delta = \frac{R}{2^{b+1}} \tag{6.3.5}$$

where $R$ is the range of the quantizer.

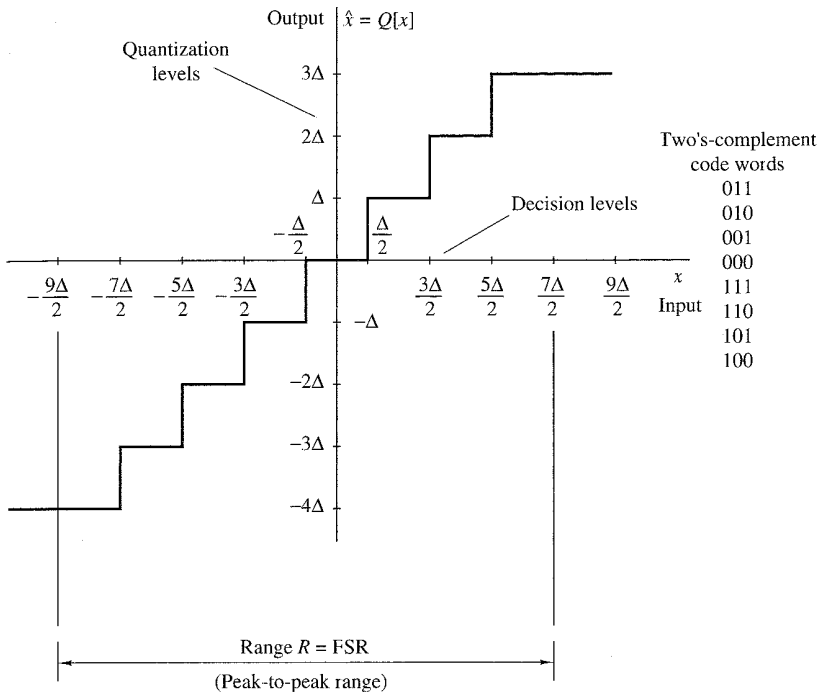



**Figure 6.3.3** Example of a midtread quantizer.

There are various binary coding schemes, each with its advantages and disadvantages. Table 6.1 illustrates some existing schemes for 3-bit binary coding. These number representation schemes are described in more detail in Section 9.4.

The two's-complement representation is used in most digital signal processors. Thus it is convenient to use the same system to represent digital signals because we can operate on them directly without any extra format conversion. In general, a $(b+1)$-bit binary fraction of the form $\beta_0\beta_1\beta_2\cdots\beta_b$ has the value

$$-\beta_0 \cdot 2^0 + \beta_1 \cdot 2^{-1} + \beta_2 \cdot 2^{-2} + \cdots + \beta_b \cdot 2^{-b}$$

if we use the two's-complement representation. Note that $\beta_0$ is the most significant bit (MSB) and $\beta_b$ is the least significant bit (LSB). Although the binary code used to represent the quantization levels is important for the design of the A/D converter and the subsequent numerical computations, it does not have any effect in the performance of the quantization process. Thus in our subsequent discussions we ignore the process of coding when we analyze the performance of A/D converters.

The only degradation introduced by an ideal converter is the quantization error, which can be reduced by increasing the number of bits. This error, which dominates the performance of practical A/D converters, is analyzed in the next section.

Practical A/D converters differ from ideal converters in several ways. Various degradations are usually encountered in practice. Specifically, practical A/D converters may have an *offset* error (the first transition may not occur at exactly $+\frac{1}{2}$LSB),

**TABLE 6.1** Commonly Used Bipolar Codes

| | Decimal Fraction | | | | | |
|---|---|---|---|---|---|---|
| Number | Positive Reference | Negative Reference | Sign + Magnitude | Two's Complement | Offset Binary | One's Complement |
| +7 | $+\frac{7}{8}$ | $-\frac{7}{8}$ | 0111 | 0111 | 1111 | 0111 |
| +6 | $+\frac{6}{8}$ | $-\frac{6}{8}$ | 0110 | 0110 | 1110 | 0110 |
| +5 | $+\frac{5}{8}$ | $-\frac{5}{8}$ | 0101 | 0101 | 1101 | 0101 |
| +4 | $+\frac{4}{8}$ | $-\frac{4}{8}$ | 0100 | 0100 | 1100 | 0100 |
| +3 | $+\frac{3}{8}$ | $-\frac{3}{8}$ | 0011 | 0011 | 1011 | 0011 |
| +2 | $+\frac{2}{8}$ | $-\frac{2}{8}$ | 0010 | 0010 | 1010 | 0010 |
| +1 | $+\frac{1}{8}$ | $-\frac{1}{8}$ | 0001 | 0001 | 1001 | 0001 |
| 0 | 0+ | 0− | 0000 | 0000 | 1000 | 0000 |
| 0 | 0− | 0+ | 1000 | (0000) | (1000) | 1111 |
| −1 | $-\frac{1}{8}$ | $+\frac{1}{8}$ | 1001 | 1111 | 0111 | 1110 |
| −2 | $-\frac{2}{8}$ | $+\frac{2}{8}$ | 1010 | 1110 | 0110 | 1101 |
| −3 | $-\frac{3}{8}$ | $+\frac{3}{8}$ | 1011 | 1101 | 0101 | 1100 |
| −4 | $-\frac{4}{8}$ | $+\frac{4}{8}$ | 1100 | 1100 | 0100 | 1011 |
| −5 | $-\frac{5}{8}$ | $+\frac{5}{8}$ | 1101 | 1011 | 0011 | 1010 |
| −6 | $-\frac{6}{8}$ | $+\frac{6}{8}$ | 1110 | 1010 | 0010 | 1001 |
| −7 | $-\frac{7}{8}$ | $+\frac{7}{8}$ | 1111 | 1001 | 0001 | 1000 |
| −8 | $-\frac{8}{8}$ | $+\frac{8}{8}$ | | (1000) | (0000) | |

*scale-factor* (or gain) error (the difference between the values at which the first transition and the last transition occur is not equal to FS − 2LSB), and a *linearity* error (the differences between transition values are not all equal or uniformly changing). If the *differential linearity* error is large enough, it is possible for one or more code words to be missed. Performance data on commercially available A/D converters are specified in manufacturers' data sheets.

### 6.3.3 Analysis of Quantization Errors

To determine the effects of quantization on the performance of an A/D converter, we adopt a statistical approach. The dependence of the quantization error on the characteristics of the input signal and the nonlinear nature of the quantizer make a deterministic analysis intractable, except in very simple cases.

In the statistical approach, we assume that the quantization error is random in nature. We model this error as noise that is added to the original (unquantized) signal. If the input analog signal is within the range of the quantizer, the quantization error

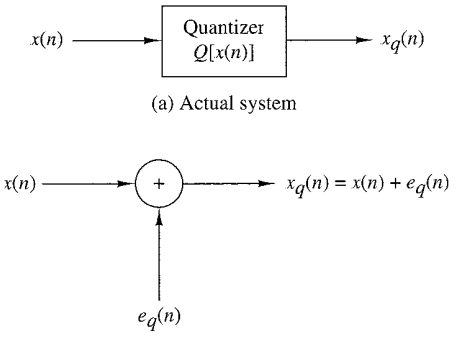



**Figure 6.3.4**
Mathematical model of quantization noise.

$e_q(n)$ is bounded in magnitude [i.e., $|e_q(n)| < \Delta/2$], and the resulting error is called *granular noise*. When the input falls outside the range of the quantizer (clipping), $e_q(n)$ becomes unbounded and results in *overload noise*. This type of noise can result in severe signal distortion when it occurs. Our only remedy is to scale the input signal so that its dynamic range falls within the range of the quantizer. The following analysis is based on the assumption that there is no overload noise.

The mathematical model for the quantization error $e_q(n)$ is shown in Fig. 6.3.4. To carry out the analysis, we make the following assumptions about the statistical properties of $e_q(n)$:

1. The error $e_q(n)$ is uniformly distributed over the range $-\Delta/2 < e_q(n) < \Delta/2$.
2. The error sequence $\{e_q(n)\}$ is a stationary white noise sequence. In other words, the error $e_q(n)$ and the error $e_q(m)$ for $m \neq n$ are uncorrelated.
3. The error sequence $\{e_q(n)\}$ is uncorrelated with the signal sequence $x(n)$.
4. The signal sequence $x(n)$ is zero mean and stationary.

These assumptions do not hold, in general. However, they do hold when the quantization step size is small and the signal sequence $x(n)$ traverses several quantization levels between two successive samples.

Under these assumptions, the effect of the additive noise $e_q(n)$ on the desired signal can be quantified by evaluating the signal-to-quantization-noise (power) ratio (SQNR), which can be expressed on a logarithmic scale (in decibels or dB) as

$$\text{SQNR} = 10 \log_{10} \frac{P_x}{P_n} \tag{6.3.6}$$

where $P_x = \sigma_x^2 = E[x^2(n)]$ is the signal power and $P_n = \sigma_e^2 = E[e_q^2(n)]$ is the power of the quantization noise.

If the quantization error is uniformly distributed in the range $(-\Delta/2, \Delta/2)$ as shown in Fig. 6.3.5, the mean value of the error is zero and the variance (the quantization noise power) is

$$P_n = \sigma_e^2 = \int_{-\Delta/2}^{\Delta/2} e^2 p(e)\, de = \frac{1}{\Delta} \int_{-\Delta/2}^{\Delta/2} e^2\, de = \frac{\Delta^2}{12} \tag{6.3.7}$$

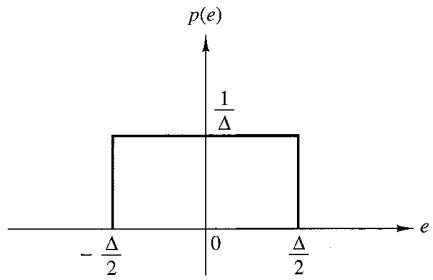



**Figure 6.3.5** Probability density function for the quantization error.

By combining (6.3.5) with (6.3.7) and substituting the result into (6.3.6), the expression for the SQNR becomes

$$\begin{aligned} \text{SQNR} &= 10\log\frac{P_x}{P_n} = 20\log\frac{\sigma_x}{\sigma_e} \\ &= 6.02b + 16.81 - 20\log\frac{R}{\sigma_x}\ \text{dB} \end{aligned} \tag{6.3.8}$$

The last term in (6.3.8) depends on the range $R$ of the A/D converter and the statistics of the input signal. For example, if we assume that $x(n)$ is Gaussian distributed and the range of the quantizer extends from $-3\sigma_x$ to $3\sigma_x$ (i.e., $R = 6\sigma_x$), then less than 3 out of every 1000 input signal amplitudes would result in an overload on the average. For $R = 6\sigma_x$, (6.3.8) becomes

$$\text{SQNR} = 6.02b + 1.25\ \text{dB} \tag{6.3.9}$$

The formula in (6.3.8) is frequently used to specify the precision needed in an A/D converter. It simply means that each additional bit in the quantizer increases the signal-to-quantization-noise ratio by 6 dB. (It is interesting to note that the same result was derived in Section 1.4 for a sinusoidal signal using a deterministic approach.) However, we should bear in mind the conditions under which this result has been derived.

Due to limitations in the fabrication of A/D converters, their performance falls short of the theoretical value given by (6.3.8). As a result, the effective number of bits may be somewhat less than the number of bits in the A/D converter. For instance, a 16-bit converter may have only an effective 14 bits of accuracy.

### 6.3.4 Digital-to-Analog Converters

In practice, D/A conversion is usually performed by combining a D/A converter with a sample-and-hold (S/H) followed by a lowpass (smoothing) filter, as shown in Fig. 6.3.6. The D/A converter accepts, at its input, electrical signals that correspond

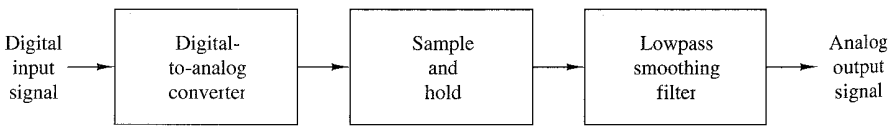



**Figure 6.3.6** Basic operations in converting a digital signal into an analog signal.

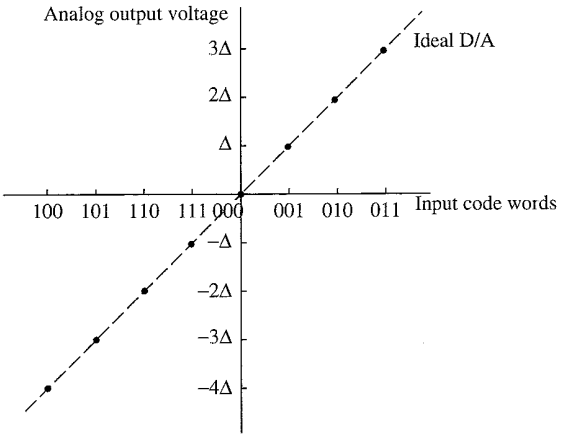



**Figure 6.3.7**
Ideal D/A converter characteristic.

to a binary word, and produces an output voltage or current that is proportional to the value of the binary word. Ideally, its input–output characteristic is as shown in Fig. 6.3.7 for a 3-bit bipolar signal. The line connecting the dots is a straight line through the origin. In practical D/A converters, the line connecting the dots may deviate from the ideal. Some of the typical deviations from ideal are offset errors, gain errors, and nonlinearities in the input–output characteristic.

An important parameter of a D/A converter is its *settling time*, which is defined as the time required for the output of the D/A converter to reach and remain within a given fraction (usually, $\pm\frac{1}{2}$LSB) of the final value, after application of the input code word. Often, the application of the input code word results in a high-amplitude transient, called a "glitch." This is especially the case when two consecutive code words to the A/D differ by several bits. The usual way to remedy this problem is to use an S/H circuit designed to serve as a "deglitcher." Hence the basic task of the S/H is to hold the output of the D/A converter constant at the previous output value until the new sample at the output of the D/A reaches steady state. Then it samples and holds the new value in the next sampling interval. Thus the S/H approximates the analog signal by a series of rectangular pulses whose height is equal to the corresponding value of the signal pulse. Figure 6.3.8 illustrates the response of an S/H to a discrete-time sinusoidal signal. As shown, the approximation, is basically a staircase function which takes the signal sample from the D/A converter and holds it for $T$ seconds. When the next sample arrives, it jumps to the next value and holds it for $T$ seconds, and so on.

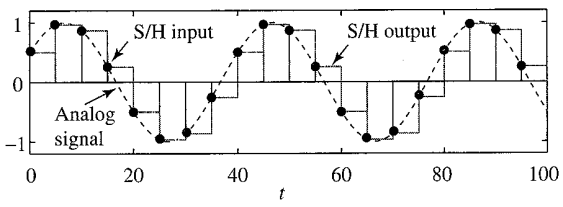



**Figure 6.3.8**
Response of an S/H interpolator to a discrete-time sinusoidal signal.

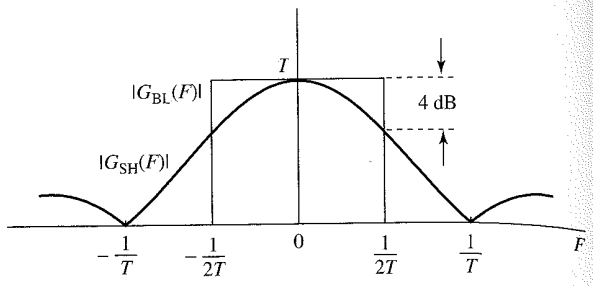



**Figure 6.3.9**
Frequency responses of sample-and-hold and the ideal bandlimited interpolator.

The interpolation function of the S/H system is a square pulse defined by

$$g_{SH}(t) = \begin{cases} 1, & 0 \le t \le T \\ 0, & \text{otherwise} \end{cases} \tag{6.3.10}$$

The frequency-domain characteristics are obtained by evaluating its Fourier transform

$$G_{SH}(F) = \int_{-\infty}^{\infty} g_{SH}(t) e^{-j2\pi Ft}\, dt = T \frac{\sin \pi FT}{\pi FT} e^{-2\pi F(T/2)} \tag{6.3.11}$$

The magnitude of $G_{SH}(F)$ is shown in Figure 6.3.9, where we superimpose the magnitude response of the ideal bandlimited interpolator for comparison purposes. It is apparent that the S/H does not possess a sharp cutoff frequency characteristic. This is due to a large extent to the sharp transitions of its interpolation function $g_{SH}(t)$. As a consequence, the S/H passes undesirable aliased frequency components (frequencies above $F_s/2$) to its output. This effect is sometimes referred to as *post-aliasing*. To remedy this problem, it is common practice to filter the output of the S/H by passing it through a lowpass filter, which highly attenuates frequency components above $F_s/2$. In effect, the lowpass filter following the S/H smooths its output by removing sharp discontinuities. Sometimes, the frequency response of the lowpass filter is defined by

$$H_a(F) = \begin{cases} \dfrac{\pi FT}{\sin \pi FT} e^{2\pi F(T/2)}, & |F| \le F_s/2 \\ 0, & |F| > F_s/2 \end{cases} \tag{6.3.12}$$

to compensate for the $\sin x/x$ distortion of the S/H (aperture effect). The aperture effect attenuation compensation, which reaches a maximum of $2/\pi$ or 4 dB at $F = F_s/2$, is usually neglected. However, it may be introduced using a digital filter before the sequence is applied to the D/A converter. The half-sample delay introduced by the S/H cannot be compensated because we cannot design analog filters that can introduce a time advance.

## 6.4 Sampling and Reconstruction of Continuous-Time Bandpass Signals

A continuous-time bandpass signal with bandwidth $B$ and center frequency $F_c$ has its frequency content in the two frequency bands defined by $0 < F_L < |F| < F_H$, where $F_c = (F_L + F_H)/2$ (see Figure 6.4.1(a)). A naive application of the sampling theorem

would suggest a sampling rate $F_s \geq 2F_H$; however, as we show in this section, there are sampling techniques that allow sampling rates consistent with the bandwidth $B$, rather than the highest frequency, $F_H$, of the signal spectrum. Sampling of bandpass signals is of great interest in the areas of digital communications, radar, and sonar systems.

### 6.4.1 Uniform or First-Order Sampling

Uniform or first-order sampling is the typical periodic sampling introduced in Section 6.1. Sampling the bandpass signal in Figure 6.4.1(a) at a rate $F_s = 1/T$ produces a sequence $x(n) = x_a(nT)$ with spectrum

$$X(F) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X_a(F - kF_s) \tag{6.4.1}$$

The positioning of the shifted replicas $X(F - kF_s)$ is controlled by a single parameter, the sampling frequency $F_s$. Since bandpass signals have two spectral bands, in general, it is more complicated to control their positioning, in order to avoid aliasing, with the single parameter $F_s$.

**Integer Band Positioning.** We initially restrict the higher frequency of the band to be an integer multiple of the bandwidth, that is, $F_H = mB$ (*integer band positioning*). The number $m = F_H/B$, which is in general fractional, is known as the *band position*. Figures 6.4.1(a) and 6.4.1(d) show two bandpass signals with even ($m = 4$) and odd ($m = 3$) band positioning. It can be easily seen from Figure 6.4.1(b) that, for integer-positioned bandpass signals, choosing $F_s = 2B$ results in a sequence with a spectrum without aliasing. From Figure 6.4.1(c), we see that the original bandpass signal can be reconstructed using the reconstruction formula

$$x_a(t) = \sum_{n=-\infty}^{\infty} x_a(nT) g_a(t - nT) \tag{6.4.2}$$

where

$$g_a(t) = \frac{\sin \pi Bt}{\pi Bt} \cos 2\pi F_c t \tag{6.4.3}$$

is the inverse Fourier transform of the bandpass frequency gating function shown in Figure 6.4.1(c). We note that $g_a(t)$ is equal to the ideal interpolation function for lowpass signals [see (6.1.21)], modulated by a carrier with frequency $F_c$.

It is worth noticing that, by properly choosing the center frequency $F_c$ of $G_a(F)$, we can reconstruct a continuous-time bandpass signal with spectral bands centered at $F_c = \pm(kB + B/2)$, $k = 0, 1, \ldots$. For $k = 0$ we obtain the equivalent baseband signal, a process known as *down-conversion*. A simple inspection of Figure 6.4.1 demonstrates that the baseband spectrum for $m = 3$ has the same spectral structure as the original spectrum; however, the baseband spectrum for $m = 4$ has been "inverted." In general, when the band position is an *even* integer the baseband spectral images are inverted versions of the original ones. Distinguishing between these two cases is important in communications applications.

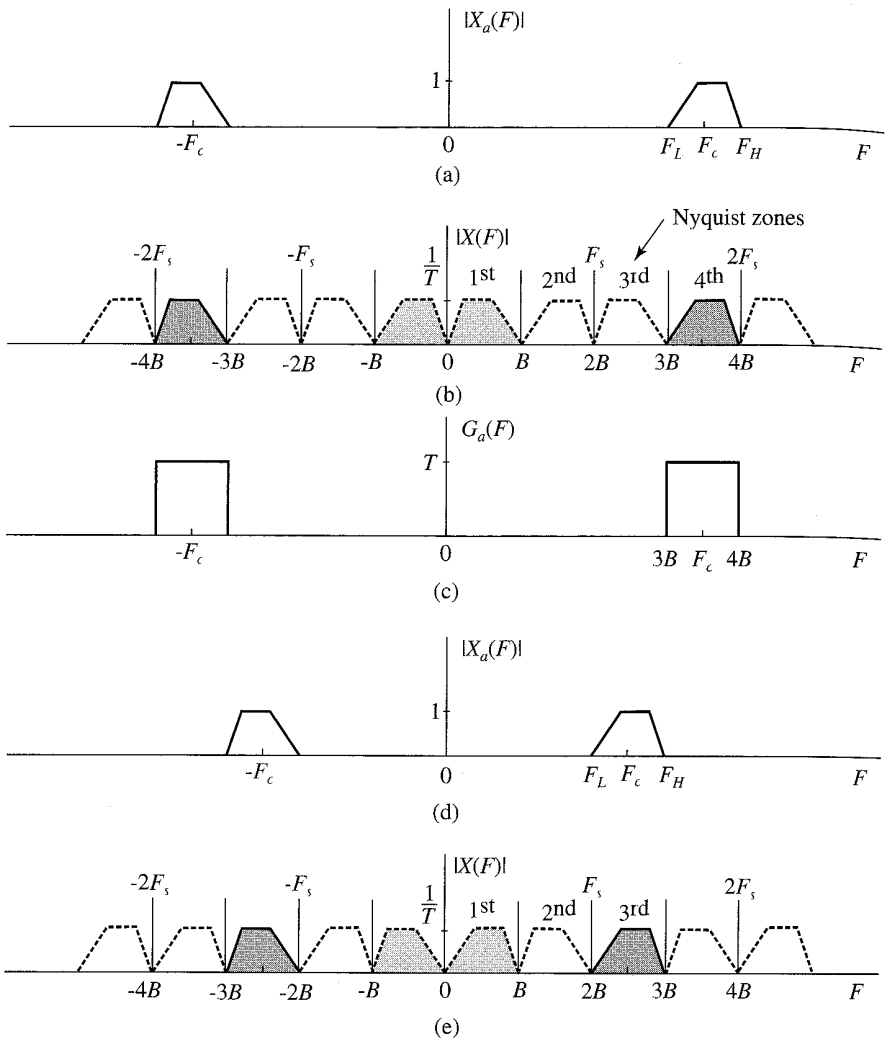



**Figure 6.4.1** Illustration of bandpass signal sampling for integer band positioning.

**Arbitrary Band Positioning.** Consider now a bandpass signal with arbitrarily positioned spectral bands, as shown in Figure 6.4.2. To avoid aliasing, the sampling frequency should be such that the $(k-1)$th and $k$th shifted replicas of the "negative" spectral band do not overlap with the "positive" spectral band. From Figure 6.4.2(b) we see that this is possible if there is an integer $k$ and a sampling frequency $F_s$ that satisfy the following conditions:

$$2F_H \leq kF_s \tag{6.4.4}$$

$$(k-1)F_s \leq 2F_L \tag{6.4.5}$$

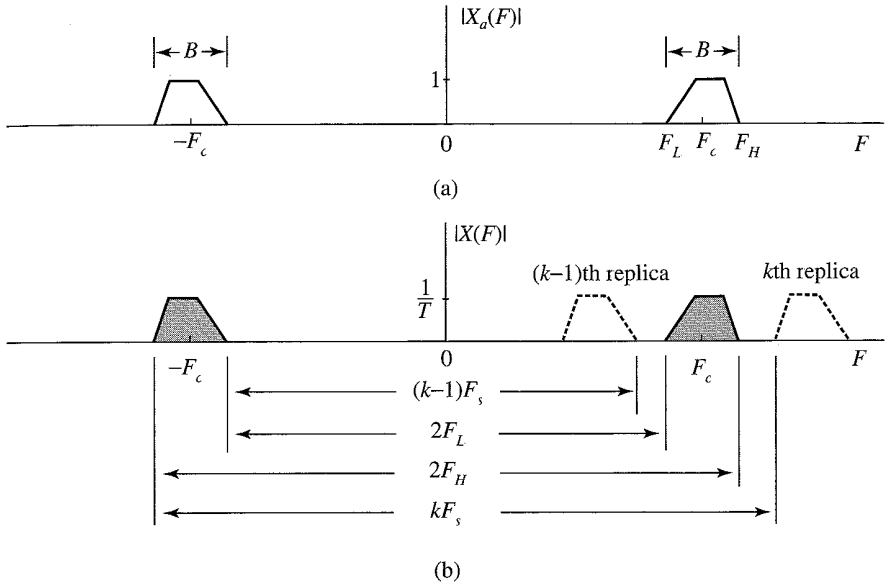



**Figure 6.4.2** Illustration of bandpass signal sampling for arbitrary band positioning

which is a system of two inequalities with two unknowns, $k$ and $F_s$. From (6.4.4) and (6.4.5) we can easily see that $F_s$ should be in the range

$$\frac{2F_H}{k} \leq F_s \leq \frac{2F_L}{k-1} \tag{6.4.6}$$

To determine the integer $k$ we rewrite (6.4.4) and (6.4.5) as follows:

$$\frac{1}{F_s} \leq \frac{k}{2F_H} \tag{6.4.7}$$

$$(k-1)F_s \leq 2F_H - 2B \tag{6.4.8}$$

By multiplying (6.4.7) and (6.4.8) by sides and solving the resulting inequality for $k$ we obtain

$$k_{\max} \leq \frac{F_H}{B} \tag{6.4.9}$$

The maximum value of integer $k$ is the number of bands that we can fit in the range from 0 to $F_H$, that is

$$k_{\max} = \left\lfloor \frac{F_H}{B} \right\rfloor \tag{6.4.10}$$

where $\lfloor b \rfloor$ denotes the integer part of $b$. The minimum sampling rate required to avoid aliasing is $F_{s\max} = 2F_H/k_{\max}$. Therefore, the range of acceptable uniform sampling rates is determined by

$$\frac{2F_H}{k} \leq F_s \leq \frac{2F_L}{k-1} \tag{6.4.11}$$

where $k$ is an integer number given by

$$1 \le k \le \left\lfloor \frac{F_H}{B} \right\rfloor \tag{6.4.12}$$

As long as there is no aliasing, reconstruction is done using (6.4.2) and (6.4.3), which are valid for both integer and arbitrary band positioning.

**Choosing a Sampling Frequency.** To appreciate the implications of conditions (6.4.11) and (6.4.12), we depict them graphically in Figure 6.4.3, as suggested by Vaughan et al. (1991). The plot shows the sampling frequency, normalized by $B$, as a function of the band position, $F_H/B$. This is facilitated by rewriting (6.4.11) as follows:

$$\frac{2}{k}\frac{F_H}{B} \le \frac{F_s}{B} \le \frac{2}{k-1}\left(\frac{F_H}{B} - 1\right) \tag{6.4.13}$$

The shaded areas represent sampling rates that result in aliasing. The allowed range of sampling frequencies is inside the white wedges. For $k = 1$, we obtain $2F_H \le F_s \le \infty$, which is the sampling theorem for lowpass signals. Each wedge in the plot corresponds to a different value of $k$.

To determine the allowed sampling frequencies, for a given $F_H$ and $B$, we draw a vertical line at the point determined by $F_H/B$. The segments of the line within the allowed areas represent permissible sampling rates. We note that the theoretically minimum sampling frequency $F_s = 2B$, corresponding to integer band positioning,

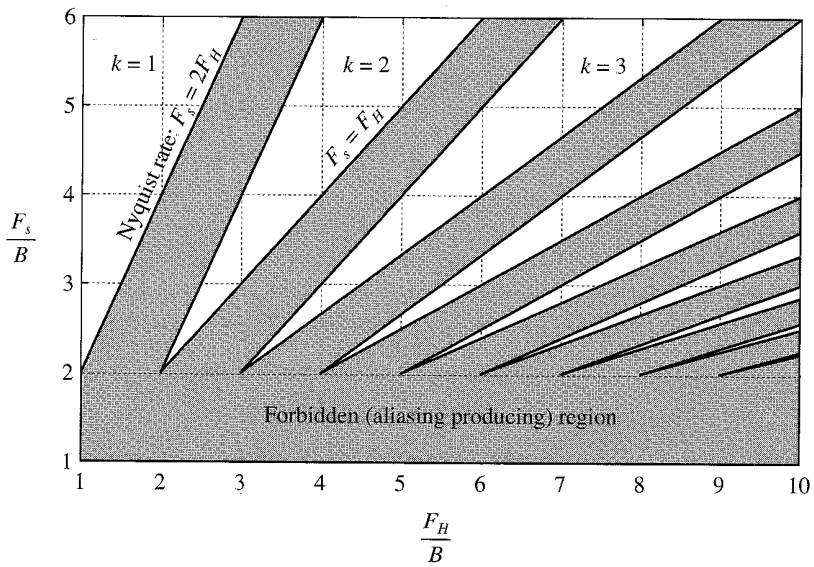



**Figure 6.4.3** Allowed (white) and forbidden (shaded) sampling frequency regions for bandpass signals. The minimum sampling frequency $F_s = 2B$, which corresponds to the corners of the alias-free wedges, is possible for integer-positioned bands only.

occurs at the tips of the wedges. Therefore, any small variation of the sampling rate or the carrier frequency of the signal will move the sampling frequency into the forbidden area. A practical solution is to sample at a higher sampling rate, which is equivalent to augmenting the signal band with a guard band $\Delta B = \Delta B_L + \Delta B_H$. The augmented band locations and bandwidth are given by

$$F'_L = F_L - \Delta B_L \tag{6.4.14}$$

$$F'_H = F_H + \Delta B_H \tag{6.4.15}$$

$$B' = B + \Delta B \tag{6.4.16}$$

The lower-order wedge and the corresponding range of allowed sampling are given by

$$\frac{2F'_H}{k'} \le F_s \le \frac{2F'_L}{k'-1} \quad \text{where} \quad k' = \left\lfloor \frac{F'_H}{B'} \right\rfloor \tag{6.4.17}$$

The $k'$th wedge with the guard bands and the sampling frequency tolerances are illustrated in Figure 6.4.4. The allowable range of sampling rates is divided into values above and below the practical operating points as

$$\Delta F_s = \frac{2F'_L}{k'-1} - \frac{2F'_H}{k'} = \Delta F_{sL} + \Delta F_{sH} \tag{6.4.18}$$

From the shaded orthogonal triangles in Figure 6.4.4, we obtain

$$\Delta B_L = \frac{k'-1}{2}\Delta F_{sH} \tag{6.4.19}$$

$$\Delta B_H = \frac{k'}{2}\Delta F_{sL} \tag{6.4.20}$$

which shows that symmetric guard bands lead to asymmetric sampling rate tolerance.

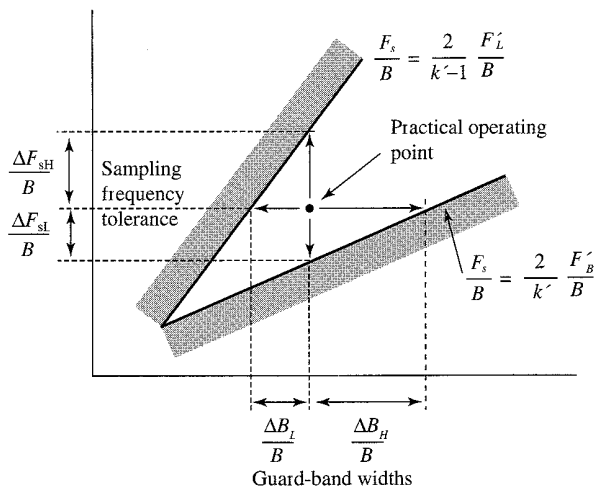



**Figure 6.4.4**
Illustration of the relationship between the size of guard bands and allowed sampling frequency deviations from its nominal value for the $k$th wedge.

If we choose the practical operating point at the vertical midpoint of the wedge, the sampling rate is

$$F_s = \frac{1}{2}\left(\frac{2F'_H}{k'} + \frac{2F'_L}{k'-1}\right) \tag{6.4.21}$$

Since, by construction, $\Delta F_{sL} = \Delta F_{sH} = \Delta F_s/2$, the guard bands become

$$\Delta B_L = \frac{k'-1}{4}\Delta F_s \tag{6.4.22}$$

$$\Delta B_H = \frac{k'}{4}\Delta F_s \tag{6.4.23}$$

We next provide an example that illustrates the use of this approach.

**EXAMPLE 6.4.1**

Suppose we are given a bandpass signal with $B = 25$ kHz and $F_L = 10{,}702.5$ kHz. From (6.4.10) the maximum wedge index is

$$k_{\max} = \lfloor F_H/B \rfloor = 429$$

This yields the theoretically minimum sampling frequency

$$F_s = \frac{2F_H}{k_{\max}} = 50.0117 \text{ kHz}$$

To avoid potential aliasing due to hardware imperfections, we wish to use two guard bands of $\Delta B_L = 2.5$ kHz and $\Delta B_H = 2.5$ kHz on each side of the signal band. The effective bandwidth of the signal becomes $B' = B + \Delta B_L + \Delta B_H = 30$ kHz. In addition, $F'_L = F_L - \Delta B_L = 10{,}700$ kHz and $F'_H = F_H + \Delta B_H = 10{,}730$ kHz. From (6.4.17), the maximum wedge index is

$$k'_{\max} = \lfloor F'_H/B' \rfloor = 357$$

Substitution of $k_{\max}$ into the inequality in (6.4.17) provides the range of acceptable sampling frequencies

$$60.1120 \text{ kHz} \le F_s \le 60.1124 \text{ kHz}$$

---

A detailed analysis on how to choose in practice the sampling rate for bandpass signals is provided by Vaughan et al. (1991) and Qi et al. (1996).

### 6.4.2 Interleaved or Nonuniform Second-Order Sampling

Suppose that we sample a continuous-time signal $x_a(t)$ with sampling rate $F_i = 1/T_i$ at time instants $t = nT_i + \Delta_i$, where $\Delta_i$ is a fixed time offset. Using the sequence

$$x_i(nT_i) = x_a(nT_i + \Delta_i), \quad -\infty < n < \infty \tag{6.4.24}$$

and a reconstruction function $g_a^{(i)}(t)$ we generate a continuous-time signal

$$y_a^{(i)}(t) = \sum_{n=-\infty}^{\infty} x_i(nT_i) g_a^{(i)}(t - nT_i - \Delta_i) \tag{6.4.25}$$

The Fourier transform of $y_a^{(i)}(t)$ is given by

$$Y_a^{(i)}(F) = \sum_{n=-\infty}^{\infty} x_i(nT_i) G_a^{(i)}(F) e^{-j2\pi F(nT_i+\Delta_i)} \tag{6.4.26}$$

$$= G_a^{(i)}(F) X_i(F) e^{-j2\pi F \Delta_i} \tag{6.4.27}$$

where $X_i(F)$ is the Fourier transform of $x_i(nT_i)$. From the sampling theorem (6.1.14), the Fourier transform of $x_i(nT_i)$ can be expressed in terms of the Fourier transform $X_a(F)e^{j2\pi F\Delta_i}$ of $x_a(t + \Delta_i)$ as

$$X_i(F) = \frac{1}{T_i} \sum_{k=-\infty}^{\infty} X_a\left(F - \frac{k}{T_i}\right) e^{j2\pi(F - \frac{k}{T_i})\Delta_i} \tag{6.4.28}$$

Substitution of (6.4.28) into (6.4.27) yields

$$Y_a^{(i)}(F) = G_a^{(i)}(F) \frac{1}{T_i} \sum_{k=-\infty}^{\infty} X_a\left(F - \frac{k}{T_i}\right) e^{-j2\pi \frac{k}{T_i}\Delta_i} \tag{6.4.29}$$

If we repeat the sampling process (6.4.24) for $i = 1, 2, \ldots, p$, we obtain $p$ interleaved uniformly sampled sequences $x_i(nT_i)$, $-\infty < n < \infty$. The sum of the $p$ reconstructed signals is given by

$$y_a(t) = \sum_{i=1}^{p} y_a^{(i)}(t) \tag{6.4.30}$$

Using (6.4.29) and (6.4.30), the Fourier transform of $y_a(t)$ can be expressed as

$$Y_a(F) = \sum_{i=1}^{p} G_a^{(i)}(F) V^{(i)}(F) \tag{6.4.31}$$

where

$$V^{(i)}(F) = \frac{1}{T_i} \sum_{k=-\infty}^{\infty} X_a\left(F - \frac{k}{T_i}\right) e^{-j2\pi \frac{k}{T_i}\Delta_i} \tag{6.4.32}$$

We will focus on the most commonly used second-order sampling, defined by

$$p = 2, \Delta_1 = 0, \Delta_2 = \Delta, T_1 = T_2 = \frac{1}{B} = T \tag{6.4.33}$$

In this case, which is illustrated in Figure 6.4.5, relations (6.4.31) and (6.4.32) yield

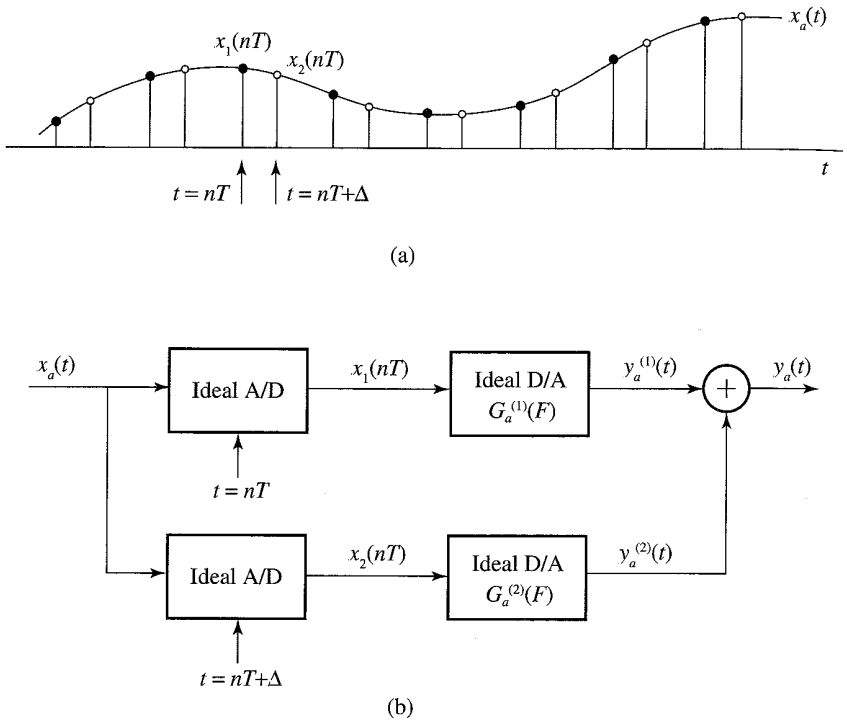



**Figure 6.4.5** Illustration of second-order bandpass sampling: (a) interleaved sampled sequences (b) second-order sampling and reconstruction system.

$$Y_a(F) = BG_a^{(1)}(F) \sum_{k=-\infty}^{\infty} X_a(F - kB) + BG_a^{(2)}(F) \sum_{k=-\infty}^{\infty} \gamma^k X_a(F - kB) \tag{6.4.34}$$

where

$$\gamma = e^{-j2\pi B\Delta} \tag{6.4.35}$$

To understand the nature of (6.4.34) we first split the spectrum $X_a(F)$ into a "positive" band and a "negative" band as follows:

$$X_a^+(F) = \begin{cases} X_a(F), & F \geq 0 \\ 0, & F < 0 \end{cases}, \quad X_a^-(F) = \begin{cases} X_a(F), & F \leq 0 \\ 0, & F > 0 \end{cases} \tag{6.4.36}$$

Then, we plot the repeated replicas of $X_a(F-kB)$ and $\gamma^k X_a(F-kB)$ as four separate components, as illustrated in Figure 6.4.6. We note that because each individual component has bandwidth $B$ and sampling rate $F_s = 1/B$, its repeated copies fill the entire frequency axis without overlapping, that is, without aliasing. However, when we combine them, the negative bands cause aliasing to the positive bands, and vice versa.

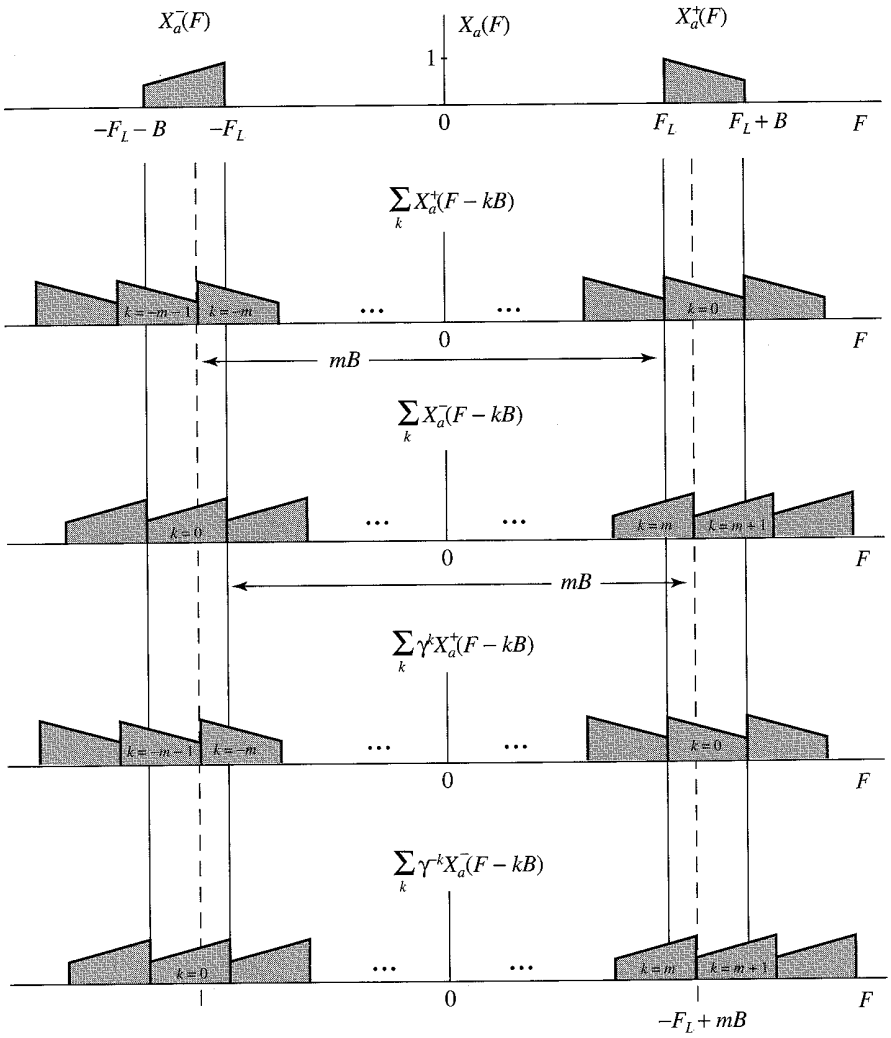



**Figure 6.4.6** Illustration of aliasing in second-order bandpass sampling.

We want to determine the interpolation functions $G_a^{(1)}(F)$, $G_a^{(2)}(F)$, and the time offset $\Delta$ so that $Y_a(F) = X_a(F)$. From Figure 6.4.6 we see that the first requirement is

$$G_a^{(1)}(F) = G_a^{(2)}(F) = 0, \quad \text{for } |F| < F_L \text{ and } |F| > F_L + B \tag{6.4.37}$$

To determine $G_a^{(1)}(F)$ and $G_a^{(2)}(F)$ for $F_L \le |F| \le F_L + B$, we see from Figure 6.4.6 that only the components with $k = \pm m$ and $k = \pm(m+1)$, where

$$m = \left\lceil \frac{2F_L}{B} \right\rceil \tag{6.4.38}$$

is the smallest integer greater or equal to $2F_L/B$, overlap with the original spectrum.

In the region $F_L \le F \le -F_L + mB$, equation (6.4.34) becomes

$$\begin{aligned} Y_a^+(F) &= \left[BG_a^{(1)}(F) + BG_a^{(2)}(F)\right] X_a^+(F) && \text{(Signal component)} \\ &+ \left[BG_a^{(1)}(F) + B\gamma^m G_a^{(2)}(F)\right] X_a^+(F - mB) && \text{(Aliasing component)} \end{aligned}$$

The conditions that assure perfect reconstruction $Y_a^+(F) = X_a^+(F)$ are given by

$$BG_a^{(1)}(F) + BG_a^{(2)}(F) = 1 \tag{6.4.39}$$

$$BG_a^{(1)}(F) + B\gamma^m G_a^{(2)}(F) = 0 \tag{6.4.40}$$

Solving this system of equations yields the solution

$$G_a^{(1)}(F) = \frac{1}{B}\frac{1}{1-\gamma^{-m}}, \quad G_a^{(2)}(F) = \frac{1}{B}\frac{1}{1-\gamma^{m}} \tag{6.4.41}$$

which exists for all $\Delta$ such that $\gamma^{\pm m} = e^{\mp j2\pi mB\Delta} \neq 1$.

In the region $-F_L + mB \le F \le F_L + B$, equation (6.4.34) becomes

$$\begin{aligned} Y_a^+(F) &= \left[BG_a^{(1)}(F) + BG_a^{(2)}(F)\right] X_a^+(F) \\ &+ \left[BG_a^{(1)}(F) + B\gamma^{m+1} G_a^{(2)}(F)\right] X_a^+(F - (m+1)B) \end{aligned}$$

The conditions that assure perfect reconstruction $Y_a^+(F) = X_a^+(F)$ are given by

$$BG_a^{(1)}(F) + BG_a^{(2)}(F) = 1 \tag{6.4.42}$$

$$BG_a^{(1)}(F) + B\gamma^{m+1} G_a^{(2)}(F) = 0 \tag{6.4.43}$$

Solving this system of equations yields the solution

$$G_a^{(1)}(F) = \frac{1}{B}\frac{1}{1-\gamma^{-(m+1)}}, \quad G_a^{(2)}(F) = \frac{1}{B}\frac{1}{1-\gamma^{m+1}} \tag{6.4.44}$$

which exists for all $\Delta$ such that $\gamma^{\pm(m+1)} = e^{\mp j2\pi(m+1)B\Delta} \neq 1$.

The reconstruction functions in the frequency range $-(F_L + B) \leq F \leq -F_L$ can be obtained in a similar manner. The formulas are given by (6.4.41) and (6.4.44) if we replace $m$ by $-m$ and $m+1$ by $-(m+1)$. The function $G_a^{(1)}(F)$ has the bandpass response shown in Figure 6.4.7. A similar plot for $G_a^{(2)}(F)$ reveals that

$$G_a^{(2)}(F) = G_a^{(1)}(-F) \tag{6.4.45}$$

which implies that $g_a^{(2)}(t) = g_a^{(1)}(-t)$. Therefore, for simplicity, we adopt the notation $g_a(t) = g_a^{(1)}(t)$ and express the reconstruction formula (6.4.30) as follows

$$x_a(t) = \sum_{n=-\infty}^{\infty} x_a\left(\frac{n}{B}\right) g_a\left(t - \frac{n}{B}\right) + x_a\left(\frac{n}{B} + \Delta\right) g_a\left(-t + \frac{n}{B} + \Delta\right) \tag{6.4.46}$$

Taking the inverse Fourier transform of the function shown in Figure 6.4.7, we can show (see Problem 6.7) that the interpolation function is given by

$$g_a(t) = a(t) + b(t) \tag{6.4.47}$$

$$a(t) = \frac{\cos[2\pi(mB - F_L)t - \pi m B\Delta] - \cos(2\pi F_L t - \pi m B\Delta)}{2\pi B t \sin(\pi m B\Delta)} \tag{6.4.48}$$

$$b(t) = \frac{\cos[2\pi(F_L + B)t - \pi(m+1)B\Delta] - \cos[2\pi(mB - F_L)t - \pi(m+1)B\Delta]}{2\pi B t \sin[\pi(m+1)B\Delta]} \tag{6.4.49}$$

We can see that $g_a(0) = 1$, $g_a(n/B) = 0$ for $n \neq 0$, and $g_a(n/B \pm \Delta) = 0$ for $n = 0, \pm 1, \pm 2, \ldots$, as expected for any interpolation function.

We have shown that a bandpass signal $x_a(t)$ with frequencies in the range $F_L \leq |F| \leq F_L + B$ can be perfectly reconstructed from two interleaved uniformly sampled sequences $x_a(n/B)$ and $x_a(n/B + \Delta)$, $-\infty < n < \infty$, using the interpolation formula (6.4.46) with an average rate $F_s = 2B$ samples/second without any restrictions on the band location. The time offset $\Delta$ cannot take values that may cause the interpolation function to take infinite values. This second-order sampling theorem was introduced by Kohlenberg (1953). The general $p$th-order sampling case ($p > 2$) is discussed by Coulson (1995).

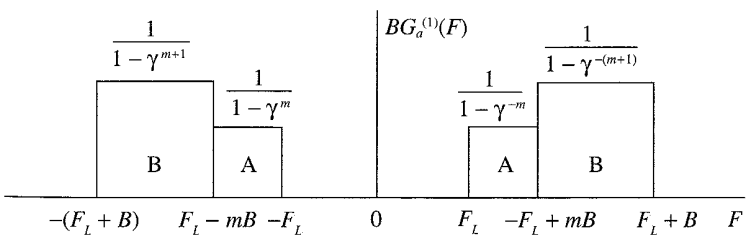



**Figure 6.4.7** Frequency domain characterization of the bandpass interpolation function for second-order sampling.

Some useful simplifications occur when $m = 2F_L/B$, that is, for integer band positioning (Linden 1959, Vaughan et al. 1991). In this case, the region A becomes zero, which implies that $a(t) = 0$. Therefore, we have $g_a(t) = b(t)$. There are two cases of special interest.

For low pass signals, $F_L = 0$ and $m = 0$, and the interpolation function becomes

$$g_{\mathrm{LP}}(t) = \frac{\cos(2\pi Bt - \pi B\Delta) - \cos(\pi B\Delta)}{2\pi Bt \sin(\pi B\Delta)} \tag{6.4.50}$$

The additional constraint $\Delta = 1/2B$, which results in uniform sampling rate, yields the well-known sine interpolation function $g_{\mathrm{LP}}(t) = \sin(2\pi Bt)/2\pi Bt$.

For bandpass signals with $F_L = mB/2$ we can choose the time offset $\Delta$ such that $\gamma^{\pm(m+1)} = -1$. This requirement is satisfied if

$$\Delta = \frac{2k+1}{2B(m+1)} = \frac{1}{4F_c} + \frac{k}{2F_c}, \quad k = 0, \pm 1, \pm 2, \ldots \tag{6.4.51}$$

where $F_c = F_L + B/2 = B(m+1)/2$ is the center frequency of the band. In this case, the interpolation function is specified by $G_Q(F) = 1/2$ in the range $mB/2 \le |F| \le (m+1)B/2$ and $G_Q(F) = 0$ elsewhere. Taking the inverse Fourier transform, we obtain

$$g_Q(t) = \frac{\sin \pi Bt}{\pi Bt} \cos 2\pi F_c t \tag{6.4.52}$$

which is a special case of (6.4.47)–(6.4.49). This special case is known as direct quadrature sampling because the in-phase and quadrature components are obtained explicitly from the bandpass signal (see Section 6.4.3).

Finally, we note that it is possible to sample a bandpass signal, and then to reconstruct the discrete-time signal at a band position other than the original. This spectral relocation or frequency shifting of the bandpass signal is most commonly done using direct quadrature sampling [Coulson et al. (1994)]. The significance of this approach is that it can be implemented using digital signal processing.

### 6.4.3 Bandpass Signal Representations

The main cause of complications in the sampling of a real bandpass signal $x_a(t)$ is the presence of two separate spectral bands in the frequency regions $-(F_L + B) \le F \le -F_L$ and $F_L \le F \le F_L + B$. Since $x_a(t)$ is real, the negative and positive frequencies in its spectrum are related by

$$X_a(-F) = X_a^*(F) \tag{6.4.53}$$

Therefore, the signal can be completely specified by one half of the spectrum. We next exploit this idea to introduce simplified representations for bandpass signals. We start with the identity

$$\cos 2\pi F_c t = \frac{1}{2} e^{j2\pi F_c t} + \frac{1}{2} e^{-j2\pi F_c t} \tag{6.4.54}$$

which represents the real signal $\cos 2\pi F_c t$ by two spectral lines of magnitude 1/2, one at $F = F_c$ and the other at $F = -F_c$. Equivalently, we have the identity

$$\cos 2\pi F_c t = 2\Re\left\{\frac{1}{2}e^{j2\pi F_c t}\right\} \tag{6.4.55}$$

which represents the real signal as the real part of a complex signal. In terms of the spectrum, we now specify the real signal $\cos 2\pi F_c t$ by the positive part of its spectrum, that is, the spectral line at $F = F_c$. The amplitude of the positive frequencies is doubled to compensate for the omission of the negative frequencies.

The extension to signals with continuous spectra is straightforward. Indeed, the integral of the inverse Fourier transform of $x_a(t)$ can be split into two parts as

$$x_a(t) = \int_0^{\infty} X_a(F)e^{j2\pi Ft}dF + \int_{-\infty}^{0} X_a(F)e^{j2\pi Ft}\,dF \tag{6.4.56}$$

Changing the variable in the second integral from $F$ to $-F$ and using (6.4.53) yields

$$x_a(t) = \int_0^{\infty} X_a(F)e^{j2\pi Ft}\,dF + \int_0^{\infty} X_a^*(F)e^{-j2\pi Ft}\,dF \tag{6.4.57}$$

The last equation can be equivalently written as

$$x_a(t) = \Re\left\{\int_0^{\infty} 2X_a(F)e^{j2\pi Ft}\,dF\right\} = \Re\{\psi_a(t)\} \tag{6.4.58}$$

where the complex signal

$$\psi_a(t) = \int_0^{\infty} 2X_a(F)e^{j2\pi Ft}\,dF \tag{6.4.59}$$

is known as the *analytic signal* or the *pre-envelope* of $x_a(t)$. The spectrum of the analytic signal can be expressed in terms of the unit step function $V_a(F)$ as follows:

$$\Psi_a(F) = 2X_a(F)V_a(F) = \begin{cases} 2X_a(F), & F > 0 \\ 0, & F < 0 \end{cases} \tag{6.4.60}$$

In case $X_a(0) \neq 0$, we define $\Psi_a(0) = X_a(0)$. To express the analytic signal $\psi_a(t)$ in terms of $x_a(t)$, we recall that the inverse Fourier transform of $V_a(F)$ is given by

$$v_a(t) = \frac{1}{2}\delta(t) + \frac{j}{2\pi t} \tag{6.4.61}$$

From (6.4.60), (6.4.61), and the frequency-domain convolution theorem, we obtain

$$\psi_a(t) = 2x_a(t) * v_a(t) = x_a(t) + j\frac{1}{\pi t} * x_a(t) \tag{6.4.62}$$

The signal obtained from the convolution of the impulse response

$$h_Q(t) = \frac{1}{\pi t} \tag{6.4.63}$$

and the input signal $x_a(t)$ is given by

$$\hat{x}_a(t) = \frac{1}{\pi t} * x_a(t) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x_a(\tau)}{t - \tau}\, d\tau \tag{6.4.64}$$

and it is called the *Hilbert transform* of $x_a(t)$, denoted by $\hat{x}_a(t)$. We emphasize that the Hilbert transform is a convolution and does not change the domain, so both its input $x_a(t)$ and its output $\hat{x}_a(t)$ are functions of time.

The filter defined by (6.4.63) has the frequency response function given by

$$H_Q(F) = \int_{-\infty}^{\infty} h_Q(t) e^{-j2\pi F t}\, dt = \begin{cases} -j, & F > 0 \\ j, & F < 0 \end{cases} \tag{6.4.65}$$

or in terms of magnitude and phase

$$|H_Q(F)| = 1, \quad \measuredangle H_Q(F) = \begin{cases} -\pi/2, & F > 0 \\ \pi/2, & F < 0 \end{cases} \tag{6.4.66}$$

The Hilbert transformer $H_Q(F)$ is an allpass quadrature filter that simply shifts the phase of positive frequency components by $-\pi/2$ and the phase of negative frequency components by $\pi/2$. From (6.4.63) we see that $h_Q(t)$ is noncausal, which means that the Hilbert transformer is physically unrealizable.

We can now express the analytic signal using the Hilbert transform as

$$\psi_a(t) = x_a(t) + j\hat{x}_a(t) \tag{6.4.67}$$

We see that the Hilbert transform of $x_a(t)$ provides the imaginary part of its analytic signal representation.

The analytic signal $\psi_a(t)$ of $x_a(t)$ is bandpass in the region $F_L \le F \le F_L + B$. Therefore, it can be shifted to the baseband region $-B/2 \le F \le B/2$ using the modulation property of the Fourier transform

$$x_{\mathrm{LP}}(t) = e^{-j2\pi F_c t} \psi_a(t) \overset{\mathcal{F}}{\longleftrightarrow} X_{\mathrm{LP}}(F) = \Psi_a(F + F_c) \tag{6.4.68}$$

The complex lowpass signal $x_{\mathrm{LP}}(t)$ is known as the *complex envelope* of $x_a(t)$.

The complex envelope can be expressed in rectangular coordinates as

$$x_{\mathrm{LP}}(t) = x_I(t) + j x_Q(t) \tag{6.4.69}$$

where $x_I(t)$ and $x_Q(t)$ are both real-valued lowpass signals in the same frequency region with $x_{\mathrm{LP}}(t)$. From (6.4.58), (6.4.68), and (6.4.69) we can easily deduce the so-called *quadrature representation* of bandpass signals

$$x_a(t) = x_I(t) \cos 2\pi F_c t - x_Q(t) \sin 2\pi F_c t \tag{6.4.70}$$

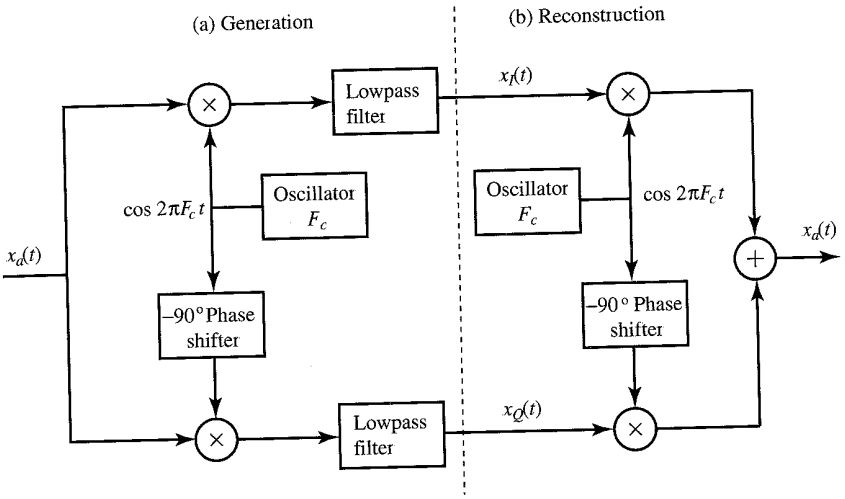



**Figure 6.4.8** (a) Scheme for generating the in-phase and quadrature components of a bandpass signal. (b) Scheme for reconstructing a bandpass signal from its in-phase and quadrature components.

We refer to $x_I(t)$ as the *in-phase component* of the bandpass signal and $x_Q(t)$ as the *quadrature component* because the carriers $\cos 2\pi F_c t$ and $\sin 2\pi F_c t$ are in-phase quadrature with respect to each other. The in-phase and quadrature components can be obtained from the signal $x_a(t)$ using quadrature demodulation as shown in Figure 6.4.8(a). The bandpass signal can be reconstructed using the scheme in 6.4.8(b).

Alternatively, we can express the complex envelope in polar coordinates as

$$x_{\mathrm{LP}}(t) = A(t)e^{j\phi(t)} \tag{6.4.71}$$

where $A(t)$ and $\phi(t)$ are both real-valued lowpass signals. In terms of this polar representation, the bandpass signal $x_a(t)$ can be written as

$$x_a(t) = A(t)\cos[2\pi F_c t + \phi(t)] \tag{6.4.72}$$

where $A(t)$ is known as the *envelope* and $\phi(t)$ as the *phase* of the bandpass signal. Equation (6.4.72) represents a bandpass signal using a combination of amplitude and angle modulation. We can easily see that $x_I(t)$ and $x_Q(t)$ are related to $A(t)$ and $\phi(t)$ as follows:

$$x_I(t) = A(t)\cos 2\pi F_c t, \qquad x_Q(t) = A(t)\sin 2\pi F_c t \tag{6.4.73}$$

$$A(t) = \sqrt{x_I^2(t) + x_Q^2(t)}, \qquad \phi(t) = \tan^{-1}\left[\frac{x_Q(t)}{x_I(t)}\right] \tag{6.4.74}$$

The phase $\phi(t)$ is uniquely defined in terms of $x_I(t)$ and $x_Q(t)$, modulo $2\pi$.

### 6.4.4 Sampling Using Bandpass Signal Representations

The complex envelope (6.4.68) and quadrature representations (6.4.70) allow the sampling of a bandpass signal at a rate $F_s = 2B$ independently of the band location.

Since the analytic signal $\psi_a(t)$ has a single band at $F_L \le F \le F_L + B$, it can be sampled at a rate of $B$ complex samples per second or $2B$ real samples per second without aliasing (see second graph in Figure 6.4.6). According to (6.4.67) these samples can be obtained by sampling $x_a(t)$ and its Hilbert transform $\hat{x}_a(t)$ at a rate of $B$ samples per second. Reconstruction requires a complex bandpass interpolation function defined by

$$g_a(t) = \frac{\sin \pi Bt}{\pi Bt} e^{j2\pi F_c t} \overset{\mathcal{F}}{\longleftrightarrow} G_a(F) = \begin{cases} 1, & F_L \le F \le F_L + B \\ 0, & \text{otherwise} \end{cases} \tag{6.4.75}$$

where $F_c = F_L + B/2$. The major problem with this approach is the design of practical analog Hilbert transformers.

Similarly, since the in-phase $x_I(t)$ and quadrature $x_Q(t)$ components of the bandpass signal $x_a(t)$ are lowpass signals with one-sided bandwidth $B/2$, they can be uniquely represented by the sequences $x_I(nT)$ and $x_Q(nT)$, where $T = 1/B$. This results in a total sampling rate of $F_s = 2B$ real samples per second. The original bandpass signal can be reconstructed by first reconstructing the in-phase and quadrature components using ideal interpolation and then recombining them using formula (6.4.70).

The in-phase and quadrature components can be obtained by directly sampling the signal $x_a(t)$, using second-order sampling with a proper choice of $\Delta$. This leads to a major simplification because we can avoid the complex demodulation process required to generate the in-phase and quadrature signals. To extract directly $x_I(t)$ from $x_a(t)$, that is

$$x_a(t_n) = x_I(t_n) \tag{6.4.76}$$

requires sampling at time instants

$$2\pi F_c t_n = \pi n, \text{ or } t_n = \frac{n}{2F_c}, n = 0, \pm 1, \pm 2, \ldots \tag{6.4.77}$$

Similarly, to obtain $x_Q(t)$, we should sample at time instants

$$2\pi F_c t_n = \frac{\pi}{2}(2n+1), \text{ or } t_n = \frac{2n+1}{4F_c}, n = 0, \pm 1, \pm 2, \ldots \tag{6.4.78}$$

which yields

$$x_a(t_n) = -x_Q(t_n) \tag{6.4.79}$$

which is equivalent to the special case second-order sampling defined by (6.4.51). Several variations of this approach are described by Grace and Pitt (1969), Rice and Wu (1982), Waters and Jarret (1982), and Jackson and Matthewson (1986).

Finally, we note that the quadrature approach to bandpass sampling has been widely used in radar and communications systems to generate in-phase and quadrature sequences for further processing. However, with the development of faster A/D

converters and digital signal processors it is more convenient and economic to sample directly the bandpass signal, as described in Section 6.4.1, and then obtain $x_I(n)$ and $x_Q(n)$ using the discrete-time approach developed in Section 6.5.

## 6.5 Sampling of Discrete-Time Signals

In this section we use the techniques developed for the sampling and representation of continuous-time signals to discuss the sampling and reconstruction of lowpass and bandpass discrete-time signals. Our approach is to conceptually reconstruct the underlying continuous-time signal and then resample at the desired sampling rate. However, the final implementations involve only discrete-time operations. The more general area of sampling rate conversion is the subject of Chapter 11.

### 6.5.1 Sampling and Interpolation of Discrete-Time Signals

Suppose that a sequence $x(n)$ is sampled periodically by keeping every $D$th sample of $x(n)$ and deleting the $(D-1)$ samples in between. This operation, which is also known as decimation or down-sampling, yields a new sequence defined by

$$x_d(n) = x(nD), \quad -\infty < n < \infty \tag{6.5.1}$$

Without loss of generality we assume that $x(n)$ has been obtained by sampling a signal $x_a(t)$ with spectrum $X_a(F) = 0$, $|F| > B$ at a sampling rate $F_s = 1/T \geq 2B$, that is, $x(n) = x_a(nT)$. Therefore, the spectrum $X(F)$ of $x(n)$ is given by

$$X(F) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X_a(F - kF_s) \tag{6.5.2}$$

We next sample $x_a(t)$ at time instants $t = nDT$, that is, with a sampling rate $F_s/D$. The spectrum of the sequence $x_d(n) = x_a(nDT)$ is provided by

$$X_d(F) = \frac{1}{DT} \sum_{k=-\infty}^{\infty} X_a\left(F - k\frac{F_s}{D}\right) \tag{6.5.3}$$

This process is illustrated in Figure 6.5.1 for $D = 2$ and $D = 4$. We can easily see from Figure 6.5.1(c) that the spectrum $X_d(F)$ can be expressed in terms of the periodic spectrum $X(F)$ as

$$X_d(F) = \frac{1}{D} \sum_{k=0}^{D-1} X\left(F - k\frac{F_s}{D}\right) \tag{6.5.4}$$

To avoid aliasing, the sampling rate should satisfy the condition $F_s/D \geq 2B$. If the sampling frequency $F_s$ is fixed, we can avoid aliasing by reducing the bandwidth of $x(n)$ to $(F_s/2)/D$. In terms of the normalized frequency variables, we can avoid aliasing if the highest frequency $f_{\max}$ or $\omega_{\max}$ in $x(n)$ satisfies the conditions

$$f_{\max} \leq \frac{1}{2D} = \frac{f_s}{2} \quad \text{or} \quad \omega_{\max} \leq \frac{\pi}{D} = \frac{\omega_s}{2} \tag{6.5.5}$$

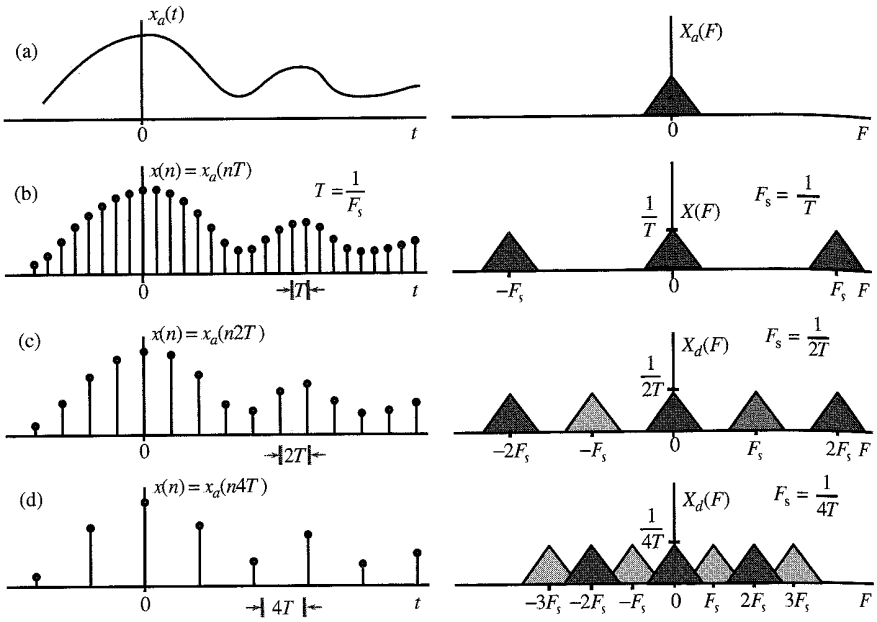



**Figure 6.5.1** Illustration of discrete-time signal sampling in the frequency domain.

In continuous-time sampling the continuous-time spectrum $X_a(F)$ is repeated an infinite number of times to create a periodic spectrum covering the infinite frequency range. In discrete-time sampling the periodic spectrum $X(F)$ is repeated $D$ times to cover one period of the periodic frequency domain.

To reconstruct the original sequence $x(n)$ from the sampled sequence $x_d(n)$, we start with the ideal interpolation formula

$$x_a(t) = \sum_{m=-\infty}^{\infty} x_d(m) \frac{\sin \frac{\pi}{DT}(t - mDT)}{\frac{\pi}{DT}(t - mDT)} \tag{6.5.6}$$

which reconstructs $x_a(t)$ assuming that $F_s/D \geq 2B$. Since $x(n) = x_a(nT)$, substitution into (6.5.6) yields

$$x(n) = \sum_{m=-\infty}^{\infty} x_d(m) \frac{\sin \frac{\pi}{D}(n - mD)}{\frac{\pi}{D}(n - mD)} \tag{6.5.7}$$

This is not a practical interpolator, since the $\sin(x)/x$ function is infinite in extent. In practice, we use a finite summation from $m = -L$ to $m = L$. The quality of this approximation improves with increasing $L$. The Fourier transform of the ideal bandlimited interpolation sequence in (6.5.7) is

$$g_{\mathrm{BL}}(n) = D\frac{\sin(\pi/D)n}{\pi n} \overset{\mathcal{F}}{\longleftrightarrow} G_{\mathrm{BL}}(\omega) = \begin{cases} D, & |\omega| \leq \pi/D \\ 0, & \pi/D < |\omega| \leq \pi \end{cases} \tag{6.5.8}$$

Therefore, the ideal discrete-time interpolator has an ideal lowpass frequency characteristic.

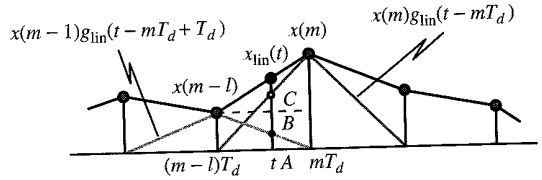



**Figure 6.5.2** Illustration of continuous-time linear interpolation.

To understand the process of discrete-time interpolation, we will analyze the widely used linear interpolation. For simplicity we use the notation $T_d = DT$ for the sampling period of $x_d(m) = x_a(mT_d)$. The value of $x_a(t)$ at a time instant between $mT_d$ and $(m+1)T_d$ is obtained by raising a vertical line from $t$ to the line segment connecting the samples $x_d(mT_d)$ and $x_d(mT_d + T_d)$, as shown in Figure 6.5.2. The interpolated value is given by

$$x_{\text{lin}}(t) = x(m-1) + \frac{x(m) - x(m-1)}{T_d}[t - (m-1)T_d], \; (m-1)T_d \le t \le mT_d \quad (6.5.9)$$

which can be rearranged as follows:

$$x_{\text{lin}}(t) = \left[1 - \frac{t - (m-1)T_d}{T_d}\right] x(m-1) + \left[1 - \frac{mT_d - t)}{T_d}\right] x(m) \quad (6.5.10)$$

To put (6.5.10) in the form of the general reconstruction formula

$$x_{\text{lin}}(t) = \sum_{m=-\infty}^{\infty} x(m) g_{\text{lin}}(t - mT_d) \quad (6.5.11)$$

we note that we always have $t - (m-1)T_d = |t - (m-1)T_d|$ and $mT_d - t = |t - mT_d|$ because $(m-1)T_d \le t \le mT_d$. Therefore, we can express (6.5.10) in the form (6.5.11) if we define

$$g_{\text{lin}}(t) = \begin{cases} 1 - \frac{|t|}{T_d}, & |t| \le T_d \\ 0, & |t| > T_d \end{cases} \quad (6.5.12)$$

The discrete-time interpolation formulas are obtained by replacing $t$ by $nT$ in (6.5.11) and (6.5.12). Since $T_d = DT$, we obtain

$$x_{\text{lin}}(n) = \sum_{m=-\infty}^{\infty} x(m) g_{\text{lin}}(n - mD) \quad (6.5.13)$$

where

$$g_{\text{lin}}(n) = \begin{cases} 1 - \frac{|n|}{D}, & |n| \le D \\ 0, & |n| > D \end{cases} \quad (6.5.14)$$

As expected from any interpolation function, $g_{\text{lin}}(0) = 1$ and $g_{\text{lin}}(n) = 0$ for $n = \pm D, \pm 2D, \ldots$. The performance of the linear interpolator can be assessed by comparing its Fourier transform

$$G_{\text{lin}}(\omega) = \frac{1}{D}\left[\frac{\sin(\omega D/2)}{\sin(\omega/2)}\right]^2 \quad (6.5.15)$$

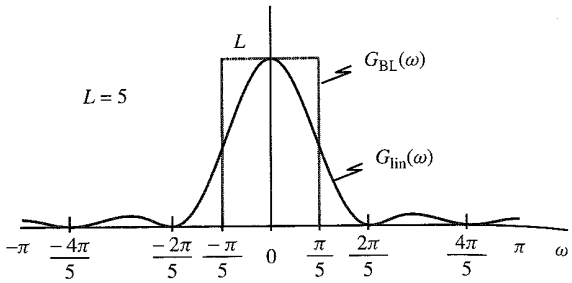



**Figure 6.5.3**
Frequency response of ideal and linear discrete-time interpolators.

to that of the ideal interpolator (6.5.8). This is illustrated in Figure 6.5.3 which shows that the linear interpolator has good performance only when the spectrum of the interpolated signal is negligible for $|\omega| > \pi/D$, that is, when the original continuous-time signal has been oversampled.

Equations (6.5.11) and (6.5.13) resemble a convolution summation; however, they are *not* convolutions. This is illustrated in Figure 6.5.4 which shows the computation of interpolated samples $x(nT)$ and $x((n+1)T)$ for $D = 5$. We note that only a subset of the coefficients of the linear interpolator is used in each case. Basically, we decompose $g_{\text{lin}}(n)$ into $D$ components and we use one at a time periodically to compute the interpolated values. This is essentially the idea behind the polyphase filter structures discussed in Chapter 11. However, if we create a new sequence $\tilde{x}(n)$ by inserting $(D-1)$ zero samples between successive samples of $x_d(m)$, we can compute $x(n)$ using the convolution

$$x(n) = \sum_{k=-\infty}^{\infty} \tilde{x}(k) g_{\text{lin}}(n-k) \tag{6.5.16}$$

at the expense of unnecessary computations involving zero values. A more efficient implementation can be obtained using equation (6.5.13).

Sampling and interpolation of a discrete-time signal essentially corresponds to a change of its sampling rate by an integer factor. The subject of sampling rate conversion, which is very important in practical applications, it is extensively discussed in Chapter 11.

### 6.5.2 Representation and Sampling of Bandpass Discrete-Time Signals

The bandpass representations of continuous-time signals, discussed in Section 6.4.3, can be adapted for discrete-time signals with some simple modifications that take into consideration the periodic nature of discrete-time spectra. Since we cannot require that the discrete-time Fourier transform is zero for $\omega < 0$ without violating its periodicity, we define the analytic signal $\psi(n)$ of a bandpass sequence $x(n)$ by

$$\Psi(\omega) = \begin{cases} 2X(\omega), & 0 \le \omega < \pi \\ 0, & -\pi \le \omega < 0 \end{cases} \tag{6.5.17}$$

where $X(\omega)$ and $\Psi(\omega)$ are the Fourier transforms of $x(n)$ and $\psi(n)$, respectively.

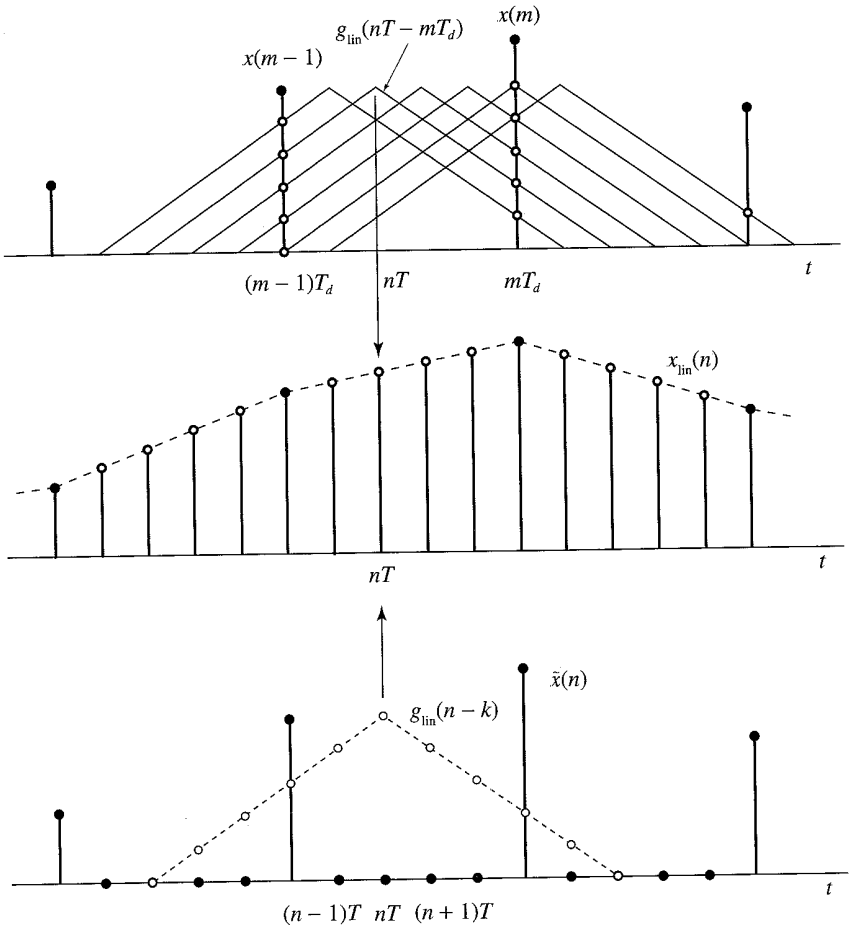



**Figure 6.5.4** Illustration of linear interpolation as a linear filtering process.

The ideal discrete-time Hilbert transformer, defined by

$$H(\omega) = \begin{cases} -j, & 0 < \omega < \pi \\ j, & -\pi < \omega < 0 \end{cases} \tag{6.5.18}$$

is a 90-degree phase shifter as in the continuous-time case. We can easily show that

$$\Psi(\omega) = X(\omega) + j\hat{X}(\omega) \tag{6.5.19}$$

where

$$\hat{X}(\omega) = H(\omega)X(\omega) \tag{6.5.20}$$

To compute the analytic signal in the time domain, we need the impulse response of the Hilbert transformer. It is obtained by

$$h(n) = \frac{1}{2\pi}\int_{-\pi}^{0} je^{j\omega n}d\omega - \frac{1}{2\pi}\int_{0}^{\pi} je^{j\omega n}d\omega \tag{6.5.21}$$

which yields

$$h(n) = \begin{cases} \frac{2}{\pi}\frac{\sin^2(\pi n/2)}{n}, & n \neq 0 \\ 0, & n = 0 \end{cases} = \begin{cases} 0, & n = \text{even} \\ \frac{2}{\pi n}, & n = \text{odd} \end{cases} \tag{6.5.22}$$

The sequence $h(n)$ is nonzero for $n < 0$ and not absolutely summable; thus, the ideal Hilbert transformer is noncausal and unstable. The impulse response and the frequency response of the ideal Hilbert transformer are illustrated in Figure 6.5.5.

As in the continuous-time case the Hilbert transform $\hat{x}(n)$ of a sequence $x(n)$ provides the imaginary part of its analytic signal representation, that is,

$$\psi(n) = x(n) + j\hat{x}(n) \tag{6.5.23}$$

The complex envelope, quadrature, and envelope/phase representations are obtained by the corresponding formulas for continuous-time signals by replacing $t$ by $nT$ in all relevant equations.

Given a bandpass sequence $x(n)$, $0 < \omega_L \leq |\omega| \leq \omega_L + w$ with normalized bandwidth $w = 2\pi B/F_s$, we can derive equivalent complex envelope or in-phase and quadrature lowpass representations that can be sampled at a rate $f_s = 1/D$

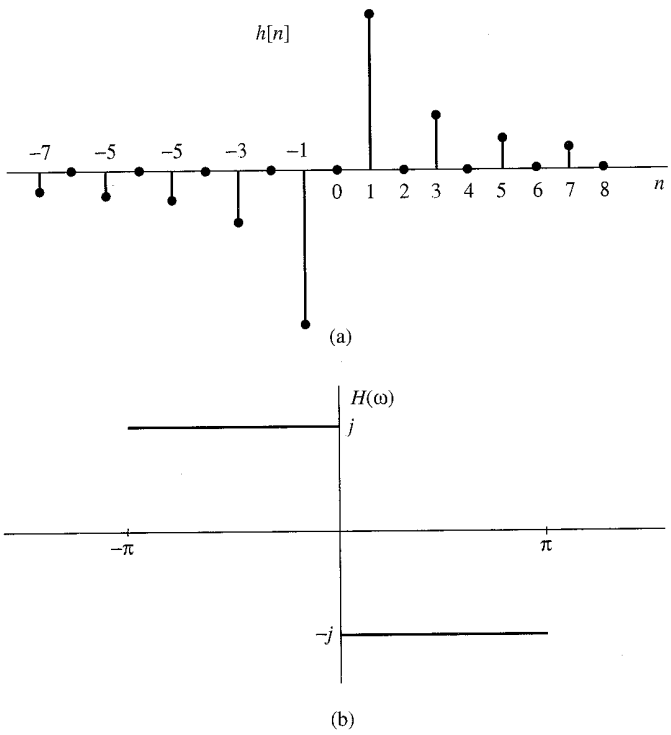



**Figure 6.5.5** Impulse response (a) and frequency response (b) of the discrete-time Hilbert transformer.

compatible with the bandwidth $w$. If $\omega_L = (k-1)\pi/D$ and $w = \pi/D$ the sequence $x(n)$ can be sampled directly without any aliasing as described in Section 11.7.

In many radar and communication systems applications it is necessary to process a bandpass signal $x_a(t)$, $F_L \leq |F| \leq F_L + B$ in lowpass form. Conventional techniques employ two quadrature analog channels and two A/D converters following the two lowpass filters in Figure 6.4.8. A more up-to-date approach is to uniformly sample the analog signal and then obtain the quadrature representation using digital quadrature demodulation, that is, a discrete-time implementation of the first part of Figure 6.4.8. A similar approach can be used to digitally generate single sideband signals for communications applications (Frerking 1994).

## 6.6 Oversampling A/D and D/A Converters

In this section we treat oversampling A/D and D/A converters.

### 6.6.1 Oversampling A/D Converters

The basic idea in oversampling A/D converters is to increase the sampling rate of the signal to the point where a low-resolution quantizer suffices. By oversampling, we can reduce the dynamic range of the signal values between successive samples and thus reduce the resolution requirements on the quantizer. As we have observed in the preceding section, the variance of the quantization error in A/D conversion is $\sigma_e^2 = \Delta^2/12$, where $\Delta = R/2^{b+1}$. Since the dynamic range of the signal, which is proportional to its standard deviation $\sigma_x$, should match the range $R$ of the quantizer, it follows that $\Delta$ is proportional to $\sigma_x$. Hence for a given number of bits, the power of the quantization noise is proportional to the variance of the signal to be quantized. Consequently, for a given fixed SQNR, a reduction in the variance of the signal to be quantized allows us to reduce the number of bits in the quantizer.

The basic idea for reducing the dynamic range leads us to consider *differential quantization*. To illustrate this point, let us evaluate the variance of the difference between two successive signal samples. Thus we have

$$d(n) = x(n) - x(n-1) \tag{6.6.1}$$

The variance of $d(n)$ is

$$\begin{aligned} \sigma_d^2 &= E[d^2(n)] = E\{[x(n) - x(n-1)]^2\} \\ &= E[x^2(n)] - 2E[x(n)x(n-1)] + E[x^2(n-1)] \\ &= 2\sigma_x^2[1 - \gamma_{xx}(1)] \end{aligned} \tag{6.6.2}$$

where $\gamma_{xx}(1)$ is the value of the autocorrelation sequence $\gamma_{xx}(m)$ of $x(n)$ evaluated at $m = 1$. If $\gamma_{xx}(1) > 0.5$, we observe that $\sigma_d^2 < \sigma_x^2$. Under this condition, it is better to quantize the difference $d(n)$ and to recover $x(n)$ from the quantized values $\{d_q(n)\}$. To obtain a high correlation between successive samples of the signal, we require that the sampling rate be significantly higher than the Nyquist rate.

An even better approach is to quantize the difference

$$d(n) = x(n) - ax(n-1) \tag{6.6.3}$$

where $a$ is a parameter selected to minimize the variance in $d(n)$. This leads to the result (see Problem 6.16) that the optimum choice of $a$ is

$$a = \frac{\gamma_{xx}(1)}{\gamma_{xx}(0)} = \frac{\gamma_{xx}(1)}{\sigma_x^2}$$

and

$$\sigma_d^2 = \sigma_x^2[1 - a^2] \tag{6.6.4}$$

In this case, $\sigma_d^2 < \sigma_x^2$, since $0 \le a \le 1$. The quantity $ax(n-1)$ is called a first-order predictor of $x(n)$.

Figure 6.6.1 shows a more general *differential predictive* signal quantizer system. This system is used in speech encoding and transmission over telephone channels and is known as differential pulse code modulation (DPCM). The goal of the predictor is to provide an estimate $\hat{x}(n)$ of $x(n)$ from a linear combination of past values of $x(n)$, so as to reduce the dynamic range of the difference signal $d(n) = x(n) - \hat{x}(n)$. Thus a predictor of order $p$ has the form

$$\hat{x}(n) = \sum_{k=1}^{p} a_k x(n-k) \tag{6.6.5}$$

The use of the feedback loop around the quantizer as shown in Fig. 6.6.1 is necessary to avoid the accumulation of quantization errors at the decoder. In this configuration, the error $e(n) = d(n) - d_q(n)$ is

$$e(n) = d(n) - d_q(n) = x(n) - \hat{x}(n) - d_q(n) = x(n) - x_q(n)$$

Thus the error in the reconstructed quantized signal $x_q(n)$ is equal to the quantization error for the sample $d(n)$. The decoder for DPCM that reconstructs the signal from the quantized values is also shown in Fig. 6.6.1.

The simplest form of differential predictive quantization is called *delta modulation* (DM). In DM, the quantizer is a simple 1-bit (two-level) quantizer and the

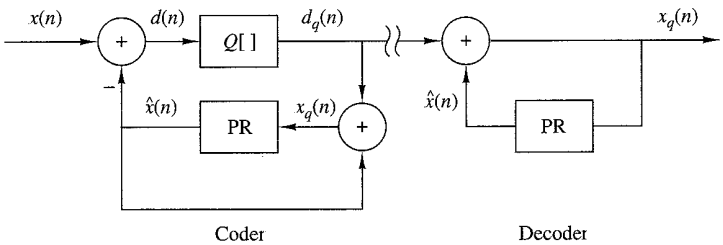



**Figure 6.6.1** Encoder and decoder for differential predictive signal quantizer system.

predictor is a first-order predictor, as shown in Fig. 6.6.2(a). Basically, DM provides a staircase approximation of the input signal. At every sampling instant, the sign of the difference between the input sample $x(n)$ and its most recent staircase approximation $\hat{x}(n) = ax_q(n-1)$ is determined, and then the staircase signal is updated by a step $\Delta$ in the direction of the difference.

From Fig. 6.6.2(a) we observe that

$$x_q(n) = ax_q(n-1) + d_q(n) \tag{6.6.6}$$

which is the discrete-time equivalent of an analog integrator. If $a = 1$, we have an ideal accumulator (integrator) whereas the choice $a < 1$ results in a "leaky integra-

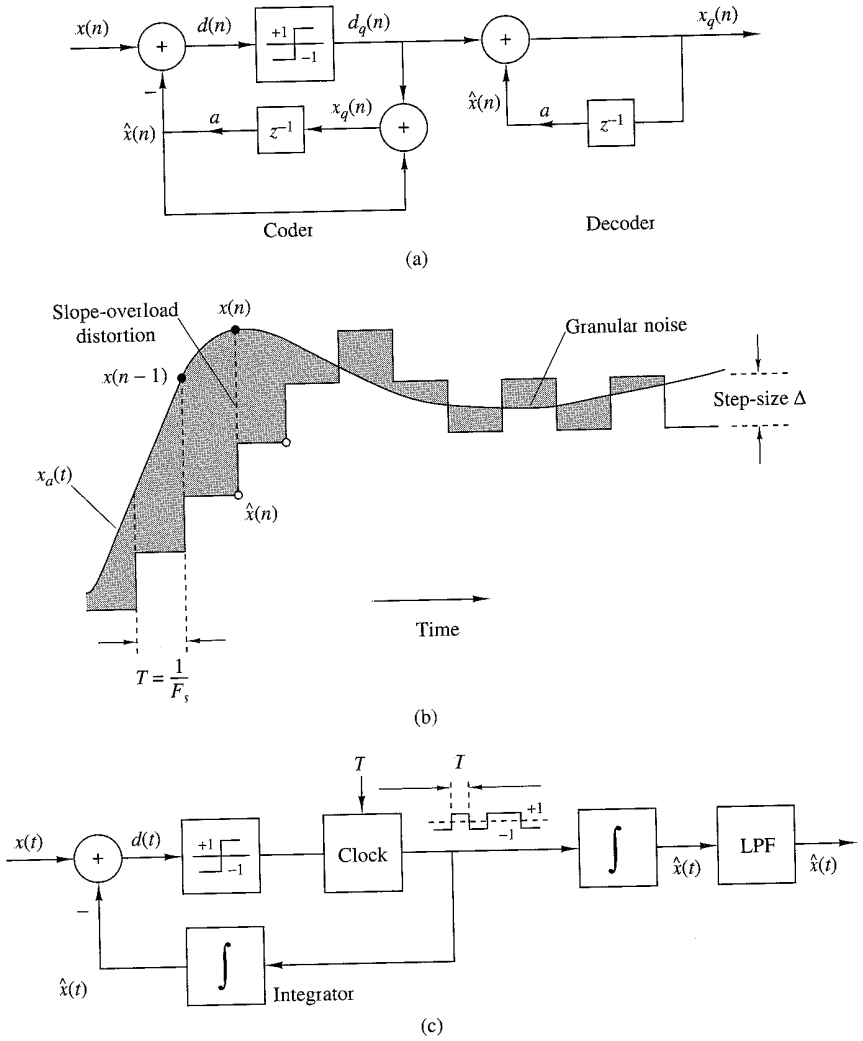



**Figure 6.6.2** Delta modulation system and two types of quantization errors.

tor." Figure 6.6.2(c) shows an analog model that illustrates the basic principle for the practical implementation of a DM system. The analog lowpass filter is necessary for the rejection of out-of-band components in the frequency range between $B$ and $F_s/2$, since $F_s >> B$ due to oversampling.

The crosshatched areas in Fig. 6.6.2(b) illustrate two types of quantization error in DM, slope-overload distortion and granular noise. Since the maximum slope $\Delta/T$ in $x(n)$ is limited by the step size, slope-overload distortion can be avoided if $\max |dx(t)/dt| \leq \Delta/T$. The granular noise occurs when the DM tracks a relatively flat (slowly changing) input signal. We note that increasing $\Delta$ reduces overload distortion but increases the granular noise, and vice versa.

One way to reduce these two types of distortion is to use an integrator in front of the DM, as shown in Fig. 6.6.3(a). This has two effects. First, it emphasizes the low frequencies of $x(t)$ and increases the correlation of the signal into the DM input. Second, it simplifies the DM decoder because the differentiator (inverse system) required at the decoder is canceled by the DM integrator. Hence the decoder is simply a lowpass filter, as shown in Fig. 6.6.3(a). Furthermore, the two integrators at the encoder can be replaced by a single integrator placed before the comparator, as shown in Fig. 6.6.3(b). This system is known as *sigma-delta modulation* (SDM).

SDM is an ideal candidate for A/D conversion. Such a converter takes advantage of the high sampling rate and spreads the quantization noise across the band up to $F_s/2$. Since $F_s >> B$, the noise in the signal-free band $B \leq F \leq F_s/2$ can be

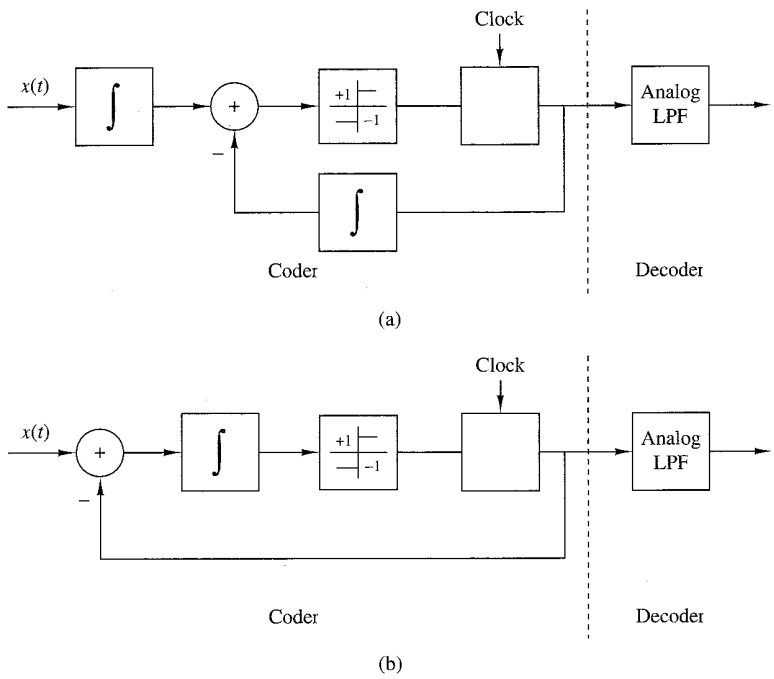



**Figure 6.6.3** Sigma-delta modulation system.

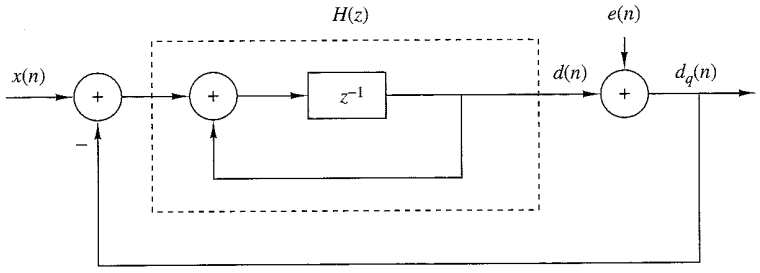



**Figure 6.6.4** Discrete-time model of sigma-delta modulation.

removed by appropriate digital filtering. To illustrate this principle, let us consider the discrete-time model of SDM, shown in Fig. 6.6.4, where we have assumed that the comparator (1-bit quantizer) is modeled by an additive white noise source with variance $\sigma_e^2 = \Delta^2/12$. The integrator is modeled by the discrete-time system with system function

$$H(z) = \frac{z^{-1}}{1 - z^{-1}} \tag{6.6.7}$$

The $z$-transform of the sequence $\{d_q(n)\}$ is

$$\begin{aligned} D_q(z) &= \frac{H(z)}{1 + H(z)} X(z) + \frac{1}{1 + H(z)} E(z) \\ &= H_s(z)X(z) + H_n(z)E(z) \end{aligned} \tag{6.6.8}$$

where $H_s(z)$ and $H_n(z)$ are the signal and noise system functions, respectively. A good SDM system has a flat frequency response $H_s(\omega)$ in the signal frequency band $0 \le F \le B$. On the other hand, $H_n(z)$ should have high attenuation in the frequency band $0 \le F \le B$ and low attenuation in the band $B \le F \le F_s/2$.

For the first-order SDM system with the integrator specified by (6.6.7), we have

$$H_s(z) = z^{-1}, \qquad H_n(z) = 1 - z^{-1} \tag{6.6.9}$$

Thus $H_s(z)$ does not distort the signal. The performance of the SDM system is therefore determined by the noise system function $H_n(z)$, which has a magnitude frequency response

$$|H_n(F)| = 2\left|\sin \frac{\pi F}{F_s}\right| \tag{6.6.10}$$

as shown in Fig. 6.6.5. The in-band quantization noise variance is given as

$$\sigma_n^2 = \int_{-B}^{B} |H_n(F)|^2 S_e(F)\, dF \tag{6.6.11}$$

where $S_e(F) = \sigma_e^2/F_s$ is the power spectral density of the quantization noise. From this relationship we note that doubling $F_s$ (increasing the sampling rate by a factor of 2), while keeping $B$ fixed, reduces the power of the quantization noise by 3 dB. This result is true for any quantizer. However, additional reduction may be possible by properly choosing the filter $H(z)$.

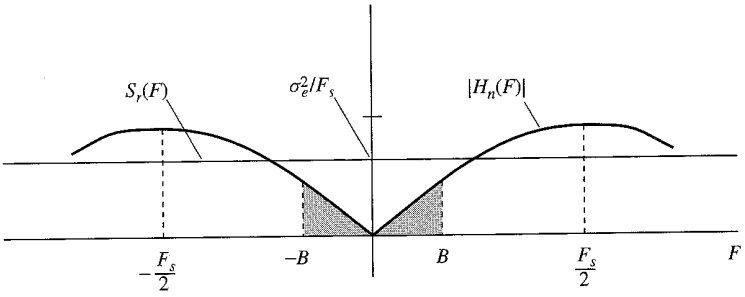



**Figure 6.6.5** Frequency (magnitude) response of noise system function.

For the first-order SDM, it can be shown (see Problem 6.19) that for $F_s >> 2B$, the in-band quantization noise power is

$$\sigma_n^2 \approx \frac{1}{3}\pi^2\sigma_e^2\left(\frac{2B}{F_s}\right)^3 \tag{6.6.12}$$

Note that doubling the sampling frequency reduces the noise power by 9 dB, of which 3 dB is due to the reduction in $S_e(F)$ and 6 dB is due to the filter characteristic $H_n(F)$. An additional 6-dB reduction can be achieved by using a double integrator (see Problem 6.20).

In summary, the noise power $\sigma_n^2$ can be reduced by increasing the sampling rate to spread the quantization noise power over a larger frequency band $(-F_s/2, F_s/2)$, and then shaping the noise power spectral density by means of an appropriate filter. Thus, SDM provides a 1-bit quantized signal at a sampling frequency $F_s = 2IB$, where the oversampling (interpolation) factor $I$ determines the SNR of the SDM quantizer.

Next, we explain how to convert this signal into a $b$-bit quantized signal at the Nyquist rate. First, we recall that the SDM decoder is an analog lowpass filter with a cutoff frequency $B$. The output of this filter is an approximation to the input signal $x(t)$. Given the 1-bit signal $d_q(n)$ at sampling frequency $F_s$, we can obtain a signal $x_q(n)$ at a lower sampling frequency, say the Nyquist rate of $2B$ or somewhat faster, by resampling the output of the lowpass filter at the $2B$ rate. To avoid aliasing, we first filter out the out-of-band $(B, F_s/2)$ noise by processing the wideband signal. The signal is then passed through the lowpass filter and resampled (down-sampled) at the lower rate. The down-sampling process is called *decimation* and is treated in great detail in Chapter 11.

For example, if the interpolation factor is $I = 256$, the A/D converter output can be obtained by averaging successive nonoverlapping blocks of 128 bits. This averaging would result in a digital signal with a range of values from zero to $256(b \approx 8$ bits) at the Nyquist rate. The averaging process also provides the required antialiasing filtering.

Figure 6.6.6 illustrates the basic elements of an oversampling A/D converter. Oversampling A/D converters for voiceband (3-kHz) signals are currently fabricated

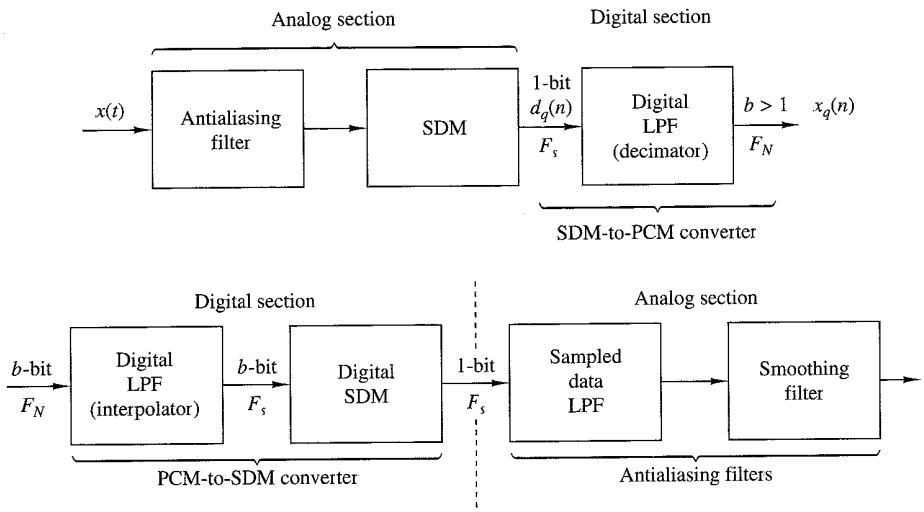



**Figure 6.6.6** Basic elements of an oversampling A/D converter.

as integrated circuits. Typically, they operate at a 2-MHz sampling rate, down-sample to 8 kHz, and provide 16-bit accuracy.

### 6.6.2 Oversampling D/A Converters

The elements of an oversampling D/A converter are shown in Fig. 6.6.7. As we observe, it is subdivided into a digital front end followed by an analog section. The digital section consists of an interpolator whose function is to increase the sampling rate by some factor $I$, which is followed by an SDM. The interpolator simply increases the digital sampling rate by inserting $I-1$ zeros between successive low rate samples. The resulting signal is then processed by a digital filter with cutoff frequency $F_c = B/F_s$ in order to reject the images (replicas) of the input signal spectrum. This higher rate signal is fed to the SDM, which creates a noise-shaped 1-bit sample. Each 1-bit sample is fed to the 1-bit D/A, which provides the analog interface to the antialiasing and smoothing filters. The output analog filters have a passband of $0 \leq F \leq B$ hertz and serve to smooth the signal and to remove the quantization noise in the frequency band $B \leq F \leq F_s/2$. In effect, the oversampling D/A converter uses SDM with the roles of the analog and digital sections reversed compared to the A/D converter.

In practice, oversampling D/A (and A/D) converters have many advantages over the more conventional D/A (and A/D) converters. First, the high sampling rate and

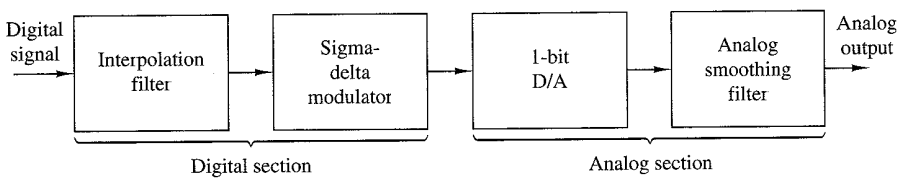



**Figure 6.6.7** Elements of an oversampling D/A converter.

the subsequent digital filtering minimize or remove the need for complex and expensive analog antialiasing filters. Furthermore, any analog noise introduced during the conversion phase is filtered out. Also, there is no need for S/H circuits. Oversampling SDM A/D and D/A converters are very robust with respect to variations in the analog circuit parameters, are inherently linear, and have low cost.

This concludes our discussion of signal reconstruction based on simple interpolation techniques. The techniques that we have described are easily incorporated into the design of practical D/A converters for the reconstruction of analog signals from digital signals. We shall consider interpolation again in Chapter 11 in the context of changing the sampling rate in a digital signal processing system.

## 6.7 Summary and References

The major focus of this chapter was on the sampling and reconstruction of signals. In particular, we treated the sampling of continuous-time signals and the subsequent operation of A/D conversion. These are necessary operations in the digital processing of analog signals, either on a general-purpose computer or on a custom-designed digital signal processor. The related issue of D/A conversion was also treated. In addition to the conventional A/D and D/A conversion techniques, we also described another type of A/D and D/A conversion, based on the principle of oversampling and a type of waveform encoding called sigma-delta modulation. Sigma-delta conversion technology is especially suitable for audio band signals due to their relatively small bandwidth (less than 20 kHz) and in some applications, the requirements for high fidelity.

The sampling theorem was introduced by Nyquist (1928) and later popularized in the classic paper by Shannon (1949). D/A and A/D conversion techniques are treated in a book by Sheingold (1986). Oversampling A/D and D/A conversion has been treated in the technical literature. Specifically, we cite the work of Candy (1986), Candy et al. (1981) and Gray (1990).

## Problems

**6.1** Given a continuous-time signal $x_a(t)$ with $X_a(F) = 0$ for $|F| > B$ determine the minimum sampling rate $F_s$ for a signal $y_a(t)$ defined by **(a)** $dx_a(t)/dt$ **(b)** $x_a^2(t)$ **(c)** $x_a(2t)$ **(d)** $x_a(t)\cos 6\pi Bt$ and **(e)** $x_a(t)\cos 7\pi Bt$

**6.2** The sampled sequence $x_a(nT)$ is reconstructed using an ideal D/A with interpolation function $g_a(t) = A$ for $|F| < F_c$ and zero otherwise to produce a continuous-time signal $\hat{x}_a(t)$.

**(a)** If the spectrum of the original signal $x_a(t)$ satisfies $X_a(F) = 0$ for $|F| > B$, find the maximum value of $T$, and the values of $F_c$, and $A$ such that $\hat{x}_a(t) = x_a(t)$.

**(b)** If $X_1(F) = 0$ for $|F| > B$, $X_2(F) = 0$ for $|F| > 2B$, and $x_a(t) = x_1(t)x_2(t)$, find the maximum value of $T$, and the values of $F_c$, and $A$ such that $\hat{x}_a(t) = x_a(t)$.

**(c)** Repeat part (b) for $x_a(t) = x_1(t)x_2(t/2)$.

**6.3** A continuous-time periodic signal with Fourier series coefficients $c_k = (1/2)^{|k|}$ and period $T_p = 0.1$ sec passes through an ideal lowpass filter with cutoff frequency $F_c = 102.5$ Hz. The resulting signal $y_a(t)$ is sampled periodically with $T = 0.005$ sec. Determine the spectrum of the sequence $y(n) = y_a(nT)$.

**6.4** Repeat Example 6.1.2 for the signal $x_a(t) = te^{-t}u_a(t)$.

**6.5** Consider the system in Figure 6.2.1. If $X_a(F) = 0$ for $|F| > F_s/2$, determine the frequency response $H(\omega)$ of the discrete-time system such that $y_a(t) = \int_{-\infty}^{t} x_a(\tau)d\tau$.

**6.6** Consider a signal $x_a(t)$ with spectrum $X_a(F) \neq 0$ for $0 < F_1 \leq |F| \leq F_2 < \infty$ and $X_a(F) = 0$ otherwise.

**(a)** Determine the minimum sampling frequency required to sample $x_a(t)$ without aliasing.

**(b)** Find the formula needed to reconstruct $x_a(t)$ from the samples $x_a(nT)$, $-\infty < n < \infty$.

**6.7** Prove the nonuniform second-order sampling interpolation formula described by equations (6.4.47)–(6.4.49).

**6.8** A discrete-time sample-and-hold interpolator, by a factor $I$, repeats the last input sample $(I - 1)$ times.

**(a)** Determine the interpolation function $g_{SH}(n)$.

**(b)** Determine the Fourier transform $G_{SH}(\omega)$ of $g_{SH}(n)$.

**(c)** Plot the magnitude and phase responses of the ideal interpolator, the linear interpolator, and the sample-and-hold interpolator for $I = 5$.

**6.9** *Time-domain sampling* Consider the continuous-time signal

$$x_a(t) = \begin{cases} e^{-j2\pi F_0 t}, & t \geq 0 \\ 0, & t < 0 \end{cases}$$

**(a)** Compute analytically the spectrum $X_a(F)$ of $x_a(t)$.

**(b)** Compute analytically the spectrum of the signal $x(n) = x_a(nT)$, $T = 1/F_s$.

**(c)** Plot the magnitude spectrum $|X_a(F)|$ for $F_0 = 10$ Hz.

**(d)** Plot the magnitude spectrum $|X(F)|$ for $F_s = 10, 20, 40$, and $100$ Hz.

**(e)** Explain the results obtained in part (d) in terms of the aliasing effect.

**6.10** Consider the sampling of the bandpass signal whose spectrum is illustrated in Fig. P6.10. Determine the minimum sampling rate $F_s$ to avoid aliasing.

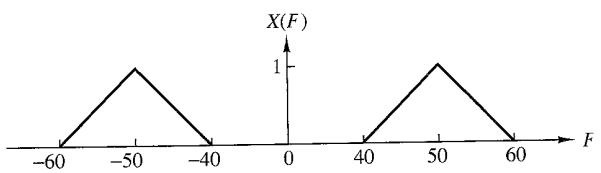



**Figure P6.10**

**6.11** Consider the sampling of the bandpass signal whose spectrum is illustrated in Fig P6.11. Determine the minimum sampling rate $F_s$ to avoid aliasing.

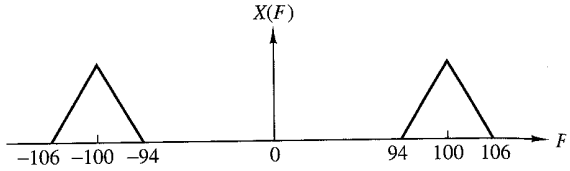



**Figure P6.11**

**6.12** Consider the two systems shown in Fig. P6.12.

**(a)** Sketch the spectra of the various signals if $x_a(t)$ has the Fourier transform shown in Fig. P6.12(b) and $F_s = 2B$. How are $y_1(t)$ and $y_2(t)$ related to $x_a(t)$?

**(b)** Determine $y_1(t)$ and $y_2(t)$ if $x_a(t) = \cos 2\pi F_0 t$, $F_0 = 20$ Hz, and $F_s = 50$ Hz or $F_s = 30$ Hz.

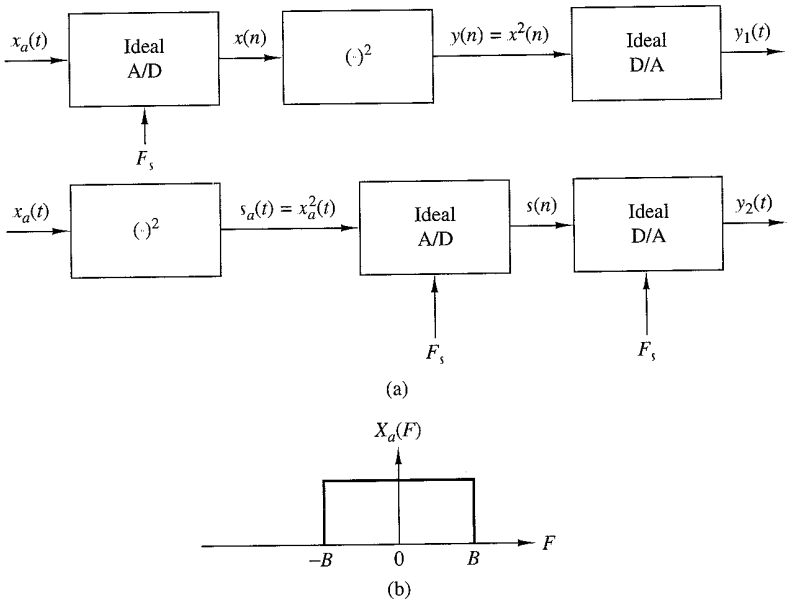



**Figure P6.12**

**6.13** A continuous-time signal $x_a(t)$ with bandwidth $B$ and its echo $x_a(t-\tau)$ arrive simultaneously at a TV receiver. The received analog signal

$$s_a(t) = x_a(t) + \alpha x_a(t-\tau), \qquad |\alpha| < 1$$

is processed by the system shown in Fig. P6.13. Is it possible to specify $F_s$ and $H(z)$ so that $y_a(t) = x_a(t)$ [i.e., remove the "ghost" $x_a(t-\tau)$ from the received signal]?

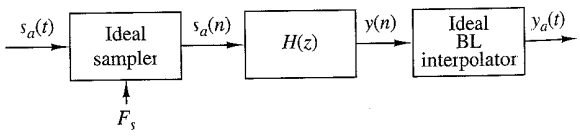



**Figure P6.13**

**6.14** A bandlimited continuous-time signal $x_a(t)$ is sampled at a sampling frequency $F_s \geq 2B$. Determine the energy $E_d$ of the resulting discrete-time signal $x(n)$ as a function of the energy of the analog signal, $E_a$, and the sampling period $T = 1/F_s$.

**6.15** In a linear interpolator successive sample points are connected by straight-line segments. Thus the resulting interpolated signal $\hat{x}(t)$ can be repressed as

$$\hat{x}(t) = x(nT - T) + \frac{x(nT) - x(nT - T)}{T}(t - nT), \quad nT \leq t \leq (n+1)T$$

We observe that at $t = nT$, $\hat{x}(nT) = x(nT - T)$ and at $t = nT + T$, $\hat{x}(nT + T) = x(nT)$. Therefore, $x(t)$ has an inherent delay of $T$ seconds in interpolating the actual signal $x(t)$. Figure P6.15 illustrates this linear interpolation technique.

**(a)** Viewed as a linear filter, show that the linear interpolation with a $T$-second delay has an impulse response

$$h(t) = \begin{cases} t/T, & 0 \leq t < T \\ 2 - t/T, & T \leq t < 2T \\ 0, & \text{otherwise} \end{cases}$$

Derive the corresponding frequency response $H(F)$.

**(b)** Plot $|H(F)|$ and compare this frequency response with that of the ideal reconstruction filter for a lowpass bandlimited signal.

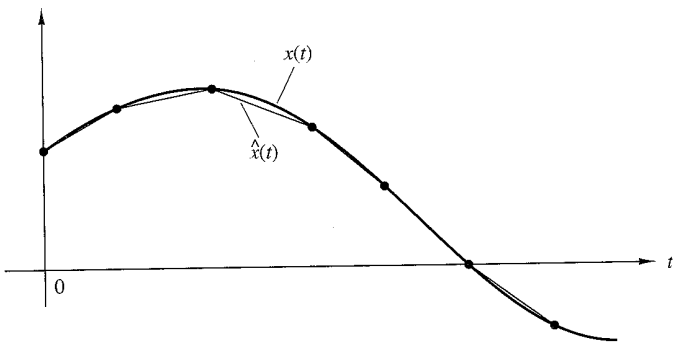



**Figure P6.15**

**6.16** Let $x(n)$ be a zero-mean stationary process with variance $\sigma_x^2$ and autocorrelation $\gamma_x(l)$.

**(a)** Show that the variance $\sigma_d^2$ of the first-order prediction error

$$d(n) = x(n) - ax(n-1)$$

is given by

$$\sigma_d^2 = \sigma_x^2[1 + a^2 - 2a\rho_x(1)]$$

where $\rho_x(1) = \gamma_x(1)/\gamma_x(0)$ is the normalized autocorrelation sequence.

**(b)** Show that $\sigma_d^2$ attains its minimum value

$$\sigma_d^2 = \sigma_x^2[1 - \rho_x^2(1)]$$

for $a = \gamma_x(1)/\gamma_x(0) = \rho_x(1)$.

**(c)** Under what conditions is $\sigma_d^2 < \sigma_x^2$?

**(d)** Repeat steps (a) to (c) for the second-order prediction error

$$d(n) = x(n) - a_1x(n-1) - a_2x(n-2)$$

**6.17** Consider a DM coder with input $x(n) = A\cos(2\pi nF/F_s)$. What is the condition for avoiding slope overload? Illustrate this condition graphically.

**6.18** Let $x_a(t)$ be a bandlimited signal with fixed bandwidth $B$ and variance $\sigma_x^2$.

**(a)** Show that the signal-to-quantization noise ratio, SQNR $= 10\log_{10}(\sigma_x^2/\sigma_e^2)$, increases by 3 dB each time we double the sampling frequency $F_s$. Assume that the quantization noise model discussed in Section 6.3.3 is valid.

**(b)** If we wish to increase the SQNR of a quantizer by doubling its sampling frequency, what is the most efficient way to do it? Should we choose a linear multibit A/D converter or an oversampling one?

**6.19** Consider the first-order SDM model shown in Fig. 6.6.4.

**(a)** Show that the quantization noise power in the signal band $(-B, B)$ is given by

$$\sigma_n^2 = \frac{2\sigma_e^2}{\pi}\left[\frac{2\pi B}{Fs} - \sin\left(2\pi\frac{B}{F_s}\right)\right]$$

**(b)** Using a two-term Taylor series expansion of the sine function and assuming that $F_s >> B$, show that

$$\sigma_n^2 \approx \frac{1}{3}\pi 2\sigma_e^2\left(\frac{2B}{F_s}\right)^3$$

**6.20** Consider the second-order SDM model shown in Fig. P6.20.

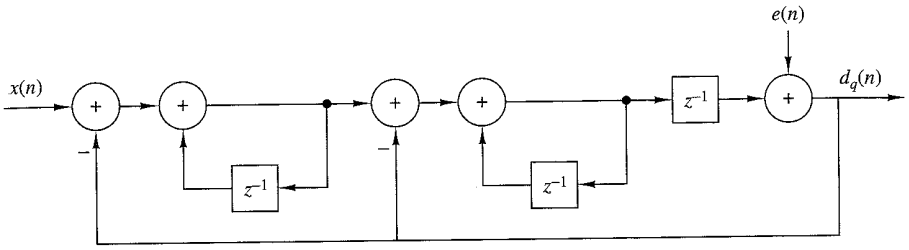



**Figure P6.20**

**(a)** Determine the signal and noise system functions $H_s(z)$ and $H_n(z)$, respectively.

**(b)** Plot the magnitude response for the noise system function and compare it with the one for the first-order SDM. Can you explain the 6-dB difference from these curves?

**(c)** Show that the in-band quantization noise power $\sigma_n^2$ is given approximately by

$$\sigma_n^2 \approx \frac{\pi\sigma_e^2}{5}\left(\frac{2B}{F_s}\right)^5$$

which implies a 15-dB increase for every doubling of the sampling frequency.

**6.21** Figure P6.21 illustrates the basic idea for a lookup-table-based sinusoidal signal generator. The samples of one period of the signal

$$x(n) = \cos\left(\frac{2\pi}{N}n\right), \qquad n = 0, 1, \ldots, N-1$$

are stored in memory. A digital sinusoidal signal is generated by stepping through the table and wrapping around at the end when the angle exceeds $2\pi$. This can be done by using modulo-$N$ addressing (i.e., using a "circular" buffer). Samples of $x(n)$ are feeding the ideal D/A converter every $T$ seconds.

**(a)** Show that by changing $F_s$ we can adjust the frequency $F_0$ of the resulting analog sinusoid.

**(b)** Suppose now that $F_s = 1/T$ is fixed. How many distinct analog sinusoids can be generated using the given lookup table? Explain.

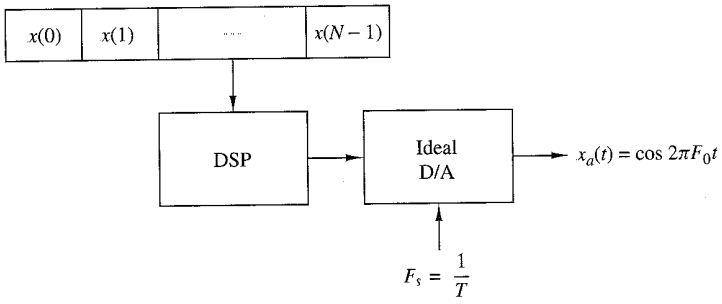



**Figure P6.21**

**6.22** Suppose that we represent an analog bandpass filter by the frequency response

$$H(F) = C(F - F_c) + C^*(-F - F_c)$$

where $C(f)$ is the frequency response of an equivalent lowpass filter, as shown in Fig. P6.22.

**(a)** Show that the impulse response $c(t)$ of the equivalent lowpass filter is related to the impulse response $h(t)$ of the bandpass filter as follows:

$$h(t) = 2\Re[c(t)e^{j2\pi F_c t}]$$

**(b)** Suppose that the bandpass system with frequency response $H(F)$ is excited by a bandpass signal of the form

$$x(t) = \Re[u(t)e^{j2\pi F_c t}]$$

where $u(t)$ is the equivalent lowpass signal. Show that the filter output may be expressed as

$$y(t) = \Re[v(t)e^{j2\pi F_c t}]$$

where

$$v(t) = \int^{\omega} c(\tau)u(t - \tau)d\tau$$

(*Hint:* Use the frequency domain to prove this result.)

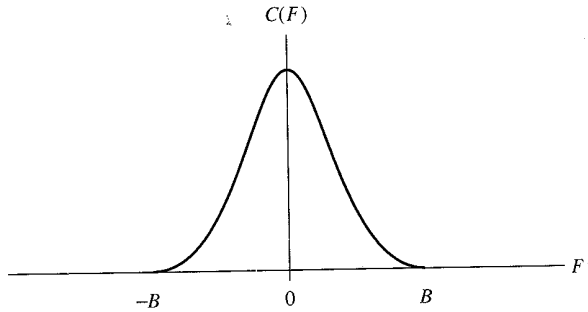



**Figure P6.22**

**6.23** Consider the sinusoidal signal generator in Fig. P6.23, where both the stored sinusoidal data

$$x(n) = \cos\left(\frac{2A}{N}n\right), \qquad 0 \le n \le N - 1$$

and the sampling frequency $F_s = 1/T$ are fixed. An engineer wishing to produce a sinusoid with period $2N$ suggests that we use either zero-order or first-order (linear) interpolation to double the number of samples per period in the original sinusoid as illustrated in Fig. P6.23(a).

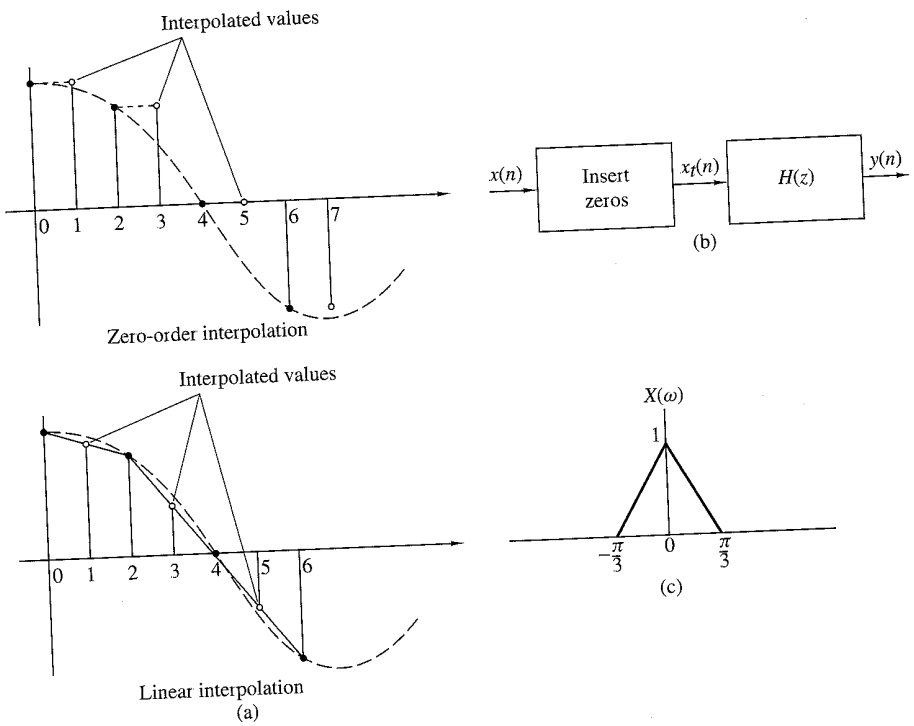



**Figure P6.23**

**(a)** Determine the signal sequences $y(n)$ generated using zero-order interpolation and linear interpolation and then compute the total harmonic distortion (THD) in each case for $N = 32, 64, 128$.

**(b)** Repeat part (a) assuming that all sample values are quantized to 8 bits.

**(c)** Show that the interpolated signal sequences $y(n)$ can be obtained by the system shown in Fig. P6.23(b). The first module inserts one zero sample between successive samples of $x(n)$. Determine the system $H(z)$ and sketch its magnitude response for the zero-order interpolation and for the linear interpolation cases. Can you explain the difference in performance in terms of the frequency response functions?

**(d)** Determine and sketch the spectra of the resulting sinusoids in each case both analytically [using the results in part (c)] and evaluating the DFT of the resulting signals.

**(e)** Sketch the spectra of $x_i(n)$ and $y(n)$, if $x(n)$ has the spectrum shown in Fig. P6.23(c) for both zero-order and linear interpolation. Can you suggest a better choice for $H(z)$?

**6.24** Let $x_a(t)$ be a time-limited signal; that is, $x_a(t) = 0$ for $|t| > \tau$, with Fourier transform $X_a(F)$. The function $X_a(F)$ is sampled with sampling interval $\delta F = 1/T_s$.

**(a)** Show that the function

$$x_p(t) = \sum_{n=-\infty}^{\infty} x_a(t - nT_s)$$

can be expressed as a Fourier series with coefficients

$$c_k = \frac{1}{T_s} X_a(k\delta F)$$

**(b)** Show that $X_a(F)$ can be recovered from the samples $X_a(k\delta F)$, $-\infty < k < \infty$ if $T_s \geq 2\tau$.

**(c)** Show that if $T_s < 2\tau$, there is "time-domain aliasing" that prevents exact reconstruction of $X_a(F)$.

**(d)** Show that if $T_s \geq 2\tau$, perfect reconstruction of $X_a(F)$ from the samples $X(k\delta F)$ is possible using the interpolation formula

$$X_a(F) = \sum_{k=-\infty}^{\infty} X_a(k\delta F) \frac{\sin[(\pi/\delta F)(F - k\delta F)]}{(\pi/\delta F)(F - k\delta F)}$$

ElCoM
Electrical | Computer | Mechatronics
ELCOM-HU.COM

CHAPTER

7

# The Discrete Fourier Transform: Its Properties and Applications

Frequency analysis of discrete-time signals is usually and most conveniently performed on a digital signal processor, which may be a general-purpose digital computer or specially designed digital hardware. To perform frequency analysis on a discrete-time signal $\{x(n)\}$, we convert the time-domain sequence to an equivalent frequency-domain representation. We know that such a representation is given by the Fourier transform $X(\omega)$ of the sequence $\{x(n)\}$. However, $X(\omega)$ is a continuous function of frequency and therefore it is not a computationally convenient representation of the sequence $\{x(n)\}$.

In this chapter we consider the representation of a sequence $\{x(n)\}$ by samples of its spectrum $X(\omega)$. Such a frequency-domain representation leads to the discrete Fourier transform (DFT), which is a powerful computational tool for performing frequency analysis of discrete-time signals.

## 7.1 Frequency-Domain Sampling: The Discrete Fourier Transform

Before we introduce the DFT, we consider the sampling of the Fourier transform of an aperiodic discrete-time sequence. Thus, we establish the relationship between the sampled Fourier transform and the DFT.

### 7.1.1 Frequency-Domain Sampling and Reconstruction of Discrete-Time Signals

We recall that aperiodic finite-energy signals have continuous spectra. Let us consider such an aperiodic discrete-time signal $x(n)$ with Fourier transform

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} \tag{7.1.1}$$

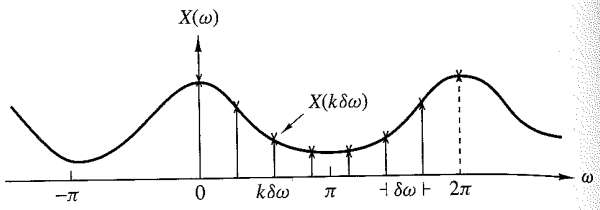



**Figure 7.1.1**
Frequency-domain sampling of the Fourier transform.

Suppose that we sample $X(\omega)$ periodically in frequency at a spacing of $\delta\omega$ radians between successive samples. Since $X(\omega)$ is periodic with period $2\pi$, only samples in the fundamental frequency range are necessary. For convenience, we take $N$ equidistant samples in the interval $0 \le \omega < 2\pi$ with spacing $\delta\omega = 2\pi/N$, as shown in Fig. 7.1.1. First, we consider the selection of $N$, the number of samples in the frequency domain.

If we evaluate (7.1.1) at $\omega = 2\pi k/N$, we obtain

$$X\left(\frac{2\pi}{N}k\right) = \sum_{n=-\infty}^{\infty} x(n)e^{-j2\pi kn/N}, \qquad k = 0, 1, \ldots, N-1 \tag{7.1.2}$$

The summation in (7.1.2) can be subdivided into an infinite number of summations, where each sum contains $N$ terms. Thus

$$\begin{aligned} X\left(\frac{2\pi}{N}k\right) &= \cdots + \sum_{n=-N}^{-1} x(n)e^{-j2\pi kn/N} + \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N} \\ &\quad + \sum_{n=N}^{2N-1} x(n)e^{-j2\pi kn/N} + \cdots \\ &= \sum_{l=-\infty}^{\infty} \sum_{n=lN}^{lN+N-1} x(n)e^{-j2\pi kn/N} \end{aligned}$$

If we change the index in the inner summation from $n$ to $n - lN$ and interchange the order of the summation, we obtain the result

$$X\left(\frac{2\pi}{N}k\right) = \sum_{n=0}^{N-1} \left[ \sum_{l=-\infty}^{\infty} x(n - lN) \right] e^{-j2\pi kn/N} \tag{7.1.3}$$

for $k = 0, 1, 2, \ldots, N-1$.

The signal

$$x_p(n) = \sum_{l=-\infty}^{\infty} x(n - lN) \tag{7.1.4}$$

obtained by the periodic repetition of $x(n)$ every $N$ samples, is clearly periodic with fundamental period $N$. Consequently, it can be expanded in a Fourier series as

$$x_p(n) = \sum_{k=0}^{N-1} c_k e^{j2\pi kn/N}, \qquad n = 0, 1, \ldots, N-1 \tag{7.1.5}$$

with Fourier coefficients

$$c_k = \frac{1}{N} \sum_{n=0}^{N-1} x_p(n) e^{-j2\pi kn/N}, \qquad k = 0, 1, \ldots, N-1 \tag{7.1.6}$$

Upon comparing (7.1.3) with (7.1.6), we conclude that

$$c_k = \frac{1}{N} X\left(\frac{2\pi}{N}k\right), \qquad k = 0, 1, \ldots, N-1 \tag{7.1.7}$$

Therefore,

$$x_p(n) = \frac{1}{N} \sum_{k=0}^{N-1} X\left(\frac{2\pi}{N}k\right) e^{j2\pi kn/N}, \qquad n = 0, 1, \ldots, N-1 \tag{7.1.8}$$

The relationship in (7.1.8) provides the reconstruction of the periodic signal $x_p(n)$ from the samples of the spectrum $X(\omega)$. However, it does not imply that we can recover $X(\omega)$ or $x(n)$ from the samples. To accomplish this, we need to consider the relationship between $x_p(n)$ and $x(n)$.

Since $x_p(n)$ is the periodic extension of $x(n)$ as given by (7.1.4), it is clear that $x(n)$ can be recovered from $x_p(n)$ if there is no aliasing in the time domain, that is, if $x(n)$ is time-limited to less than the period $N$ of $x_p(n)$. This situation is illustrated in Fig. 7.1.2, where without loss of generality, we consider a finite-duration sequence $x(n)$, which is nonzero in the interval $0 \le n \le L-1$. We observe that when $N \ge L$,

$$x(n) = x_p(n), \qquad 0 \le n \le N-1$$

so that $x(n)$ can be recovered from $x_p(n)$ without ambiguity. On the other hand, if $N < L$, it is not possible to recover $x(n)$ from its periodic extension due to *time-domain aliasing*. Thus, we conclude that the spectrum of an aperiodic discrete-time signal with finite duration $L$ can be exactly recovered from its samples at frequencies $\omega_k = 2\pi k/N$, if $N \ge L$. The procedure is to compute $x_p(n)$, $n = 0, 1, \ldots, N-1$ from (7.1.8); then

$$x(n) = \begin{cases} x_p(n), & 0 \le n \le N-1 \\ 0, & \text{elsewhere} \end{cases} \tag{7.1.9}$$

and finally, $X(\omega)$ can be computed from (7.1.1).

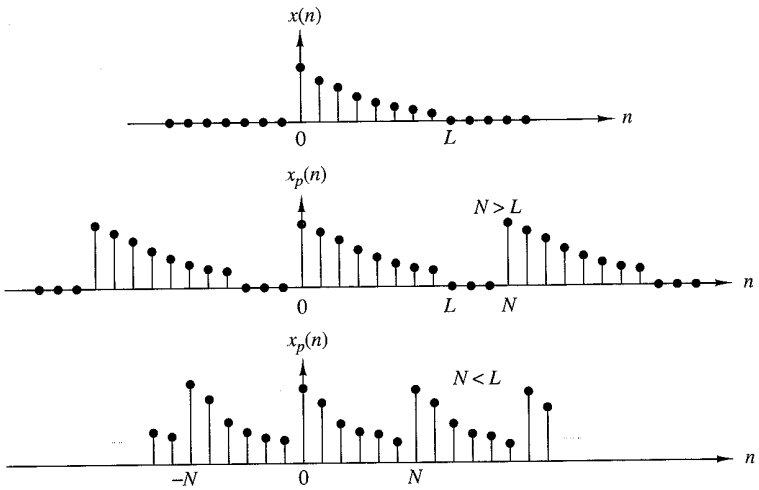



**Figure 7.1.2** Aperiodic sequence $x(n)$ of length $L$ and its periodic extension for $N \geq L$ (no aliasing) and $N < L$ (aliasing).

As in the case of continuous-time signals, it is possible to express the spectrum $X(\omega)$ directly in terms of its samples $X(2\pi k/N)$, $k = 0, 1, \ldots, N-1$. To derive such an interpolation formula for $X(\omega)$, we assume that $N \geq L$ and begin with (7.1.8). Since $x(n) = x_p(n)$ for $0 \leq n \leq N-1$,

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X\left(\frac{2\pi}{N}k\right) e^{j2\pi kn/N}, \qquad 0 \leq n \leq N-1 \tag{7.1.10}$$

If we use (7.1.1) and substitute for $x(n)$, we obtain

$$\begin{aligned} X(\omega) &= \sum_{n=0}^{N-1}\left[\frac{1}{N}\sum_{k=0}^{N-1} X\left(\frac{2\pi}{N}k\right) e^{j2\pi kn/N}\right] e^{-j\omega n} \\ &= \sum_{k=0}^{N-1} X\left(\frac{2\pi}{N}k\right)\left[\frac{1}{N}\sum_{n=0}^{N-1} e^{-j(\omega-2\pi k/N)n}\right] \end{aligned} \tag{7.1.11}$$

The inner summation term in the brackets of (7.1.11) represents the basic interpolation function shifted by $2\pi k/N$ in frequency. Indeed, if we define

$$\begin{aligned} P(\omega) &= \frac{1}{N}\sum_{n=0}^{N-1} e^{-j\omega n} = \frac{1}{N}\frac{1-e^{-j\omega N}}{1-e^{-j\omega}} \\ &= \frac{\sin(\omega N/2)}{N\sin(\omega/2)} e^{-j\omega(N-1)/2} \end{aligned} \tag{7.1.12}$$

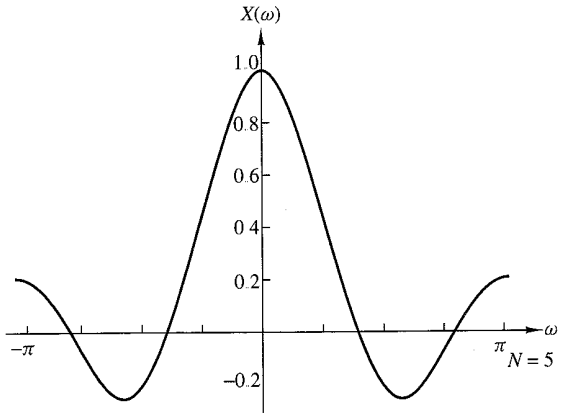



**Figure 7.1.3**
Plot of the function $[\sin(\omega N/2)]/[N\sin(\omega/2)]$.

then (7.1.11) can be expressed as

$$X(\omega) = \sum_{k=0}^{N-1} X\left(\frac{2\pi}{N}k\right) P\left(\omega - \frac{2\pi}{N}k\right), \qquad N \geq L \tag{7.1.13}$$

The interpolation function $P(\omega)$ is not the familiar $(\sin\theta)/\theta$ but instead, it is a periodic counterpart of it, and it is due to the periodic nature of $X(\omega)$. The phase shift in (7.1.12) reflects the fact that the signal $x(n)$ is a causal, finite-duration sequence of length $N$. The function $\sin(\omega N/2)/(N\sin(\omega/2))$ is plotted in Fig. 7.1.3 for $N = 5$. We observe that the function $P(\omega)$ has the property

$$P\left(\frac{2\pi}{N}k\right) = \begin{cases} 1, & k = 0 \\ 0, & k = 1, 2, \ldots, N-1 \end{cases} \tag{7.1.14}$$

Consequently, the interpolation formula in (7.1.13) gives exactly the sample values $X(2\pi k/N)$ for $\omega = 2\pi k/N$. At all other frequencies, the formula provides a properly weighted linear combination of the original spectral samples.

The following example illustrates the frequency-domain sampling of a discrete-time signal and the time-domain aliasing that results.

### EXAMPLE 7.1.1

Consider the signal

$$x(n) = a^n u(n), \qquad 0 < a < 1$$

The spectrum of this signal is sampled at frequencies $\omega_k = 2\pi k/N$, $k = 0, 1, \ldots, N-1$. Determine the reconstructed spectra for $a = 0.8$ when $N = 5$ and $N = 50$.

**Solution.** The Fourier transform of the sequence $x(n)$ is

$$X(\omega) = \sum_{n=0}^{\infty} a^n e^{-j\omega n} = \frac{1}{1 - ae^{-j\omega}}$$

Suppose that we sample $X(\omega)$ at $N$ equidistant frequencies $\omega_k = 2\pi k/N$, $k = 0, 1, \ldots, N-1$. Thus we obtain the spectral samples

$$X(\omega k) \equiv X\left(\frac{2\pi k}{N}\right) = \frac{1}{1 - ae^{-j2\pi k/N}}, \qquad k = 0, 1, \ldots, N-1$$

The periodic sequence $x_p(n)$, corresponding to the frequency samples $X(2\pi k/N)$, $k = 0, 1, \ldots, N-1$, can be obtained from either (7.1.4) or (7.1.8). Hence

$$\begin{aligned} x_p(n) &= \sum_{l=-\infty}^{\infty} x(n - lN) = \sum_{l=-\infty}^{0} a^{n-lN} \\ &= a^n \sum_{l=0}^{\infty} a^{lN} = \frac{a^n}{1 - a^N}, \qquad 0 \le n \le N-1 \end{aligned}$$

where the factor $1/(1 - a^N)$ represents the effect of aliasing. Since $0 < a < 1$, the aliasing error tends toward zero as $N \to \infty$.

For $a = 0.8$, the sequence $x(n)$ and its spectrum $X(\omega)$ are shown in Fig. 7.1.4(a) and 7.1.4(b), respectively. The aliased sequences $x_p(n)$ for $N = 5$ and $N = 50$ and the corresponding spectral samples are shown in Fig. 7.1.4(c) and 7.1.4(d), respectively. We note that the aliasing effects are negligible for $N = 50$.

If we define the aliased finite-duration sequence $x(n)$ as

$$\hat{x}(n) = \begin{cases} x_p(n), & 0 \le n \le N-1 \\ 0, & \text{otherwise} \end{cases}$$

then its Fourier transform is

$$\begin{aligned} \hat{X}(\omega) &= \sum_{n=0}^{N-1} \hat{x}(n) e^{-j\omega n} = \sum_{n=0}^{N-1} x_p(n) e^{-j\omega N} \\ &= \frac{1}{1 - a^N} \cdot \frac{1 - a^N e^{-j\omega n}}{1 - ae^{-j\omega}} \end{aligned}$$

Note that although $\hat{X}(\omega) \neq X(\omega)$, the sample values at $\omega_k = 2\pi k/N$ are identical. That is,

$$\hat{X}\left(\frac{2\pi}{N}k\right) = \frac{1}{1 - a^N} \cdot \frac{1 - a^N}{1 - ae^{-j2\pi k/N}} = X\left(\frac{2\pi}{N}k\right)$$

---

### 7.1.2 The Discrete Fourier Transform (DFT)

The development in the preceding section is concerned with the frequency-domain sampling of an aperiodic finite-energy sequence $x(n)$. In general, the equally spaced frequency samples $X(2\pi k/N)$, $k = 0, 1, \ldots, N-1$, do not uniquely represent the original sequence $x(n)$ when $x(n)$ has infinite duration. Instead, the frequency samples $X(2\pi k/N)$, $k = 0, 1, \ldots, N-1$, correspond to a periodic sequence $x_p(n)$ of

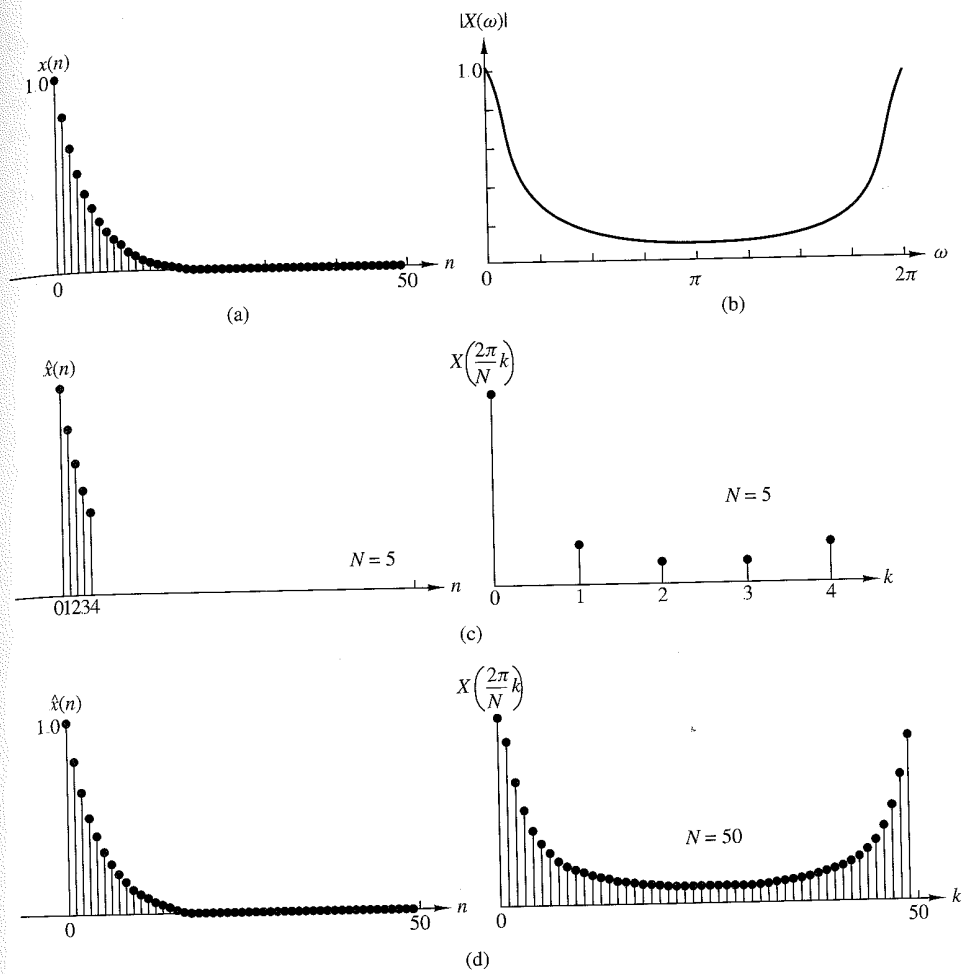



**Figure 7.1.4** (a) Plot of sequence $x(n) = (0.8)^n u(n)$; (b) its Fourier transform (magnitude only); (c) effect of aliasing with $N = 5$; (d) reduced effect of aliasing with $N = 50$.

period $N$, where $x_p(n)$ is an aliased version of $x(n)$, as indicated by the relation in (7.1.4), that is,

$$x_p(n) = \sum_{l=-\infty}^{\infty} x(n - lN) \tag{7.1.15}$$

When the sequence $x(n)$ has a finite duration of length $L \leq N$, then $x_p(n)$ is simply a periodic repetition of $x(n)$, where $x_p(n)$ over a single period is given as

$$x_p(n) = \begin{cases} x(n), & 0 \leq n \leq L-1 \\ 0, & L \leq n \leq N-1 \end{cases} \tag{7.1.16}$$

Consequently, the frequency samples $X(2\pi k/N)$, $k = 0, 1, \ldots, N-1$, uniquely represent the finite-duration sequence $x(n)$. Since $x(n) \equiv x_p(n)$ over a single period

(padded by $N - L$ zeros), the original finite-duration sequence $x(n)$ can be obtained from the frequency samples $\{X(2\pi k/N\}$ by means of the formula (7.1.8).

It is important to note that *zero padding* does not provide any additional information about the spectrum $X(\omega)$ of the sequence $\{x(n)\}$. The $L$ equidistant samples of $X(\omega)$ are sufficient to reconstruct $X(\omega)$ using the reconstruction formula (7.1.13). However, padding the sequence $\{x(n)\}$ with $N - L$ zeros and computing an $N$-point DFT results in a "better display" of the Fourier transform $X(\omega)$.

In summary, a finite-duration sequence $x(n)$ of length $L$ [i.e., $x(n) = 0$ for $n < 0$ and $n \geq L$] has a Fourier transform

$$X(\omega) = \sum_{n=0}^{L-1} x(n)e^{-j\omega n}, \qquad 0 \leq \omega \leq 2\pi \tag{7.1.17}$$

where the upper and lower indices in the summation reflect the fact that $x(n) = 0$ outside the range $0 \leq n \leq L-1$. When we sample $X(\omega)$ at equally spaced frequencies $\omega_k = 2\pi k/N$, $k = 0, 1, 2, \ldots, N-1$, where $N \geq L$, the resultant samples are

$$\begin{aligned} X(k) &\equiv X\left(\frac{2\pi k}{N}\right) = \sum_{n=0}^{L-1} x(n)e^{-j2\pi kn/N} \\ X(k) &= \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N}, \qquad k = 0, 1, 2, \ldots, N-1 \end{aligned} \tag{7.1.18}$$

where for convenience, the upper index in the sum has been increased from $L - 1$ to $N - 1$ since $x(n) = 0$ for $n \geq L$.

The relation in (7.1.18) is a formula for transforming a sequence $\{x(n)\}$ of length $L \leq N$ into a sequence of frequency samples $\{X(k)\}$ of length $N$. Since the frequency samples are obtained by evaluating the Fourier transform $X(\omega)$ at a set of $N$ (equally spaced) discrete frequencies, the relation in (7.1.18) is called the *discrete Fourier transform* (DFT) of $x(n)$. In turn, the relation given by (7.1.10), which allows us to recover the sequence $x(n)$ from the frequency samples

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{j2\pi kn/N}, \qquad n = 0, 1, \ldots, N-1 \tag{7.1.19}$$

is called the *inverse DFT* (IDFT). Clearly, when $x(n)$ has length $L < N$, the $N$-point IDFT yields $x(n) = 0$ for $L \leq n \leq N - 1$. To summarize, the formulas for the DFT and IDFT are

$$\text{DFT:} \quad X(k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N}, \qquad k = 0, 1, 2, \ldots, N-1 \tag{7.1.20}$$

$$\text{IDFT:} \quad x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{j2\pi kn/N}, \qquad n = 0, 1, 2, \ldots, N-1 \tag{7.1.21}$$

## EXAMPLE 7.1.2

A finite-duration sequence of length $L$ is given as

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

Determine the $N$-point DFT of this sequence for $N \ge L$.

**Solution.** The Fourier transform of this sequence is

$$X(\omega) = \sum_{n=0}^{L-1} x(n)e^{-j\omega n}$$

$$= \sum_{n=0}^{L-1} e^{-j\omega n} = \frac{1-e^{-j\omega L}}{1-e^{-j\omega}} = \frac{\sin(\omega L/2)}{\sin(\omega/2)} e^{-j\omega(L-1)/2}$$

The magnitude and phase of $X(\omega)$ are illustrated in Fig. 7.1.5 for $L = 10$. The $N$-point DFT of $x(n)$ is simply $X(\omega)$ evaluated at the set of $N$ equally spaced frequencies $\omega_k = 2\pi k/N$, $k = 0, 1, \ldots, N-1$. Hence

$$X(k) = \frac{1-e^{-j2\pi kL/N}}{1-e^{-j2\pi k/N}}, \qquad k = 0, 1, \ldots, N-1$$

$$= \frac{\sin(\pi kL/N)}{\sin(\pi k/N)} e^{-j\pi k(L-1)/N}$$

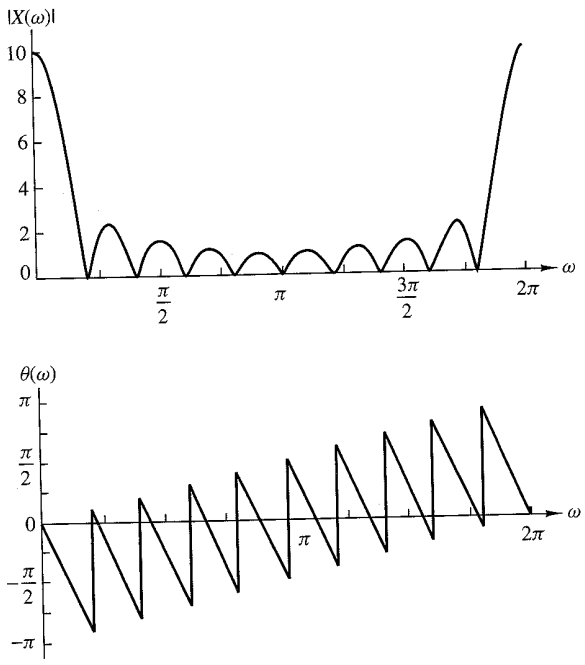



**Figure 7.1.5** Magnitude and phase characteristics of the Fourier transform for signal in Example 7.1.2.

If $N$ is selected such that $N = L$, then the DFT becomes

$$X(k) = \begin{cases} L, & k = 0 \\ 0, & k = 1, 2, \ldots, L-1 \end{cases}$$

Thus there is only one nonzero value in the DFT. This is apparent from observation of $X(\omega)$, since $X(\omega) = 0$ at the frequencies $\omega_k = 2\pi k/L$, $k \neq 0$. The reader should verify that $x(n)$ can be recovered from $X(k)$ by performing an $L$-point IDFT.

Although the $L$-point DFT is sufficient to uniquely represent the sequence $x(n)$ in the frequency domain, it is apparent that it does not provide sufficient detail to yield a good picture of the spectral characteristics of $x(n)$. If we wish to have a better picture, we must evaluate (interpolate) $X(\omega)$ at more closely spaced frequencies, say $\omega_k = 2\pi k/N$, where $N > L$. In effect, we can view this computation as expanding the size of the sequence from $L$ points to $N$ points by appending $N - L$ zeros to the sequence $x(n)$, that is, zero padding. Then the $N$-point DFT provides finer interpolation than the $L$-point DFT.

Figure 7.1.6 provides a plot of the $N$-point DFT, in magnitude and phase, for $L = 10$, $N = 50$, and $N = 100$. Now the spectral characteristics of the sequence are more clearly evident, as one will conclude by comparing these spectra with the continuous spectrum $X(\omega)$.

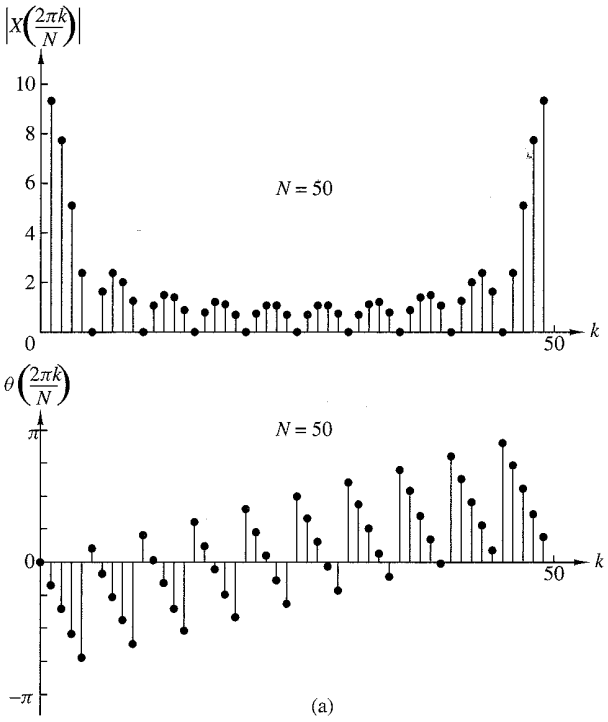



**Figure 7.1.6** Magnitude and phase of an $N$-point DFT in Example 7.1.2; (a) $L = 10$, $N = 50$; (b) $L = 10$, $N = 100$.

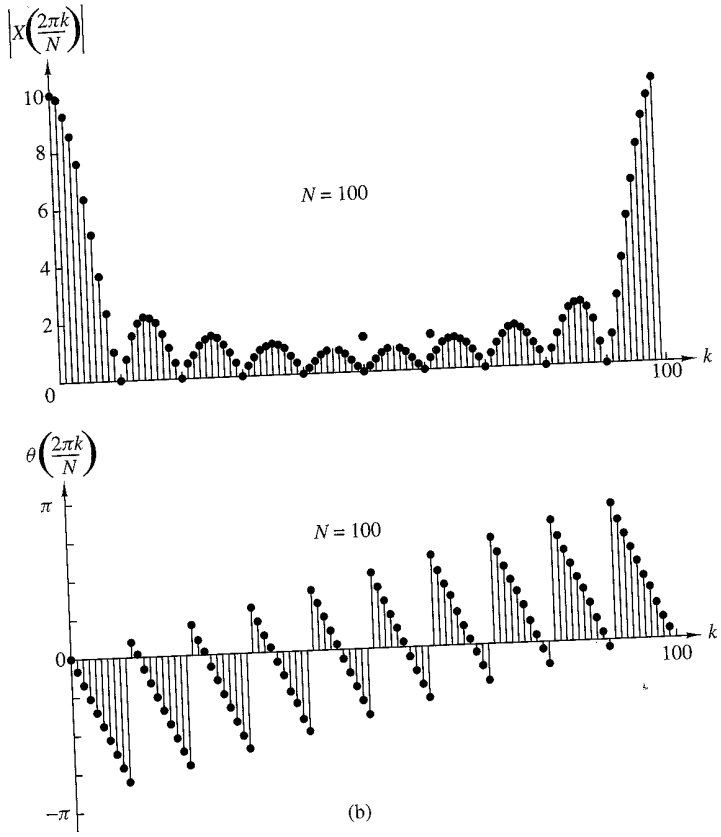



**Figure 7.1.6** *continued*

### 7.1.3 The DFT as a Linear Transformation

The formulas for the DFT and IDFT given by (7.1.18) and (7.1.19) may be expressed as

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}, \qquad k = 0, 1, \ldots, N-1 \tag{7.1.22}$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-kn}, \qquad n = 0, 1, \ldots, N-1 \tag{7.1.23}$$

where, by definition,

$$W_N = e^{-j2\pi/N} \tag{7.1.24}$$

which is an $N$th root of unity.

We note that the computation of each point of the DFT can be accomplished by $N$ complex multiplications and $(N-1)$ complex additions. Hence the $N$-point DFT values can be computed in a total of $N^2$ complex multiplications and $N(N-1)$ complex additions.

It is instructive to view the DFT and IDFT as linear transformations on sequences $\{x(n)\}$ and $\{X(k)\}$, respectively. Let us define an $N$-point vector $\mathbf{x}_N$ of the signal sequence $x(n)$, $n = 0, 1, \ldots, N-1$, an $N$-point vector $\mathbf{X}_N$ of frequency samples, and an $N \times N$ matrix $\mathbf{W}_N$ as

$$\mathbf{x}_N = \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(N-1) \end{bmatrix}, \qquad \mathbf{X}_N = \begin{bmatrix} X(0) \\ X(1) \\ \vdots \\ X(N-1) \end{bmatrix}$$

$$\mathbf{W}_N = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & W_N & W_N^2 & \cdots & W_N^{N-1} \\ & W_N^2 & W_N^4 & \cdots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \cdots & W_N^{(N-1)(N-1)} \end{bmatrix} \tag{7.1.25}$$

With these definitions, the $N$-point DFT may be expressed in matrix form as

$$\mathbf{X}_N = \mathbf{W}_N \mathbf{x}_N \tag{7.1.26}$$

where $\mathbf{W}_N$ is the matrix of the linear transformation. We observe that $\mathbf{W}_N$ is a symmetric matrix. If we assume that the inverse of $\mathbf{W}_N$ exists, then (7.1.26) can be inverted by premultiplying both sides by $\mathbf{W}_N^{-1}$. Thus we obtain

$$\mathbf{x}_N = \mathbf{W}_N^{-1} \mathbf{X}_N \tag{7.1.27}$$

But this is just an expression for the IDFT.

In fact, the IDFT as given by (7.1.23) can be expressed in matrix form as

$$\mathbf{x}_N = \frac{1}{N} \mathbf{W}_N^* \mathbf{X}_N \tag{7.1.28}$$

where $\mathbf{W}_N^*$ denotes the complex conjugate of the matrix $\mathbf{W}_N$. Comparison of (7.1.27) with (7.1.28) leads us to conclude that

$$\mathbf{W}_N^{-1} = \frac{1}{N} \mathbf{W}_N^* \tag{7.1.29}$$

which, in turn, implies that

$$\mathbf{W}_N \mathbf{W}_N^* = N \mathbf{I}_N \tag{7.1.30}$$

where $\mathbf{I}_N$ is an $N \times N$ identity matrix. Therefore, the matrix $\mathbf{W}_N$ in the transformation is an orthogonal (unitary) matrix. Furthermore, its inverse exists and is given as $\mathbf{W}_N^*/N$. Of course, the existence of the inverse of $\mathbf{W}_N$ was established previously from our derivation of the IDFT.

**EXAMPLE 7.1.3**

Compute the DFT of the four-point sequence

$$x(n) = (0 \quad 1 \quad 2 \quad 3)$$

**Solution.** The first step is to determine the matrix $\mathbf{W}_4$. By exploiting the periodicity property of $\mathbf{W}_4$ and the symmetry property

$$W_N^{k+N/2} = -W_N^k$$

the matrix $\mathbf{W}_4$ may be expressed as

$$\mathbf{W}_4 = \begin{bmatrix} W_4^0 & W_4^0 & W_4^0 & W_4^0 \\ W_4^0 & W_4^1 & W_4^2 & W_4^3 \\ W_4^0 & W_4^2 & W_4^4 & W_4^6 \\ W_4^0 & W_4^3 & W_4^6 & W_4^9 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W_4^1 & W_4^2 & W_4^3 \\ 1 & W_4^2 & W_4^0 & W_4^2 \\ 1 & W_4^3 & W_4^2 & W_4^1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix}$$

Then

$$\mathbf{X}_4 = \mathbf{W}_4\mathbf{x}_4 = \begin{bmatrix} 6 \\ -2+2j \\ -2 \\ -2-2j \end{bmatrix}$$

The IDFT of $\mathbf{X}_4$ may be determined by conjugating the elements in $\mathbf{W}_4$ to obtain $\mathbf{W}_4^*$ and then applying the formula (7.1.28).

---

The DFT and IDFT are computational tools that play a very important role in many digital signal processing applications, such as frequency analysis (spectrum analysis) of signals, power spectrum estimation, and linear filtering. The importance of the DFT and IDFT in such practical applications is due to a large extent to the existence of computationally efficient algorithms, known collectively as fast Fourier transform (FFT) algorithms, for computing the DFT and IDFT. This class of algorithms is described in Chapter 8.

### 7.1.4 Relationship of the DFT to Other Transforms

In this discussion we have indicated that the DFT is an important computational tool for performing frequency analysis of signals on digital signal processors. In view of the other frequency analysis tools and transforms that we have developed, it is important to establish the relationships of the DFT to these other transforms.

**Relationship to the Fourier series coefficients of a periodic sequence.** A periodic sequence $\{x_p(n)\}$ with fundamental period $N$ can be represented in a Fourier series of the form

$$x_p(n) = \sum_{k=0}^{N-1} c_k e^{j2\pi nk/N}, \qquad -\infty < n < \infty \tag{7.1.31}$$

where the Fourier series coefficients are given by the expression

$$c_k = \frac{1}{N} \sum_{n=0}^{N-1} x_p(n) e^{-j2\pi nk/N}, \qquad k = 0, 1, \ldots, N-1 \tag{7.1.32}$$

If we compare (7.1.31) and (7.1.32) with (7.1.18) and (7.1.19), we observe that the formula for the Fourier series coefficients has the form of a DFT. In fact, if we define a sequence $x(n) = x_p(n)$, $0 \le n \le N-1$, the DFT of this sequence is simply

$$X(k) = Nc_k \tag{7.1.33}$$

Furthermore, (7.1.31) has the form of an IDFT. Thus the $N$-point DFT provides the exact line spectrum of a periodic sequence with fundamental period $N$.

**Relationship to the Fourier transform of an aperiodic sequence.** We have already shown that if $x(n)$ is an aperiodic finite energy sequence with Fourier transform $X(\omega)$, which is sampled at $N$ equally spaced frequencies $\omega_k = 2\pi k/N$, $k = 0, 1, \ldots, N-1$, the spectral components

$$X(k) = X(\omega)|_{\omega=2\pi k/N} = \sum_{n=-\infty}^{\infty} x(n) e^{-j2\pi nk/N}, \qquad k = 0, 1, \ldots, N-1 \tag{7.1.34}$$

are the DFT coefficients of the periodic sequence of period $N$, given by

$$x_p(n) = \sum_{l=-\infty}^{\infty} x(n - lN) \tag{7.1.35}$$

Thus $x_p(n)$ is determined by aliasing $\{x(n)\}$ over the interval $0 \le n \le N-1$. The finite-duration sequence

$$\hat{x}(n) = \begin{cases} x_p(n), & 0 \le n \le N-1 \\ 0, & \text{otherwise} \end{cases} \tag{7.1.36}$$

bears no resemblance to the original sequence $\{x(n)\}$, unless $x(n)$ is of finite duration and length $L \le N$, in which case

$$x(n) = \hat{x}(n), \qquad 0 \le n \le N-1 \tag{7.1.37}$$

Only in this case will the IDFT of $\{X(k)\}$ yield the original sequence $\{x(n)\}$.

**Relationship to the $z$-transform.** Let us consider a sequence $x(n)$ having the $z$-transform

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n} \tag{7.1.38}$$

with an ROC that includes the unit circle. If $X(z)$ is sampled at the $N$ equally spaced points on the unit circle $z_k = e^{j2\pi k/N}$, $0, 1, 2, \ldots, N-1$, we obtain

$$\begin{aligned} X(k) &\equiv X(z)|_{z=e^{j2\pi nk/N}}, \qquad k = 0, 1, \ldots, N-1 \\ &= \sum_{n=-\infty}^{\infty} x(n)e^{-j2\pi nk/N} \end{aligned} \tag{7.1.39}$$

The expression in (7.1.39) is identical to the Fourier transform $X(\omega)$ evaluated at the $N$ equally spaced frequencies $\omega_k = 2\pi k/N$, $k = 0, 1, \ldots, N-1$, which is the topic treated in Section 7.1.1.

If the sequence $x(n)$ has a finite duration of length $N$ or less, the sequence can be recovered from its $N$-point DFT. Hence its z-transform is uniquely determined by its $N$-point DFT. Consequently, $X(z)$ can be expressed as a function of the DFT $\{X(k)\}$ as follows:

$$\begin{aligned} X(z) &= \sum_{n=0}^{N-1} x(n)z^{-n} \\ X(z) &= \sum_{n=0}^{N-1} \left[ \frac{1}{N} \sum_{k=0}^{N-1} X(k)e^{j2\pi kn/N} \right] z^{-n} \\ X(z) &= \frac{1}{N} \sum_{k=0}^{N-1} X(k) \sum_{n=0}^{N-1} \left( e^{j2\pi k/N} z^{-1} \right)^n \\ X(z) &= \frac{1 - z^{-N}}{N} \sum_{k=0}^{N-1} \frac{X(k)}{1 - e^{j2\pi k/N} z^{-1}} \end{aligned} \tag{7.1.40}$$

When evaluated on the unit circle, (7.1.40) yields the Fourier transform of the finite-duration sequence in terms of its DFT, in the form

$$X(\omega) = \frac{1 - e^{-j\omega N}}{N} \sum_{k=0}^{N-1} \frac{X(k)}{1 - e^{-j(\omega - 2\pi k/N)}} \tag{7.1.41}$$

This expression for the Fourier transform is a polynomial (Lagrange) interpolation formula for $X(\omega)$ expressed in terms of the values $\{X(k)\}$ of the polynomial at a set of equally spaced discrete frequencies $\omega_k = 2\pi k/N$, $k = 0, 1, \ldots, N-1$. With some algebraic manipulations, it is possible to reduce (7.1.41) to the interpolation formula given previously in (7.1.13).

**Relationship to the Fourier series coefficients of a continuous-time signal.** Suppose that $x_a(t)$ is a continuous-time periodic signal with fundamental period $T_p = 1/F_0$. The signal can be expressed in a Fourier series

$$x_a(t) = \sum_{k=-\infty}^{\infty} c_k e^{j2\pi k t F_0} \tag{7.1.42}$$

where $\{c_k\}$ are the Fourier coefficients. If we sample $x_a(t)$ at a uniform rate $F_s = N/T_p = 1/T$, we obtain the discrete-time sequence

$$\begin{aligned} x(n) \equiv x_a(nT) &= \sum_{k=-\infty}^{\infty} c_k e^{j2\pi k F_0 n T} = \sum_{k=-\infty}^{\infty} c_k e^{j2\pi kn/N} \\ &= \sum_{k=0}^{N-1} \left[ \sum_{l=-\infty}^{\infty} c_{k-lN} \right] e^{j2\pi kn/N} \end{aligned} \tag{7.1.43}$$

It is clear that (7.1.43) is in the form of an IDFT formula, where

$$X(k) = N \sum_{l=-\infty}^{\infty} c_{k-lN} \equiv N\tilde{c}_k \tag{7.1.44}$$

and

$$\tilde{c}_k = \sum_{l=-\infty}^{\infty} c_{k-lN} \tag{7.1.45}$$

Thus the $\{\tilde{c}_k\}$ sequence is an aliased version of the sequence $\{c_k\}$.

## 7.2 Properties of the DFT

In Section 7.1.2 we introduced the DFT as a set of $N$ samples $\{X(k)\}$ of the Fourier transform $X(\omega)$ for a finite-duration sequence $\{x(n)\}$ of length $L \le N$. The sampling of $X(\omega)$ occurs at the $N$ equally spaced frequencies $\omega_k = 2\pi k/N$, $k = 0, 1, 2, \ldots, N-1$. We demonstrated that the $N$ samples $\{X(k)\}$ uniquely represent the sequence $\{x(n)\}$ in the frequency domain. Recall that the DFT and inverse DFT (IDFT) for an $N$-point sequence $\{x(n)\}$ are given as

$$\text{DFT:} \quad X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}, \qquad k = 0, 1, \ldots, N-1 \tag{7.2.1}$$

$$\text{IDFT:} \quad x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-kn}, \qquad n = 0, 1, \ldots, N-1 \tag{7.2.2}$$

where $W_N$ is defined as

$$W_N = e^{-j2\pi/N} \tag{7.2.3}$$

In this section we present the important properties of the DFT. In view of the relationships established in Section 7.1.4 between the DFT and Fourier series, and Fourier transforms and $z$-transforms of discrete-time signals, we expect the properties of the DFT to resemble the properties of these other transforms and series. However, some important differences exist, one of which is the circular convolution property derived in the following section. A good understanding of these properties is extremely helpful in the application of the DFT to practical problems.

The notation used below to denote the $N$-point DFT pair $x(n)$ and $X(k)$ is

$$x(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X(k)$$

### 7.2.1 Periodicity, Linearity, and Symmetry Properties

**Periodicity.** If $x(n)$ and X(k) are an $N$-point DFT pair, then

$$x(n+N) = x(n) \qquad \text{for all } n \tag{7.2.4}$$

$$X(k+N) = X(k) \qquad \text{for all } k \tag{7.2.5}$$

These periodicities in $x(n)$ and $X(k)$ follow immediately from formulas (7.2.1) and (7.2.2) for the DFT and IDFT, respectively.

We previously illustrated the periodicity property in the sequence $x(n)$ for a given DFT. However, we had not previously viewed the DFT $X(k)$ as a periodic sequence. In some applications it is advantageous to do this.

**Linearity.** If

$$x_1(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X_1(k)$$

and

$$x_2(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X_2(k)$$

then for any real-valued or complex-valued constants $a_1$ and $a_2$,

$$a_1x_1(n) + a_2x_2(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} a_1X_1(k) + a_2X_2(k) \tag{7.2.6}$$

This property follows immediately from the definition of the DFT given by (7.2.1).

**Circular Symmetries of a Sequence.** As we have seen, the $N$-point DFT of a finite duration sequence $x(n)$, of length $L \le N$, is equivalent to the $N$-point DFT of a periodic sequence $x_p(n)$, of period $N$, which is obtained by periodically extending $x(n)$, that is,

$$x_p(n) = \sum_{l=-\infty}^{\infty} x(n - lN) \tag{7.2.7}$$

Now suppose that we shift the periodic sequence $x_p(n)$ by $k$ units to the right. Thus we obtain another periodic sequence

$$x_p'(n) = x_p(n-k) = \sum_{l=-\infty}^{\infty} x(n-k-lN) \tag{7.2.8}$$

The finite-duration sequence

$$x'(n) = \begin{cases} x_p'(n), & 0 \le n \le N-1 \\ 0, & \text{otherwise} \end{cases} \tag{7.2.9}$$

is related to the original sequence $x(n)$ by a circular shift. This relationship is illustrated in Fig. 7.2.1 for $N = 4$.

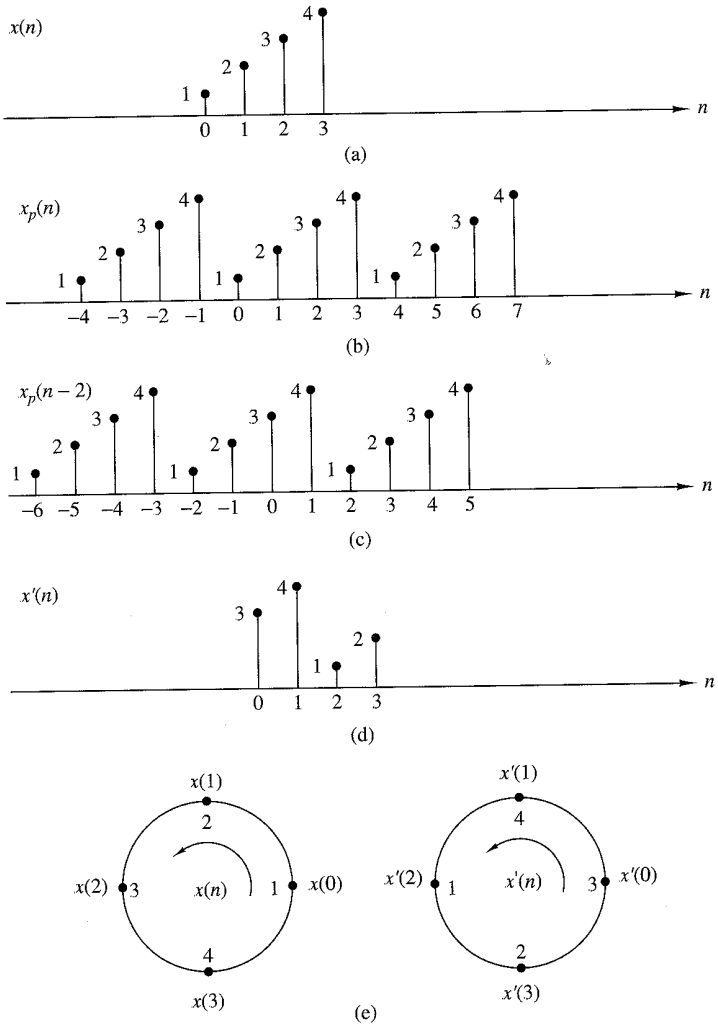



**Figure 7.2.1** Circular shift of a sequence.

In general, the circular shift of the sequence can be represented as the index modulo $N$. Thus we can write

$$\begin{aligned} x'(n) &= x(n-k, \text{modulo } N) \\ &\equiv x((n-k))_N \end{aligned} \tag{7.2.10}$$

For example, if $k = 2$ and $N = 4$, we have

$$x'(n) = x((n-2))_4$$

which implies that

$$\begin{aligned} x'(0) &= x((-2))_4 = x(2) \\ x'(1) &= x((-1))_4 = x(3) \\ x'(2) &= x((0))_4 = x(0) \\ x'(3) &= x((1))_4 = x(1) \end{aligned}$$

Hence $x'(n)$ is simply $x(n)$ shifted circularly by two units in time, where the counterclockwise direction has been arbitrarily selected as the positive direction. Thus we conclude that a circular shift of an $N$-point sequence is equivalent to a linear shift of its periodic extension, and vice versa.

The inherent periodicity resulting from the arrangement of the $N$-point sequence on the circumference of a circle dictates a different definition of even and odd symmetry, and time reversal of a sequence.

An $N$-point sequence is called circularly *even* if it is symmetric about the point zero on the circle. This implies that

$$x(N-n) = x(n), \qquad 1 \le n \le N-1 \tag{7.2.11}$$

An $N$-point sequence is called circularly *odd* if it is antisymmetric about the point zero on the circle. This implies that

$$x(N-n) = -x(n), \qquad 1 \le n \le N-1 \tag{7.2.12}$$

The time reversal of an $N$-point sequence is attained by reversing its samples about the point zero on the circle. Thus the sequence $x((-n))_N$ is simply given as

$$x((-n))_N = x(N-n), \qquad 0 \le n \le N-1 \tag{7.2.13}$$

This time reversal is equivalent to plotting $x(n)$ in a clockwise direction on a circle.

An equivalent definition of even and odd sequences for the associated periodic sequence $x_p(n)$ is given as follows

$$\begin{aligned} \text{even:} \quad & x_p(n) = x_p(-n) = x_p(N-n) \\ \text{odd:} \quad & x_p(n) = -x_p(-n) = -x_p(N-n) \end{aligned} \tag{7.2.14}$$

If the periodic sequence is complex valued, we have

$$\begin{aligned} &\text{conjugate even:} \quad x_p(n) = x_p^*(N-n) \\ &\text{conjugate odd:} \quad x_p(n) = -x_p^*(N-n) \end{aligned} \tag{7.2.15}$$

These relationships suggest that we decompose the sequence $x_p(n)$ as

$$x_p(n) = x_{pe}(n) + x_{po}(n) \tag{7.2.16}$$

where

$$\begin{aligned} x_{pe}(n) &= \frac{1}{2}[x_p(n) + x_p^*(N-n)] \\ x_{po}(n) &= \frac{1}{2}[x_p(n) - x_p^*(N-n)] \end{aligned} \tag{7.2.17}$$

**Symmetry properties of the DFT.** The symmetry properties for the DFT can be obtained by applying the methodology previously used for the Fourier transform. Let us assume that the $N$-point sequence $x(n)$ and its DFT are both complex valued. Then the sequences can be expressed as

$$x(n) = x_R(n) + jx_I(n), \qquad 0 \le n \le N-1 \tag{7.2.18}$$

$$X(k) = X_R(k) + jX_I(k), \qquad 0 \le k \le N-1 \tag{7.2.19}$$

By substituting (7.2.18) into the expression for the DFT given by (7.2.1), we obtain

$$X_R(k) = \sum_{n=0}^{N-1} \left[ x_R(n) \cos \frac{2\pi kn}{N} + x_I(n) \sin \frac{2\pi kn}{N} \right] \tag{7.2.20}$$

$$X_I(k) = -\sum_{n=0}^{N-1} \left[ x_R(n) \sin \frac{2\pi kn}{N} - x_I(n) \cos \frac{2\pi kn}{N} \right] \tag{7.2.21}$$

Similarly, by substituting (7.2.19) into the expression for the IDFT given by (7.2.2), we obtain

$$x_R(n) = \frac{1}{N} \sum_{k=0}^{N-1} \left[ X_R(k) \cos \frac{2\pi kn}{N} - X_I(k) \sin \frac{2\pi kn}{N} \right] \tag{7.2.22}$$

$$x_I(n) = \frac{1}{N} \sum_{k=0}^{N-1} \left[ X_R(k) \sin \frac{2\pi kn}{N} + X_I(k) \cos \frac{2\pi kn}{N} \right] \tag{7.2.23}$$

**Real-valued sequences.** If the sequence $x(n)$ is real, it follows directly from (7.2.1) that

$$X(N-k) = X^*(k) = X(-k) \tag{7.2.24}$$

Consequently, $|X(N-k)| = |X(k)|$ and $\angle X(N-k) = -\angle X(k)$. Furthermore, $x_I(n) = 0$ and therefore $x(n)$ can be determined from (7.2.22), which is another form for the IDFT.

**Real and even sequences.** If $x(n)$ is real and even, that is,

$$x(n) = x(N-n), \qquad 0 \le n \le N-1$$

then (7.2.21) yields $X_I(k) = 0$. Hence the DFT reduces to

$$X(k) = \sum_{n=0}^{N-1} x(n) \cos \frac{2\pi kn}{N}, \qquad 0 \le k \le N-1 \tag{7.2.25}$$

which is itself real valued and even. Furthermore, since $X_I(k) = 0$, the IDFT reduces to

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) \cos \frac{2\pi kn}{N}, \qquad 0 \le n \le N-1 \tag{7.2.26}$$

**Real and odd sequences.** If $x(n)$ is real and odd, that is,

$$x(n) = -x(N-n), \qquad 0 \le n \le N-1$$

then (7.2.20) yields $X_R(k) = 0$. Hence

$$X(k) = -j \sum_{n=0}^{N-1} x(n) \sin \frac{2\pi kn}{N}, \qquad 0 \le k \le N-1 \tag{7.2.27}$$

which is purely imaginary and odd. Since $X_R(k) = 0$, the IDFT reduces to

$$x(n) = j\frac{1}{N} \sum_{k=0}^{N-1} X(k) \sin \frac{2\pi kn}{N}, \qquad 0 \le n \le N-1 \tag{7.2.28}$$

**Purely imaginary sequences.** In this case, $x(n) = jx_I(n)$. Consequently, (7.2.20) and (7.2.21) reduce to

$$X_R(k) = \sum_{n=0}^{N-1} x_I(n) \sin \frac{2\pi kn}{N} \tag{7.2.29}$$

$$X_I(k) = \sum_{n=0}^{N-1} x_I(n) \cos \frac{2\pi kn}{N} \tag{7.2.30}$$

We observe that $X_R(k)$ is odd and $X_I(k)$ is even.

If $x_I(n)$ is odd, then $X_I(k) = 0$ and hence $X(k)$ is purely real. On the other hand, if $x_I(n)$ is even, then $X_R(k) = 0$ and hence $X(k)$ is purely imaginary.

The symmetry properties given above may be summarized as follows:

$$\begin{aligned} x(n) &= x_R^e(n) + x_R^o(n) + jx_I^e(n) + jx_I^o(n) \\ X(k) &= X_R^e(k) + X_R^o(k) + jX_I^e(k) + jX_I^o(k) \end{aligned} \tag{7.2.31}$$

All the symmetry properties of the DFT can easily be deduced from (7.2.31). For example, the DFT of the sequence

$$x_{pe}(n) = \tfrac{1}{2}[x_p(n) + x_p^*(N-n)]$$

is

$$X_R(k) = X_R^e(k) + X_R^o(k)$$

The symmetry properties of the DFT are summarized in Table 7.1. Exploitation of these properties for the efficient computation of the DFT of special sequences is considered in some of the problems at the end of the chapter.

**TABLE 7.1** Symmetry Properties of the DFT

| $N$-Point Sequence $x(n)$, $0 \le n \le N-1$ | $N$-Point DFT |
|---|---|
| $x(n)$ | $X(k)$ |
| $x^*(n)$ | $X^*(N-k)$ |
| $x^*(N-n)$ | $X^*(k)$ |
| $x_R(n)$ | $X_{ce}(k) = \frac{1}{2}[X(k) + X^*(N-k)]$ |
| $jX_I(n)$ | $X_{co}(k) = \frac{1}{2}[X(k) - X^*(N-k)]$ |
| $x_{ce}(n) = \frac{1}{2}[x(n) + x^*(N-n)]$ | $X_R(k)$ |
| $x_{co}(n) = \frac{1}{2}[x(n) - x^*(N-n)]$ | $jX_I(k)$ |
| Real Signals | |
| Any real signal $x(n)$ | $X(k) = X^*(N-k)$ |
| | $X_R(k) = X_R(N-k)$ |
| | $X_I(k) = -X_I(N-k)$ |
| | $\lvert X(k)\rvert = \lvert X(N-k)\rvert$ |
| | $\angle X(k) = -\angle X(N-k)$ |
| $x_{ce}(n) = \frac{1}{2}[x(n) + x(N-n)]$ | $X_R(k)$ |
| $x_{co}(n) = \frac{1}{2}[x(n) - x(N-n)]$ | $jX_I(k)$ |

### 7.2.2 Multiplication of Two DFTs and Circular Convolution

Suppose that we have two finite-duration sequences of length $N$, $x_1(n)$ and $x_2(n)$. Their respective $N$-point DFTs are

$$X_1(k) = \sum_{n=0}^{N-1} x_1(n)e^{-j2\pi nk/N}, \qquad k = 0, 1, \ldots, N-1 \tag{7.2.32}$$

$$X_2(k) = \sum_{n=0}^{N-1} x_2(n)e^{-j2\pi nk/N}, \qquad k = 0, 1, \ldots, N-1 \tag{7.2.33}$$

If we multiply the two DFTs together, the result is a DFT, say $X_3(k)$, of a sequence $x_3(n)$ of length $N$. Let us determine the relationship between $x_3(n)$ and the sequences $x_1(n)$ and $x_2(n)$.

We have

$$X_3(k) = X_1(k)X_2(k), \qquad k = 0, 1, \ldots, N-1 \tag{7.2.34}$$

The IDFT of $\{X_3(k)\}$ is

$$\begin{aligned} x_3(m) &= \frac{1}{N}\sum_{k=0}^{N-1} X_3(k)e^{j2\pi km/N} \\ &= \frac{1}{N}\sum_{k=0}^{N-1} X_1(k)X_2(k)e^{j2\pi km/N} \end{aligned} \tag{7.2.35}$$

Suppose that we substitute for $X_1(k)$ and $X_2(k)$ in (7.2.35) using the DFTs given in (7.2.32) and (7.2.33). Thus we obtain

$$\begin{aligned} x_3(m) &= \frac{1}{N}\sum_{k=0}^{N-1}\left[\sum_{n=0}^{N-1} x_1(n)e^{-j2\pi kn/N}\right]\left[\sum_{l=0}^{N-1} x_2(l)e^{-j2\pi kl/N}\right]e^{j2\pi km/N} \\ &= \frac{1}{N}\sum_{n=0}^{N-1} x_1(n)\sum_{l=0}^{N-1} x_2(l)\left[\sum_{k=0}^{N-1} e^{j2\pi k(m-n-l)/N}\right] \end{aligned} \tag{7.2.36}$$

The inner sum in the brackets in (7.2.36) has the form

$$\sum_{k=0}^{N-1} a^k = \begin{cases} N, & a = 1 \\ \dfrac{1-a^N}{1-a}, & a \neq 1 \end{cases} \tag{7.2.37}$$

where $a$ is defined as

$$a = e^{j2\pi(m-n-l)/N}$$

We observe that $a = 1$ when $m - n - l$ is a multiple of $N$. On the other hand, $a^N = 1$ for any value of $a \neq 0$. Consequently, (7.2.37) reduces to

$$\sum_{k=0}^{N-1} a^k = \begin{cases} N, & l = m - n + pN = ((m-n))_N, \quad p \text{ an integer} \\ 0, & \text{otherwise} \end{cases} \tag{7.2.38}$$

If we substitute the result in (7.2.38) into (7.2.36), we obtain the desired expression for $x_3(m)$ in the form

$$x_3(m) = \sum_{n=0}^{N-1} x_1(n) x_2((m-n))_N, \qquad m = 0, 1, \ldots, N-1 \tag{7.2.39}$$

The expression in (7.2.39) has the form of a convolution sum. However, it is not the ordinary linear convolution that was introduced in Chapter 2, which relates the output sequence $y(n)$ of a linear system to the input sequence $x(n)$ and the impulse response $h(n)$. Instead, the convolution sum in (7.2.39) involves the index $((m-n))_N$ and is called *circular convolution*. Thus we conclude that multiplication of the DFTs of two sequences is equivalent to the circular convolution of the two sequences in the time domain.

The following example illustrates the operations involved in circular convolution.

### EXAMPLE 7.2.1

Perform the circular convolution of the following two sequences:

$$x_1(n) = \{\underset{\uparrow}{2}, 1, 2, 1\}$$

$$x_2(n) = \{\underset{\uparrow}{1}, 2, 3, 4\}$$

**Solution.** Each sequence consists of four nonzero points. For the purposes of illustrating the operations involved in circular convolution, it is desirable to graph each sequence as points on a circle. Thus the sequences $x_1(n)$ and $x_2(n)$ are graphed as illustrated in Fig. 7.2.2(a). We note that the sequences are graphed in a counterclockwise direction on a circle. This establishes the reference direction in rotating one of the sequences relative to the other.

Now, $x_3(m)$ is obtained by circularly convolving $x_1(n)$ with $x_2(n)$ as specified by (7.2.39). Beginning with $m = 0$ we have

$$x_3(0) = \sum_{n=0}^{3} x_1(n) x_2((-n))N$$

$x_2((-n))_4$ is simply the sequence $x_2(n)$ folded and graphed on a circle as illustrated in Fig. 7.2.2(b). In other words, the folded sequence is simply $x_2(n)$ graphed in a clockwise direction.

The product sequence is obtained by multiplying $x_1(n)$ with $x_2((-n))_4$, point by point. This sequence is also illustrated in Fig. 7.2.2(b). Finally, we sum the values in the product sequence to obtain

$$x_3(0) = 14$$

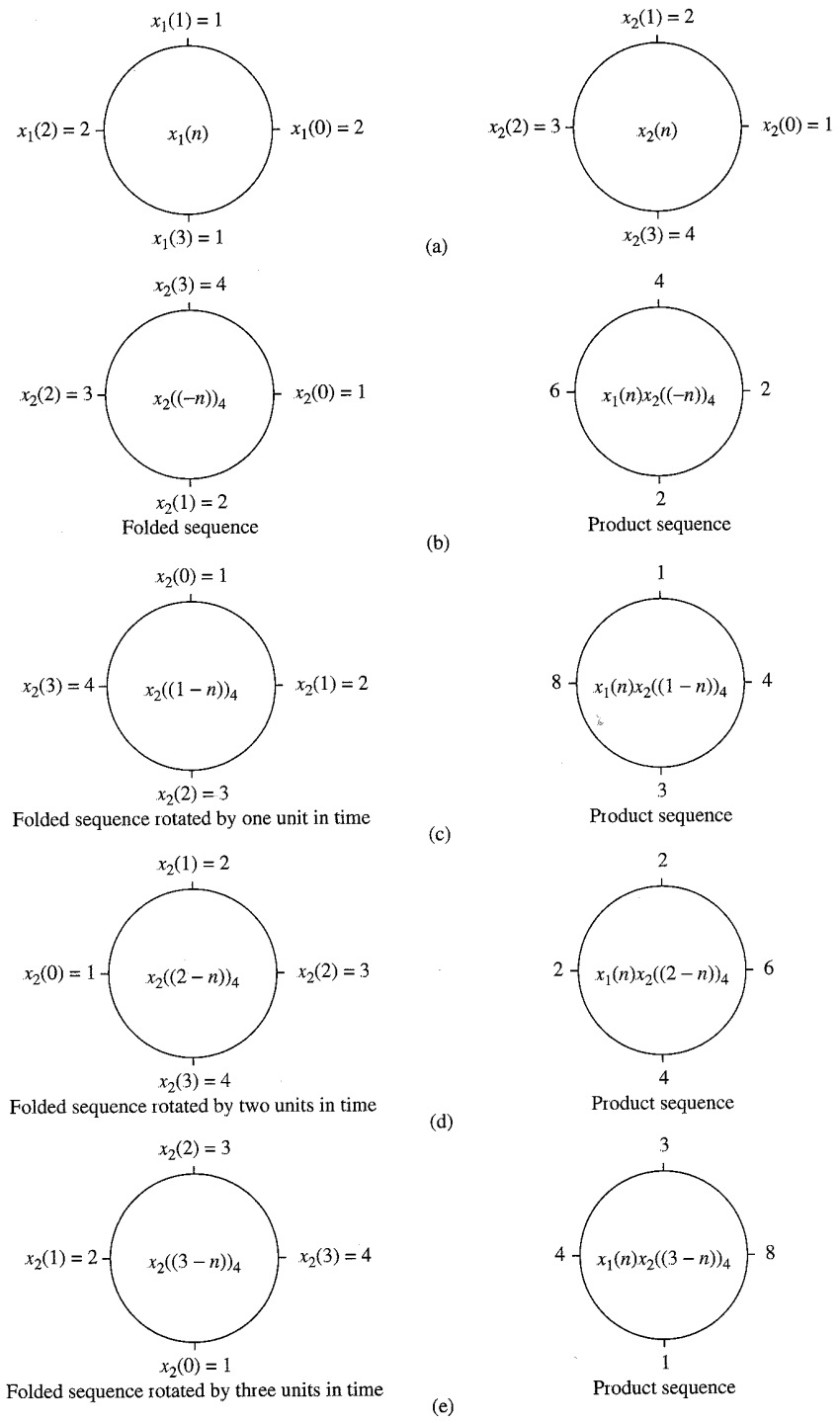



**Figure 7.2.2** Circular convolution of two sequences.

For $m = 1$ we have

$$x_3(1) = \sum_{n=0}^{3} x_1(n)x_2((1-n))_4$$

It is easily verified that $x_2((1-n))_4$ is simply the sequence $x_2((-n))_4$ rotated counterclockwise by one unit in time as illustrated in Fig. 7.2.2(c). This rotated sequence multiplies $x_1(n)$ to yield the product sequence, also illustrated in Fig. 7.2.2(c). Finally, we sum the values in the product sequence to obtain $x_3(1)$. Thus

$$x_3(1) = 16$$

For $m = 2$ we have

$$x_3(2) = \sum_{n=0}^{3} x_1(n)x_2((2-n))_4$$

Now $x_2((2-n))_4$ is the folded sequence in Fig. 7.2.2(b) rotated two units of time in the counterclockwise direction. The resultant sequence is illustrated in Fig. 7.2.2(d) along with the product sequence $x_1(n)x_2((2-n))_4$. By summing the four terms in the product sequence, we obtain

$$x_3(2) = 14$$

For $m = 3$ we have

$$x_3(3) = \sum_{n=0}^{3} x_1(n)x_2((3-n))_4$$

The folded sequence $x_2((-n))_4$ is now rotated by three units in time to yield $x_2((3-n))_4$ and the resultant sequence is multiplied by $x_1(n)$ to yield the product sequence as illustrated in Fig. 7.2.2(e). The sum of the values in the product sequence is

$$x_3(3) = 16$$

We observe that if the computation above is continued beyond $m = 3$, we simply repeat the sequence of four values obtained above. Therefore, the circular convolution of the two sequences $x_1(n)$ and $x_2(n)$ yields the sequence

$$x_3(n) = \{\underset{\uparrow}{14}, 16, 14, 16\}$$

---

From this example, we observe that circular convolution involves basically the same four steps as the ordinary *linear convolution* introduced in Chapter 2: *folding* (time reversing) one sequence, *shifting* the folded sequence, *multiplying* the two sequences to obtain a product sequence, and finally, *summing* the values of the product sequence. The basic difference between these two types of convolution is that, in circular convolution, the folding and shifting (rotating) operations are performed in a circular fashion by computing the index of one of the sequences modulo $N$. In linear convolution, there is no modulo $N$ operation.

The reader can easily show from our previous development that either one of the two sequences may be folded and rotated without changing the result of the circular convolution. Thus

$$x_3(m) = \sum_{n=0}^{N-1} x_2(n)x_1((m-n))_N, \qquad m = 0, 1, \ldots, N-1 \tag{7.2.40}$$

The following example serves to illustrate the computation of $x_3(n)$ by means of the DFT and IDFT.

EXAMPLE 7.2.2

By means of the DFT and IDFT, determine the sequence $x_3(n)$ corresponding to the circular convolution of the sequences $x_1(n)$ and $x_2(n)$ given in Example 7.2.1.

**Solution.** First we compute the DFTs of $x_1(n)$ and $x_2(n)$. The four-point DFT of $x_1(n)$ is

$$X_1(k) = \sum_{n=0}^{3} x_1(n)e^{-j2\pi nk/4}, \qquad k = 0, 1, 2, 3$$
$$= 2 + e^{-j\pi k/2} + 2e^{-j\pi k} + e^{-j3\pi k/2}$$

Thus

$$X_1(0) = 6, \qquad X_1(1) = 0, \qquad X_1(2) = 2, \qquad X_1(3) = 0$$

The DFT of $x_2(n)$ is

$$X_2(k) = \sum_{n=0}^{3} x_2(n)e^{-j2\pi nk/4}, \qquad k = 0, 1, 2, 3$$
$$= 1 + 2e^{-j\pi k/2} + 3e^{-j\pi k} + 4e^{-j3\pi k/2}$$

Thus

$$X_2(0) = 10, \qquad X_2(1) = -2 + j2, \qquad X_2(2) = -2, \qquad X_2(3) = -2 - j2$$

When we multiply the two DFTs, we obtain the product

$$X_3(k) = X_1(k)X_2(k)$$

or, equivalently,

$$X_3(0) = 60, \qquad X_3(1) = 0, \qquad X_3(2) = -4, \qquad X_3(3) = 0$$

Now, the IDFT of $X_3(k)$ is

$$x_3(n) = \sum_{k=0}^{3} X_3(k)e^{j2\pi nk/4}, \qquad n = 0, 1, 2, 3$$
$$= \frac{1}{4}(60 - 4e^{j\pi n})$$

Thus

$$x_3(0) = 14, \qquad x_3(1) = 16, \qquad x_3(2) = 14, \qquad x_3(3) = 16$$

which is the result obtained in Example 7.2.1 from circular convolution.

---

We conclude this section by formally stating this important property of the DFT.

**Circular convolution.** If

$$x_1(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X_1(k)$$

and

$$x_2(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X_2(k)$$

then

$$x_1(n) \circledN x_2(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X_1(k)X_2(k) \tag{7.2.41}$$

where $x_1(n) \circledN x_2(n)$ denotes the circular convolution of the sequence $x_1(n)$ and $x_2(n)$.

### 7.2.3 Additional DFT Properties

**Time reversal of a sequence.** If

$$x(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X(k)$$

then

$$x((-n))_N = x(N-n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X((-k))_N = X(N-k) \tag{7.2.42}$$

Hence reversing the $N$-point sequence in time is equivalent to reversing the DFT values. Time reversal of a sequence $x(n)$ is illustrated in Fig. 7.2.3.

*Proof* From the definition of the DFT in (7.2.1) we have

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

If we change the index from $n$ to $m = N - n$, then

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

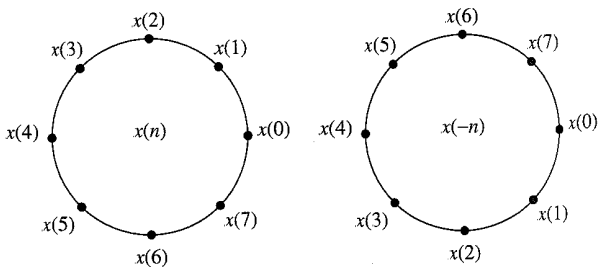



**Figure 7.2.3** Time reversal of a sequence.

$$= \sum_{m=0}^{N-1} x(m)e^{j2\pi km/N}$$

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

We note that $X(N-k) = X((-k))_N$, $0 \le k \le N-1$.

**Circular time shift of a sequence.** If

$$x(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X(k)$$

then

$$x((n-l))_N \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X(k)e^{-j2\pi kl/N} \tag{7.2.43}$$

*Proof* From the definition of the DFT we have

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

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

$$+ \sum_{n=l}^{N-1} x(n-l)e^{-j\pi kn/N}$$

But $x((n-l))_N = x(N-l+n)$. Consequently,

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

$$= \sum_{m=N-l}^{N-1} x(m)e^{-j2\pi k(m+l)/N}$$

Furthermore,

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

Therefore,

$$\text{DFT}\{x((n-l))\} = \sum_{m=0}^{N-1} x(m)e^{-j2\pi k(m+l)/N}$$

$$= X(k)e^{-j2\pi kl/N}$$

**Circular frequency shift.** If

$$x(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X(k)$$

then

$$x(n)e^{j2\pi ln/N} \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X((k-l))_N \tag{7.2.44}$$

Hence, the multiplication of the sequence $x(n)$ with the complex exponential sequence $e^{j2\pi kn/N}$ is equivalent to the circular shift of the DFT by $l$ units in frequency. This is the dual to the circular time-shifting property and its proof is similar to that of the latter.

**Complex-conjugate properties.** If

$$x(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X(k)$$

then

$$x^*(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X^*((-k))_N = X^*(N-k) \tag{7.2.45}$$

The proof of this property is left as an exercise for the reader. The IDFT of $X^*(k)$ is

$$\frac{1}{N}\sum_{k=0}^{N-1} X^*(k)e^{j2\pi kn/N} = \left[\frac{1}{N}\sum_{k=0}^{N-1} X(k)e^{j2\pi k(N-n)/N}\right]$$

Therefore,

$$x^*((-n))_N = x^*(N-n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X^*(k) \tag{7.2.46}$$

**Circular correlation.** In general, for complex-valued sequences $x(n)$ and $y(n)$, if

$$x(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X(k)$$

and

$$y(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} Y(k)$$

then

$$\tilde{r}_{xy}(l) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} \tilde{R}_{xy}(k) = X(k)Y^*(k) \tag{7.2.47}$$

where $\tilde{r}_{xy}(l)$ is the (unnormalized) circular crosscorrelation sequence, defined as

$$\tilde{r}xy(l) = \sum_{n=0}^{N-1} x(n)y^*((n-l))N$$

*Proof* We can write $\tilde{r}_{xy}(l)$ as the circular convolution of $x(n)$ with $y^*(-n)$, that is,

$$\tilde{r}_{xy}(l) = x(l) \circledN y^*(-l)$$

Then, with the aid of the properties in (7.2.41) and (7.2.46), the $N$-point DFT of $\tilde{r}_{xy}(l)$ is

$$\tilde{R}_{xy}(k) = X(k)Y^*(k)$$

In the special case where $y(n) = x(n)$, we have the corresponding expression for the circular autocorrelation of $x(n)$,

$$\tilde{r}_{xx}(l) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} \tilde{R}_{xx}(k) = |X(k)|^2 \tag{7.2.48}$$

**Multiplication of two sequences.** If

$$x_1(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X_1(k)$$

and

$$x_2(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X_2(k)$$

then

$$x_1(n)x_2(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} \frac{1}{N} X_1(k) \circledN X_2(k) \tag{7.2.49}$$

This property is the dual of (7.2.41). Its proof follows simply by interchanging the roles of time and frequency in the expression for the circular convolution of two sequences.

**Parseval's Theorem.** For complex-valued sequences $x(n)$ and $y(n)$, in general, if

$$x(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X(k)$$

and

$$y(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} Y(k)$$

then

$$\sum_{n=0}^{N-1} x(n)y^*(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)Y^*(k) \tag{7.2.50}$$

*Proof* The property follows immediately from the circular correlation property in (7.2.47). We have

$$\sum_{n=0}^{N-1} x(n)y^*(n) = \tilde{r}_{xy}(0)$$

**TABLE 7.2** Properties of the DFT

| Property | Time Domain | Frequency Domain |
|---|---|---|
| Notation | $x(n), y(n)$ | $X(k), Y(k)$ |
| Periodicity | $x(n) = x(n+N)$ | $X(k) = X(k+N)$ |
| Linearity | $a_1x_1(n) + a_2x_2(n)$ | $a_1X_1(k) + a_2X_2(k)$ |
| Time reversal | $x(N-n)$ | $X(N-k)$ |
| Circular time shift | $x((n-l))_N$ | $X(k)e^{-j2\pi kl/N}$ |
| Circular frequency shift | $x(n)e^{j2\pi ln/N}$ | $X((k-l))_N$ |
| Complex conjugate | $x^*(n)$ | $X^*(N-k)$ |
| Circular convolution | $x_1(n) \circledN x_2(n)$ | $X_1(k)X_2(k)$ |
| Circular correlation | $x(n) \circledN y^*(-n)$ | $X(k)Y^*(k)$ |
| Multiplication of two sequences | $x_1(n)x_2(n)$ | $\frac{1}{N}X_1(k) \circledN X_2(k)$ |
| Parseval's theorem | $\sum_{n=0}^{N-1} x(n)y^*(n)$ | $\frac{1}{N}\sum_{k=0}^{N-1} X(k)Y^*(k)$ |

and

$$\tilde{r}_{xy}(l) = \frac{1}{N}\sum_{k=0}^{N-1} \tilde{R}_{xy}(k)e^{j2\pi kl/N}$$

$$= \frac{1}{N}\sum_{k=0}^{N-1} X(k)Y^*(k)e^{j2\pi kl/N}$$

Hence (7.2.50) follows by evaluating the IDFT at $l = 0$.

The expression in (7.2.50) is the general form of Parseval's theorem. In the special case where $y(n) = x(n)$, (7.2.50) reduces to

$$\sum_{n=0}^{N-1} |x(n)|^2 = \frac{1}{N}\sum_{k=0}^{N-1} |X(k)|^2 \tag{7.2.51}$$

which expresses the energy in the finite-duration sequence $x(n)$ in terms of the frequency components $\{X(k)\}$.

The properties of the DFT given above are summarized in Table 7.2.

## 7.3 Linear Filtering Methods Based on the DFT

Since the DFT provides a discrete frequency representation of a finite-duration sequence in the frequency domain, it is interesting to explore its use as a computational tool for linear system analysis and, especially, for linear filtering. We have already established that a system with frequency response $H(\omega)$, when excited with an input

signal that has a spectrum $X(\omega)$, possesses an output spectrum $Y(\omega) = X(\omega)H(\omega)$. The output sequence $y(n)$ is determined from its spectrum via the inverse Fourier transform. Computationally, the problem with this frequency-domain approach is that $X(\omega)$, $H(\omega)$, and $Y(\omega)$ are functions of the continuous variable $\omega$. As a consequence, the computations cannot be done on a digital computer, since the computer can only store and perform computations on quantities at discrete frequencies.

On the other hand, the DFT does lend itself to computation on a digital computer. In the discussion that follows, we describe how the DFT can be used to perform linear filtering in the frequency domain. In particular, we present a computational procedure that serves as an alternative to time-domain convolution. In fact, the frequency-domain approach based on the DFT is computationally more efficient than time-domain convolution due to the existence of efficient algorithms for computing the DFT. These algorithms, which are described in Chapter 8, are collectively called fast Fourier transform (FFT) algorithms.

### 7.3.1 Use of the DFT in Linear Filtering

In the preceding section it was demonstrated that the product of two DFTs is equivalent to the circular convolution of the corresponding time-domain sequences. Unfortunately, circular convolution is of no use to us if our objective is to determine the output of a linear filter to a given input sequence. In this case we seek a frequency-domain methodology equivalent to linear convolution.

Suppose that we have a finite-duration sequence $x(n)$ of length $L$ which excites an FIR filter of length $M$. Without loss of generality, let

$$x(n) = 0, \qquad n < 0 \text{ and } n \geq L$$

$$h(n) = 0, \qquad n < 0 \text{ and } n \geq M$$

where $h(n)$ is the impulse response of the FIR filter.

The output sequence $y(n)$ of the FIR filter can be expressed in the time domain as the convolution of $x(n)$ and $h(n)$, that is

$$y(n) = \sum_{k=0}^{M-1} h(k)x(n-k) \tag{7.3.1}$$

Since $h(n)$ and $x(n)$ are finite-duration sequences, their convolution is also finite in duration. In fact, the duration of $y(n)$ is $L + M - 1$.

The frequency-domain equivalent to (7.3.1) is

$$Y(\omega) = X(\omega)H(\omega) \tag{7.3.2}$$

If the sequence $y(n)$ is to be represented uniquely in the frequency domain by samples of its spectrum $Y(\omega)$ at a set of discrete frequencies, the number of distinct samples must equal or exceed $L+M-1$. Therefore, a DFT of size $N \geq L+M-1$ is required to represent $\{y(n)\}$ in the frequency domain.

Now if

$$Y(k) \equiv Y(\omega)|_{\omega=2\pi k/N}, \qquad k = 0, 1, \ldots, N-1$$
$$= X(\omega)H(\omega)|_{\omega=2\pi k/N}, \qquad k = 0, 1, \ldots, N-1$$

then

$$Y(k) = X(k)H(k), \qquad k = 0, 1, \ldots, N-1 \tag{7.3.3}$$

where $\{X(k)\}$ and $\{H(k)\}$ are the $N$-point DFTs of the corresponding sequences $x(n)$ and $h(n)$, respectively. Since the sequences $x(n)$ and $h(n)$ have a duration less than $N$, we simply pad these sequences with zeros to increase their length to $N$. This increase in the size of the sequences does not alter their spectra $X(\omega)$ and $H(\omega)$, which are continuous spectra, since the sequences are aperiodic. However, by sampling their spectra at $N$ equally spaced points in frequency (computing the $N$-point DFTs), we have increased the number of samples that represent these sequences in the frequency domain beyond the minimum number ($L$ or $M$, respectively).

Since the $(N = L + M - 1)$-point DFT of the output sequence $y(n)$ is sufficient to represent $y(n)$ in the frequency domain, it follows that the multiplication of the $N$-point DFTs $X(k)$ and $H(k)$, according to (7.3.3), followed by the computation of the $N$-point IDFT, must yield the sequence $\{y(n)\}$. In turn, this implies that the $N$-point circular convolution of $x(n)$ with $h(n)$ must be equivalent to the linear convolution of $x(n)$ with $h(n)$. In other words, by increasing the length of the sequences $x(n)$ and $h(n)$ to $N$ points (by appending zeros), and then circularly convolving the resulting sequences, we obtain the same result as would have been obtained with linear convolution. Thus with zero padding, the DFT can be used to perform linear filtering.

The following example illustrates the methodology in the use of the DFT in linear filtering.

### EXAMPLE 7.3.1

By means of the DFT and IDFT, determine the response of the FIR filter with impulse response

$$h(n) = \{\underset{\uparrow}{1}, 2, 3\}$$

to the input sequence

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

**Solution.** The input sequence has length $L = 4$ and the impulse response has length $M = 3$. Linear convolution of these two sequences produces a sequence of length $N = 6$. Consequently, the size of the DFTs must be at least six.

For simplicity we compute eight-point DFTs. We should also mention that the efficient computation of the DFT via the fast Fourier transform (FFT) algorithm is usually performed for a length $N$ that is a power of 2. Hence the eight-point DFT of $x(n)$ is

$$X(k) = \sum_{n=0}^{7} x(n)e^{-j2\pi kn/8}$$
$$= 1 + 2e^{-j\pi k/4} + 2e^{-j\pi k/2} + e^{-j3\pi k/4}, \qquad k = 0, 1, \ldots, 7$$

This computation yields

$$X(0) = 6, \qquad X(1) = \frac{2+\sqrt{2}}{2} - j\left(\frac{4+3\sqrt{2}}{2}\right)$$

$$X(2) = -1 - j, \qquad X(3) = \frac{2-\sqrt{2}}{2} + j\left(\frac{4-3\sqrt{2}}{2}\right)$$

$$X(4) = 0, \qquad X(5) = \frac{2-\sqrt{2}}{2} - j\left(\frac{4-3\sqrt{2}}{2}\right)$$

$$X(6) = -1 + j, \qquad X(7) = \frac{2+\sqrt{2}}{2} + j\left(\frac{4+3\sqrt{2}}{2}\right)$$

The eight-point DFT of $h(n)$ is

$$\begin{aligned} H(k) &= \sum_{n=0}^{7} h(n)e^{-j2\pi kn/8} \\ &= 1 + 2e^{-j\pi k/4} + 3e^{-j\pi k/2} \end{aligned}$$

Hence

$$H(0) = 6, \qquad H(1) = 1 + \sqrt{2} - j\left(3+\sqrt{2}\right), \qquad H(2) = -2 - j2$$

$$H(3) = 1 - \sqrt{2} + j\left(3-\sqrt{2}\right), \qquad H(4) = 2$$

$$H(5) = 1 - \sqrt{2} - j\left(3-\sqrt{2}\right), \qquad H(6) = -2 + j2$$

$$H(7) = 1 + \sqrt{2} + j\left(3+\sqrt{2}\right)$$

The product of these two DFTs yields $Y(k)$, which is

$$Y(0) = 36, \qquad Y(1) = -14.07 - j17.48, \qquad Y(2) = j4, \qquad Y(3) = 0.07 + j0.515$$

$$Y(4) = 0, \qquad Y(5) = 0.07 - j0.515, \qquad Y(6) = -j4, \qquad Y(7) = -14.07 + j17.48$$

Finally, the eight-point IDFT is

$$y(n) = \sum_{k=0}^{7} Y(k)e^{j2\pi kn/8}, \qquad n = 0, 1, \ldots, 7$$

This computation yields the result

$$y(n) = \{\underset{\uparrow}{1}, 4, 9, 11, 8, 3, 0, 0\}$$

We observe that the first six values of $y(n)$ constitute the set of desired output values. The last two values are zero because we used an eight-point DFT and IDFT, when, in fact, the minimum number of points required is six.

Although the multiplication of two DFTs corresponds to circular convolution in the time domain, we have observed that padding the sequences $x(n)$ and $h(n)$ with a sufficient number of zeros forces the circular convolution to yield the same output sequence as linear convolution. In the case of the FIR filtering problem in Example 7.3.1, it is a simple matter to demonstrate that the six-point circular convolution of the sequences

$$h(n) = \{\underset{\uparrow}{1}, 2, 3, 0, 0, 0\} \tag{7.3.4}$$

$$x(n) = \{\underset{\uparrow}{1}, 2, 2, 1, 0, 0\} \tag{7.3.5}$$

results in the output sequence

$$y(n) = \{\underset{\uparrow}{1}, 4, 9, 11, 8, 3\} \tag{7.3.6}$$

which is the same sequence obtained from linear convolution.

It is important for us to understand the aliasing that results in the time domain when the size of the DFTs is smaller than $L + M - 1$. The following example focuses on the aliasing problem.

### EXAMPLE 7.3.2

Determine the sequence $y(n)$ that results from the use of four-point DFTs in Example 7.3.1.

**Solution.** The four-point DFT of $h(n)$ is

$$H(k) = \sum_{n=0}^{3} h(n)e^{-j2\pi kn/4}$$

$$H(k) = 1 + 2e^{-j\pi k/2} + 3e^{-jk\pi}, \qquad k = 0, 1, 2, 3$$

Hence

$$H(0) = 6, \qquad H(1) = -2 - j2, \qquad H(2) = 2, \qquad H(3) = -2 + j2$$

The four-point DFT of $x(n)$ is

$$X(k) = 1 + 2e^{-j\pi k/2} + 2e^{-j\pi k} + 1e^{-j3\pi k/2}, \qquad k = 0, 1, 2, 3$$

Hence

$$X(0) = 6, \qquad X(1) = -1 - j, \qquad X(2) = 0, \qquad X(3) = -1 + j$$

The product of these two four-point DFTs is

$$\hat{Y}(0) = 36, \qquad \hat{Y}(1) = j4, \qquad \hat{Y}(2) = 0, \qquad \hat{Y}(3) = -j4$$

The four-point IDFT yields

$$\hat{y}(n) = \frac{1}{4}\sum_{k=0}^{3} \hat{Y}(k)e^{j2\pi kn/4}, \qquad n = 0, 1, 2, 3$$

$$= \frac{1}{4}(36 + j4e^{j\pi n/2} - j4e^{j3\pi n/2})$$

Therefore,

$$\hat{y}(n) = \{\underset{\uparrow}{9}, 7, 9, 11\}$$

The reader can verify that the four-point circular convolution of $h(n)$ with $x(n)$ yields the same sequence $\hat{y}(n)$.

---

If we compare the result $\hat{y}(n)$, obtained from four-point DFTs, with the sequence $y(n)$ obtained from the use of eight-point (or six-point) DFTs, the time-domain aliasing effects derived in Section 7.2.2 are clearly evident. In particular, $y(4)$ is aliased into $y(0)$ to yield

$$\hat{y}(0) = y(0) + y(4) = 9$$

Similarly, $y(5)$ is aliased into $y(1)$ to yield

$$\hat{y}(1) = y(1) + y(5) = 7$$

All other aliasing has no effect, since $y(n) = 0$ for $n \geq 6$. Consequently, we have

$$\hat{y}(2) = y(2) = 9$$

$$\hat{y}(3) = y(3) = 11$$

Therefore, only the first two points of $\hat{y}(n)$ are corrupted by the effect of aliasing [i.e., $\hat{y}(0) \neq y(0)$ and $\hat{y}(1) \neq y(1)$]. This observation has important ramifications in the discussion of the following section, in which we treat the filtering of long sequences.

### 7.3.2 Filtering of Long Data Sequences

In practical applications involving linear filtering of signals, the input sequence $x(n)$ is often a very long sequence. This is especially true in some real-time signal processing applications concerned with signal monitoring and analysis.

Since linear filtering performed via the DFT involves operations on a block of data, which by necessity must be limited in size due to limited memory of a digital computer, a long input signal sequence must be segmented to fixed-size blocks prior to processing. Since the filtering is linear, successive blocks can be processed one at a time via the DFT, and the output blocks are fitted together to form the overall output signal sequence.

We now describe two methods for linear FIR filtering a long sequence on a block-by-block basis using the DFT. The input sequence is segmented into blocks and each block is processed via the DFT and IDFT to produce a block of output data. The output blocks are fitted together to form an overall output sequence which is identical to the sequence obtained if the long block had been processed via time-domain convolution.

The two methods are called the *overlap-save method* and the *overlap-add method*. For both methods we assume that the FIR filter has duration $M$. The input data sequence is segmented into blocks of $L$ points, where, by assumption, $L >> M$ without loss of generality.

**Overlap-save method.** In this method the size of the input data blocks is $N = L + M - 1$ and the DFTs and IDFT are of length $N$. Each data block consists of the last $M - 1$ data points of the previous data block followed by $L$ new data points to form a data sequence of length $N = L + M - 1$. An $N$-point DFT is computed for each data block. The impulse response of the FIR filter is increased in length by appending $L - 1$ zeros and an $N$-point DFT of the sequence is computed once and stored. The multiplication of the two $N$-point DFTs $\{H(k)\}$ and $\{X_m(k)\}$ for the $m$th block of data yields

$$\hat{Y}_m(k) = H(k)X_m(k), \qquad k = 0, 1, \ldots, N-1 \tag{7.3.7}$$

Then the $N$-point IDFT yields the result

$$\hat{Y}_m(n) = \{\hat{y}_m(0)\hat{y}_m(1) \cdots \hat{y}_m(M-1)\hat{y}_m(M) \cdots \hat{y}_m(N-1)\} \tag{7.3.8}$$

Since the data record is of length $N$, the first $M - 1$ points of $y_m(n)$ are corrupted by aliasing and must be discarded. The last $L$ points of $y_m(n)$ are exactly the same as the result from linear convolution and, as a consequence,

$$\hat{y}_m(n) = y_m(n), n = M, M+1, \ldots, N-1 \tag{7.3.9}$$

To avoid loss of data due to aliasing, the last $M - 1$ points of each data record are saved and these points become the first $M - 1$ data points of the subsequent record, as indicated above. To begin the processing, the first $M - 1$ points of the first record are set to zero. Thus the blocks of data sequences are

$$x_1(n) = \{\underbrace{0, 0, \ldots, 0}_{M-1 \text{ points}}, x(0), x(1), \ldots, x(L-1)\} \tag{7.3.10}$$

$$x_2(n) = \{\underbrace{x(L-M+1), \ldots, x(L-1)}_{M-1 \text{data points from } x_1(n)}, \underbrace{x(L), \ldots, x(2L-1)}_{L \text{ new data points}}\} \tag{7.3.11}$$

$$x_3(n) = \{\underbrace{x(2L-M+1), \ldots, x(2L-1)}_{M-1 \text{ data points from } x_2(n)}, \underbrace{x(2L), \ldots, x(3L-1)}_{L \text{ new data points}}\} \tag{7.3.12}$$

and so forth. The resulting data sequences from the IDFT are given by (7.3.8), where the first $M - 1$ points are discarded due to aliasing and the remaining $L$ points constitute the desired result from linear convolution. This segmentation of the input data and the fitting of the output data blocks together to form the output sequence are graphically illustrated in Fig. 7.3.1.

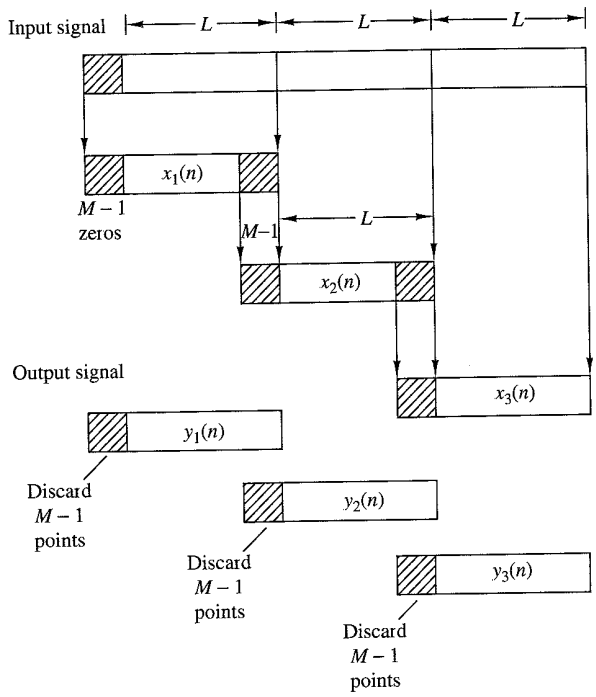



**Figure 7.3.1**
Linear FIR filtering by the overlap-save method.

**Overlap-add method.** In this method the size of the input data block is $L$ points and the size of the DFTs and IDFT is $N = L+M-1$. To each data block we append $M-1$ zeros and compute the $N$-point DFT. Thus the data blocks may be represented as

$$x_1(n) = \{x(0), x(1), \ldots, x(L-1), \underbrace{0, 0, \ldots, 0}_{M-1 \text{ zeros}}\} \tag{7.3.13}$$

$$x_2(n) = \{x(L), x(L+1), \ldots, x(2L-1), \underbrace{0, 0, \ldots, 0}_{M-1 \text{ zeros}}\} \tag{7.3.14}$$

$$x_3(n) = \{x(2L), \ldots, x(3L-1), \underbrace{0, 0, \ldots, 0}_{M-1 \text{ zeros}}\} \tag{7.3.15}$$

and so on. The two $N$-point DFTs are multiplied together to form

$$Y_m(k) = H(k)X_m(k), \qquad k = 0, 1, \ldots, N-1 \tag{7.3.16}$$

The IDFT yields data blocks of length $N$ that are free of aliasing, since the size of the DFTs and IDFT is $N = L + M - 1$ and the sequences are increased to $N$-points by appending zeros to each block.

Since each data block is terminated with $M - 1$ zeros, the last $M - 1$ points from each output block must be overlapped and added to the first $M - 1$ points of

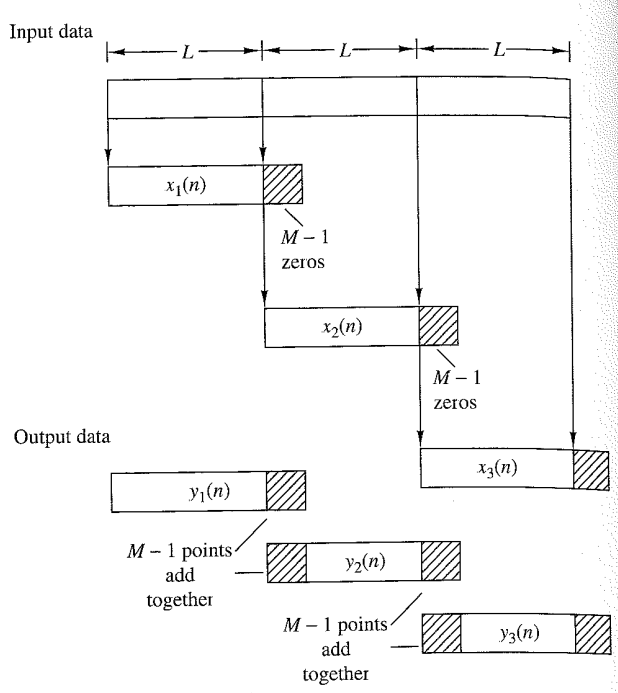



**Figure 7.3.2**
Linear FIR filtering by the overlap-add method.

the succeeding block. Hence this method is called the overlap-add method. This overlapping and adding yields the output sequence,

$$y(n) = \{y_1(0), y_1(1), \ldots, y_1(L-1), y_1(L) + y_2(0), y_1(L+1) + y_2(1), \ldots, y_1(N-1) + y_2(M-1), y_2(M), \ldots\} \tag{7.3.17}$$

The segmentation of the input data into blocks and the fitting of the output data blocks to form the output sequence are graphically illustrated in Fig. 7.3.2.

At this point, it may appear to the reader that the use of the DFT in linear FIR filtering not only is an indirect method of computing the output of an FIR filter, but also may be more expensive computationally, since the input data must first be converted to the frequency domain via the DFT, multiplied by the DFT of the FIR filter, and finally, converted back to the time domain via the IDFT. On the contrary, however, by using the fast Fourier transform algorithm, as will be shown in Chapter 8, the DFTs and IDFT require fewer computations to compute the output sequence than the direct realization of the FIR filter in the time domain. This computational efficiency is the basic advantage of using the DFT to compute the output of an FIR filter.

## 7.4 Frequency Analysis of Signals Using the DFT

To compute the spectrum of either a continuous-time or discrete-time signal, the values of the signal for all time are required. However, in practice, we observe signals for only a finite duration. Consequently, the spectrum of a signal can only be

approximated from a finite data record. In this section we examine the implications of a finite data record in frequency analysis using the DFT.

If the signal to be analyzed is an analog signal, we would first pass it through an antialiasing filter and then sample it at a rate $F_s \geq 2B$, where $B$ is the bandwidth of the filtered signal. Thus the highest frequency that is contained in the sampled signal is $F_s/2$. Finally, for practical purposes, we limit the duration of the signal to the time interval $T_0 = LT$, where $L$ is the number of samples and $T$ is the sample interval. As we shall observe in the following discussion, the finite observation interval for the signal places a limit on the frequency resolution; that is, it limits our ability to distinguish two frequency components that are separated by less than $1/T_0 = 1/LT$ in frequency.

Let $\{x(n)\}$ denote the sequence to be analyzed. Limiting the duration of the sequence to $L$ samples, in the interval $0 \leq n \leq L-1$, is equivalent to multiplying $\{x(n)\}$ by a rectangular window $w(n)$ of length $L$. That is,

$$\hat{x}(n) = x(n)w(n) \tag{7.4.1}$$

where

$$w(n) = \begin{cases} 1, & 0 \leq n \leq L-1 \\ 0, & \text{otherwise} \end{cases} \tag{7.4.2}$$

Now suppose that the sequence $x(n)$ consists of a single sinusoid, that is,

$$x(n) = \cos \omega_0 n \tag{7.4.3}$$

Then the Fourier transform of the finite-duration sequence $x(n)$ can be expressed as

$$\hat{X}(\omega) = \frac{1}{2}[W(\omega - \omega_0) + W(\omega + \omega_0)] \tag{7.4.4}$$

where $W(\omega)$ is the Fourier transform of the window sequence, which is (for the rectangular window)

$$W(\omega) = \frac{\sin(\omega L/2)}{\sin(\omega/2)} e^{-j\omega(L-1)/2} \tag{7.4.5}$$

To compute $\hat{X}(\omega)$ we use the DFT. By padding the sequence $\hat{x}(n)$ with $N-L$ zeros, we can compute the $N$-point DFT of the truncated ($L$ points) sequence $\{\hat{x}(n)\}$. The magnitude spectrum $|\hat{X}(k)| = |\hat{X}(\omega_k)|$ for $\omega_k = 2\pi k/N$, $k = 0, 1, \ldots, N$, is illustrated in Fig. 7.4.1 for $L = 25$ and $N = 2048$. We note that the windowed spectrum $\hat{X}(\omega)$ is not localized to a single frequency, but instead it is spread out over the whole frequency range. Thus the power of the original signal sequence $\{x(n)\}$ that was concentrated at a single frequency has been spread by the window into the entire frequency range. We say that the power has "leaked out" into the entire frequency range. Consequently, this phenomenon, which is a characteristic of windowing the signal, is called *leakage*.

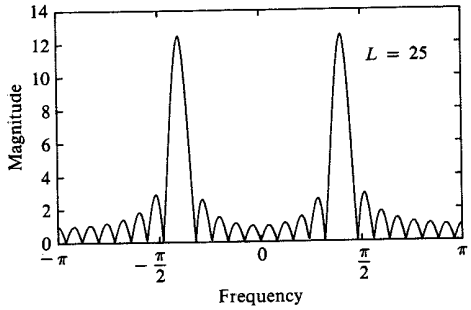



**Figure 7.4.1** Magnitude spectrum for $L = 25$ and $N = 2048$, illustrating the occurrence of leakage.

Windowing not only distorts the spectral estimate due to the leakage effects, it also reduces spectral resolution. To illustrate this problem, let us consider a signal sequence consisting of two frequency components,

$$x(n) = \cos \omega_1 n + \cos \omega_2 n \tag{7.4.6}$$

When this sequence is truncated to $L$ samples in the range $0 \le n \le L - 1$, the windowed spectrum is

$$\hat{X}(\omega) = \frac{1}{2}[W(\omega - \omega_1) + W(\omega - \omega_2) + W(\omega + \omega_1) + W(\omega + \omega_2)] \tag{7.4.7}$$

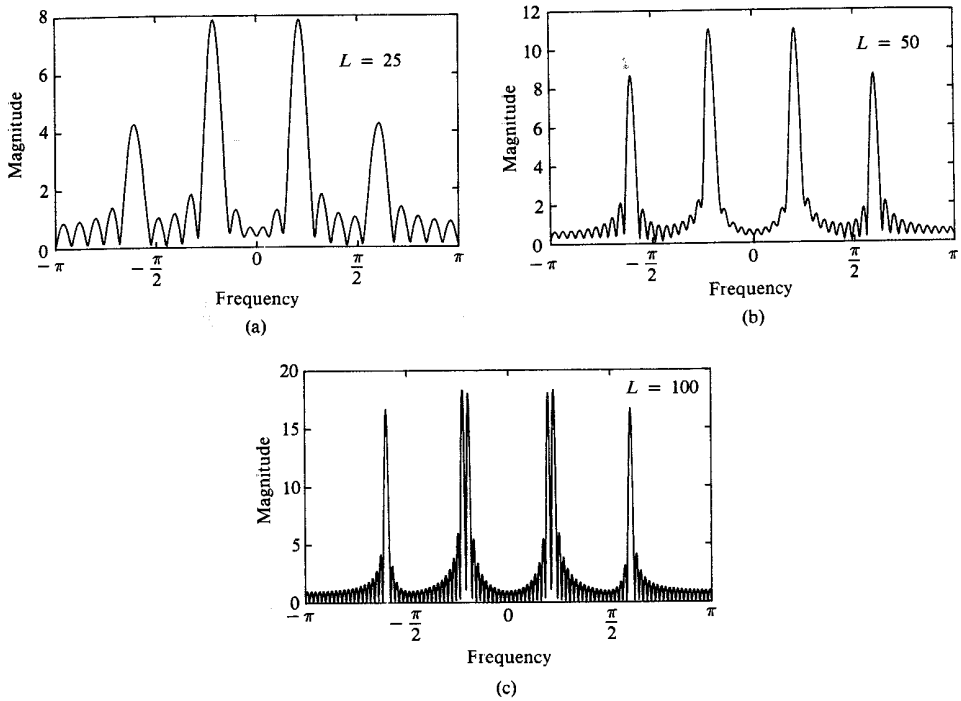



**Figure 7.4.2** Magnitude spectrum for the signal given by (7.4.8), as observed through a rectangular window.

The spectrum $W(\omega)$ of the rectangular window sequence has its first zero crossing at $\omega = 2\pi/L$. Now if $|\omega_1 - \omega_2| < 2\pi/L$, the two window functions $W(\omega - \omega_1)$ and $W(\omega - \omega_2)$ overlap and, as a consequence, the two spectral lines in $x(n)$ are not distinguishable. Only if $(\omega_1 - \omega_2) \geq 2\pi/L$ will we see two separate lobes in the spectrum $\hat{X}(\omega)$. Thus our ability to resolve spectral lines of different frequencies is limited by the window main lobe width. Figure 7.4.2 illustrates the magnitude spectrum $|\hat{X}(\omega)|$, computed via the DFT, for the sequence

$$x(n) = \cos \omega_0 n + \cos \omega_1 n + \cos \omega_2 n \tag{7.4.8}$$

where $\omega_0 = 0.2\pi$, $\omega_1 = 0.22\pi$, and $\omega_2 = 0.6\pi$. The window lengths selected are $L = 25, 50$, and $100$. Note that $\omega_0$ and $\omega_1$ are not resolvable for $L = 25$ and 50, but they are resolvable for $L = 100$.

To reduce leakage, we can select a data window $w(n)$ that has lower sidelobes in the frequency domain compared with the rectangular window. However, as we describe in more detail in Chapter 10, a reduction of the sidelobes in a window $W(\omega)$ is obtained at the expense of an increase in the width of the main lobe of $W(\omega)$ and hence a loss in resolution. To illustrate this point, let us consider the Hanning window, which is specified as

$$w(n) = \begin{cases} \frac{1}{2}(1 - \cos \frac{2\pi}{L-1} n), & 0 \leq n \leq L-1 \\ 0, & \text{otherwise} \end{cases} \tag{7.4.9}$$

Figure 7.4.3 shows $|\hat{X}(\omega)|$ given by (7.4.4) for the window of (7.4.9). Its sidelobes are significantly smaller than those of the rectangular window, but its main lobe is approximately twice as wide. Figure 7.4.4 shows the spectrum of the signal in (7.4.8), after it is windowed by the Hanning window, for $L = 50, 75$, and $100$. The reduction of the sidelobes and the decrease in the resolution, compared with the rectangular window, is clearly evident.

For a general signal sequence $\{x(n)\}$, the frequency-domain relationship between the windowed sequence $\hat{x}(n)$ and the original sequence $x(n)$ is given by the convolution formula

$$\hat{X}(\omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\theta) W(\omega - \theta) d\theta \tag{7.4.10}$$

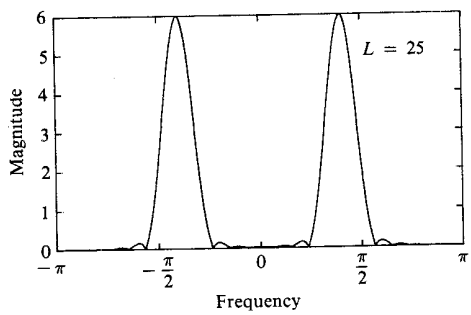



**Figure 7.4.3**
Magnitude spectrum of the Hanning window.

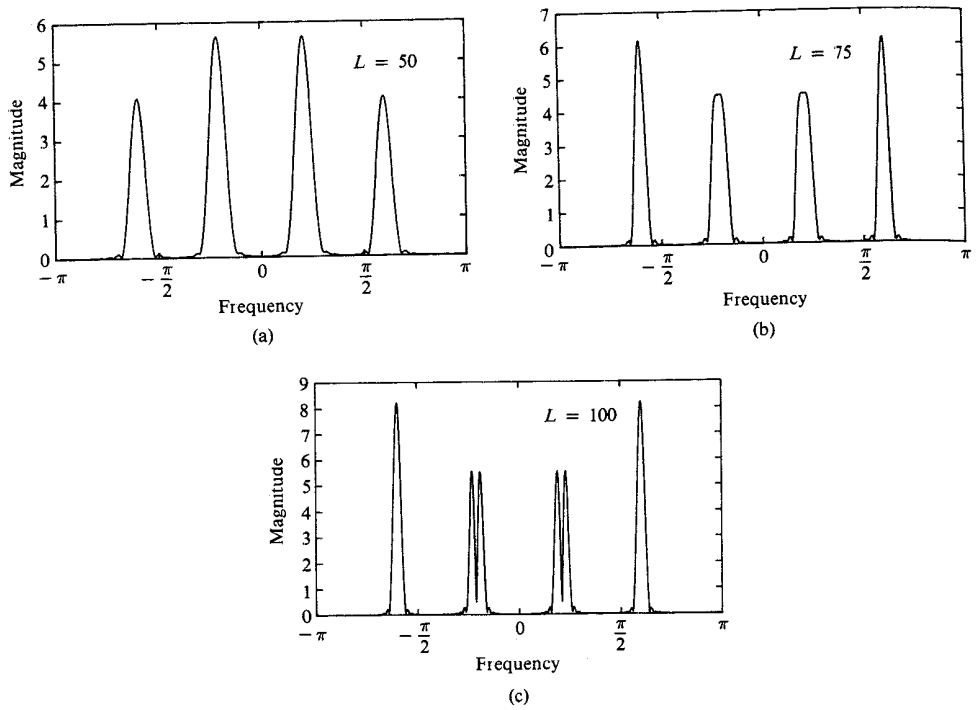



**Figure 7.4.4** Magnitude spectrum of the signal in (7.4.8) as observed through a Hanning window.

The DFT of the windowed sequence $\hat{x}(n)$ is the sampled version of the spectrum $\hat{X}(\omega)$. Thus we have

$$\begin{aligned}\hat{X}(k) &\equiv \hat{X}(\omega)|_{\omega=2\pi k/N} \\ &= \frac{1}{2\pi}\int_{-\pi}^{\pi} X(\theta)W\left(\frac{2\pi k}{N}-\theta\right)d\theta, \qquad k=0,1,\ldots,N-1\end{aligned} \tag{7.4.11}$$

Just as in the case of the sinusoidal sequence, if the spectrum of the window is relatively narrow in width compared to the spectrum $X(\omega)$ of the signal, the window function has only a small (smoothing) effect on the spectrum $X(\omega)$. On the other hand, if the window function has a wide spectrum compared to the width of $X(\omega)$, as would be the case when the number of samples $L$ is small, the window spectrum masks the signal spectrum and, consequently, the DFT of the data reflects the spectral characteristics of the window function. Of course, this situation should be avoided.

### EXAMPLE 7.4.1

The exponential signal

$$x_a(t) = \begin{cases} e^{-t}, & t \geq 0 \\ 0, & t < 0 \end{cases}$$

is sampled at the rate $F_s = 20$ samples per second, and a block of 100 samples is used to

estimate its spectrum. Determine the spectral characteristics of the signal $x_a(t)$ by computing the DFT of the finite-duration sequence. Compare the spectrum of the truncated discrete-time signal to the spectrum of the analog signal.

**Solution.** The spectrum of the analog signal is

$$X_a(F) = \frac{1}{1 + j2\pi F}$$

The exponential analog signal sampled at the rate of 20 samples per second yields the sequence

$$\begin{aligned} x(n) &= e^{-nT} = e^{-n/20}, \qquad n \geq 0 \\ &= (e^{-1/20})^n = (0.95)^n, \quad n \geq 0 \end{aligned}$$

Now, let

$$x(n) = \begin{cases} (0.95)^n, & 0 \leq n \leq 99 \\ 0, & \text{otherwise} \end{cases}$$

The $N$-point DFT of the $L = 100$ point sequence is

$$\hat{X}(k) = \sum_{k=0}^{99} \hat{x}(n) e^{-j2\pi k/N}, \qquad k = 0, 1, \ldots, N-1$$

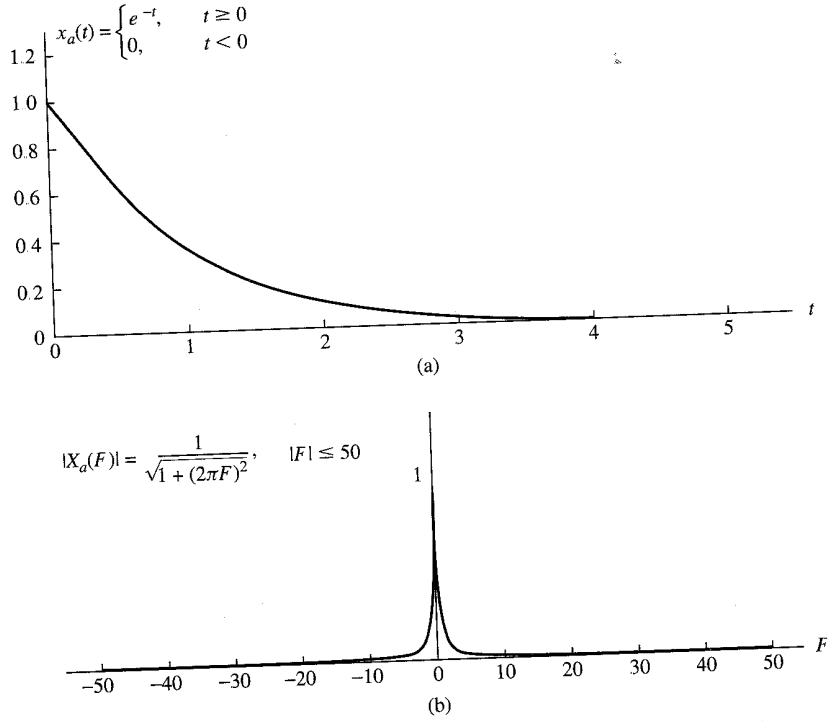



**Figure 7.4.5** Effect of windowing (truncating) the sampled version of the analog signal in Example 7.4.1

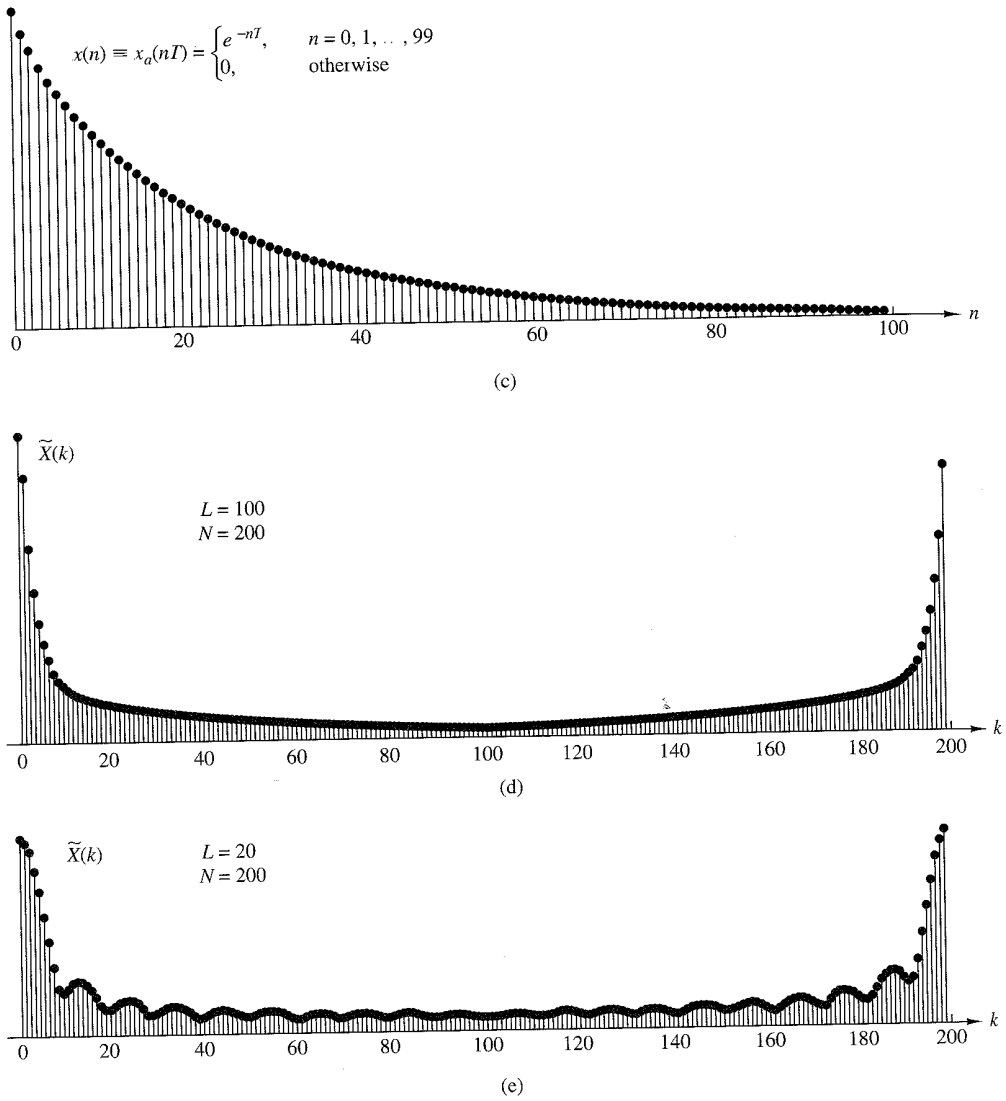



**Figure 7.4.5** *Continued*

To obtain sufficient detail in the spectrum we choose $N = 200$. This is equivalent to padding the sequence $x(n)$ with 100 zeros.

The graph of the analog signal $x_a(t)$ and its magnitude spectrum $|X_a(F)|$ are illustrated in Fig. 7.4.5(a) and 7.4.5(b), respectively. The truncated sequence $x(n)$ and its $N = 200$ point DFT (magnitude) are illustrated in Fig. 7.4.5(c) and 7.4.5(d), respectively. In this case the DFT $\{X(k)\}$ bears a close resemblance to the spectrum of the analog signal. The effect of the window function is relatively small.

On the other hand, suppose that a window function of length $L = 20$ is selected. Then the truncated sequence $x(n)$ is given as

$$\hat{x}(n) = \begin{cases} (0.95)^n, & 0 \le n \le 19 \\ 0, & \text{otherwise} \end{cases}$$

Its $N = 200$-point DFT is illustrated in Fig. 7.4.5(e). Now the effect of the wider spectral window function is clearly evident. First, the main peak is very wide as a result of the wide spectral window. Second, the sinusoidal envelope variations in the spectrum away from the main peak are due to the large sidelobes of the rectangular window spectrum. Consequently, the DFT is no longer a good approximation of the analog signal spectrum.

## 7.5 The Discrete Cosine Transform

The DFT represents an $N$-point sequence $x(n)$, $0 \le n \le N-1$, as a linear combination of complex exponentials. As a result, the DFT coefficients are, in general, complex even if $x(n)$ is real. Suppose that we wish to find an $N \times N$ orthogonal transform that expresses a real sequence $x(n)$ as a linear combination of cosine sequences. From (7.2.25) and (7.2.26), we see that this is possible if the $N$-point sequence $x(n)$ is real and even, that is, $x(n) = x(N-n)$, $0 \le n \le N-1$. The resulting DFT, $X(k)$, is itself real and even. This observation suggests that we could possibly derive a discrete cosine transform for any $N$-point real sequence by taking the $2N$-point DFT of an "even extension" of the sequence. Because there are eight ways to do this even extension, there are as many definitions of the DCT (Wang 1984, Martucci 1994). We discuss a version known as DCT-II, which is widely used in practice for speech and image compression applications as part of various standards (Rao and Huang, 1996). For simplicity, we will use the term DCT to refer to DCT-II.

### 7.5.1 Forward DCT

Let $s(n)$ be a $2N$-point even symmetric extension of $x(n)$ defined by

$$s(n) = \begin{cases} x(n), & 0 \le n \le N-1 \\ x(2N-n-1), & N \le n \le 2N-1 \end{cases} \tag{7.5.1}$$

The sequence $s(n)$ has even symmetry about the "half-sample" point $n = N + (1/2)$ (see Figure 7.5.1). The $2N$-point DFT of $s(n)$ is given by

$$S(k) = \sum_{n=0}^{2N-1} s(n) W_{2N}^{nk}, \quad 0 \le k \le 2N-1 \tag{7.5.2}$$

Substitution of (7.5.1) in (7.5.2) yields

$$S(k) = \sum_{n=0}^{N-1} x(n) W_{2N}^{nk} + \sum_{n=N}^{2N-1} x(2N-n-1) W_{2N}^{nk} \tag{7.5.3}$$

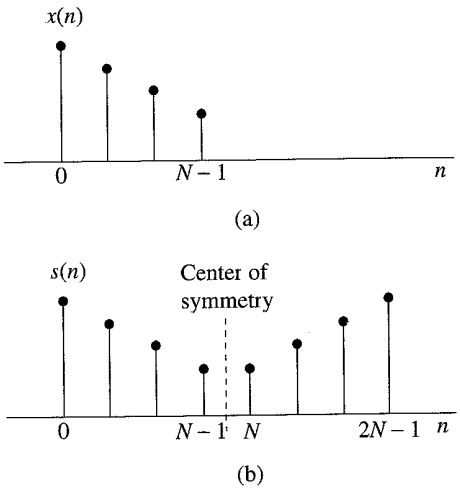



**Figure 7.5.1**
Original sequence $x(n)$, $0 \le n \le N-1$ and its $2N$-point even extension $s(n)$, $0 \le n \le 2N-1$.

If we change the second index of summation using $n = 2N - 1 - m$, we recall that $W_{2N}^{2mN} = 1$ for integer $m$, and we factor out $W_{2N}^{-k/2}$, we obtain

$$S(k) = W_{2N}^{-k/2} \sum_{n=0}^{N-1} x(n) \left[ W_{2N}^{nk} W_{2N}^{k/2} + W_{2N}^{-nk} W_{2N}^{-k/2} \right], \quad 0 \le k \le 2N-1 \tag{7.5.4}$$

The last expression may be written as

$$S(k) = W_{2N}^{-k/2} 2 \sum_{n=0}^{N-1} x(n) \cos\left[ \frac{\pi}{N} \left( n + \frac{1}{2} \right) k \right], \quad 0 \le k \le 2N-1 \tag{7.5.5}$$

or equivalently

$$S(k) = W_{2N}^{-k/2} 2\Re\left[ W_{2N}^{k/2} \sum_{n=0}^{N-1} x(n) W_{2N}^{kn} \right], \quad 0 \le k \le 2N-1 \tag{7.5.6}$$

If we define the forward DCT by

$$V(k) = 2 \sum_{n=0}^{N-1} x(n) \cos\left[ \frac{\pi}{N} \left( n + \frac{1}{2} \right) k \right], \quad 0 \le k \le N-1 \tag{7.5.7}$$

we can easily show that

$$V(k) = W_{2N}^{k/2} S(k) \text{ or } S(k) = W_{2N}^{-k/2} V(k), \quad 0 \le k \le N-1 \tag{7.5.8}$$

and

$$V(k) = 2\Re\left[ W_{2N}^{k/2} \sum_{n=0}^{N-1} x(n) W_{2N}^{kn} \right], \quad 0 \le k \le N-1 \tag{7.5.9}$$

We note that $V(k)$ is real and $S(k)$ is complex. $S(k)$ is complex because the real sequence $s(n)$ satisfies the symmetry relation $s(2N-1-n) = s(n)$ instead of $s(2N-n) = s(n)$.

The DCT of $x(n)$ can be computed by taking the $2N$-point DFT of $s(n)$, as in (7.5.2), and multiplying the result by $W_{2N}^{k/2}$, as in (7.5.8). Another approach, suggested by (7.5.9), is to take the $2N$-point DFT of the original sequence $x(n)$ with $N$ zeros appended to it, multiply the result by $W_{2N}^{k/2}$, and then take twice the real part.

### 7.5.2 Inverse DCT

We shall derive the inverse DCT from the inverse DFT of the even extended sequence $s(n)$. The inverse DFT of $S(k)$ is given by

$$s(n) = \frac{1}{2N} \sum_{k=0}^{2N-1} S(k) W_{2N}^{-nk} \tag{7.5.10}$$

Since $s(n)$ is real, $S(k)$ is Hermitian symmetric, that is,

$$S(2N-k) = S^*(k) \tag{7.5.11}$$

Furthermore, from (7.5.7), it is easy to show that

$$S(N) = 0 \tag{7.5.12}$$

With the help of (7.5.11) and (7.5.12), (7.5.10) yields

$$\begin{aligned} s(n) &= \frac{1}{2N} \sum_{k=0}^{N-1} S(k) W_{2N}^{-kn} + \frac{1}{2N} \sum_{k=N}^{2N-1} S(k) W_{2N}^{-kn} \\ &= \frac{1}{2N} \sum_{k=0}^{N-1} S(k) W_{2N}^{-kn} + \frac{1}{2N} \sum_{m=1}^{N} S(2N-m) W_{2N}^{-(2N-m)n} \\ &= \frac{1}{2N} S(0) + \frac{1}{2N} \sum_{k=1}^{N-1} S(k) W_{2N}^{-kn} + \frac{1}{2N} \sum_{k=1}^{N-1} S^*(k) W_{2N}^{kn} \end{aligned}$$

or since $S(0)$ is real

$$s(n) = \frac{1}{N} \Re \left[ \frac{S(0)}{2} + \sum_{k=1}^{N-1} S(k) W_{2N}^{-kn} \right], \quad 0 \le n \le 2N-1 \tag{7.5.13}$$

Substituting (7.5.8) in (7.5.13) and using (7.5.1) yields the desired inverse DCT

$$x(n) = \frac{1}{N} \left\{ \frac{V(0)}{2} + \sum_{k=1}^{N-1} V(k) \cos\left[ \frac{\pi}{N} \left( n + \frac{1}{2} \right) k \right] \right\}, \quad 0 \le n \le N-1 \tag{7.5.14}$$

Given $V(k)$, we first compute $S(k)$ by (7.5.8). In the next step, we take the $2N$-point inverse DFT implied by (7.5.13). The real part of this inverse DFT yields $s(n)$ and, hence $x(n)$.

An approach to compute the DCT and inverse DCT using an $N$-point DFT is discussed in Makhoul (1980). Many special algorithms for the hardware and software implementation of the DCT are discussed in Rao and Yip (1990).

### 7.5.3 DCT as an Orthogonal Transform

Equations (7.5.7) and (7.5.14) form a DCT pair. However, for reasons to be seen later, we usually redistribute symmetrically the normalization factors between the forward and inverse transform. Thus, the DCT of the sequence $x(n)$, $0 \le n \le N-1$ and its inverse are defined by

$$C(k) = \alpha(k) \sum_{n=0}^{N-1} x(n) \cos\left[\frac{\pi(2n+1)k}{2N}\right], \qquad 0 \le k \le N-1 \tag{7.5.15}$$

$$x(n) = \sum_{k=0}^{N-1} \alpha(k) C(k) \cos\left[\frac{\pi(2n+1)k}{2N}\right], \qquad 0 \le n \le N-1 \tag{7.5.16}$$

where

$$\alpha(0) = \sqrt{\frac{1}{N}}, \quad \alpha(k) = \sqrt{\frac{2}{N}} \text{ for } 1 \le k \le N-1 \tag{7.5.17}$$

Like the DFT in Section 7.1.3, the DCT formulas (7.5.15) and (7.5.16) can be expressed in matrix form using the $N \times N$ DCT matrix $\boldsymbol{C}_N$ with elements given by

$$c_{kn} = \begin{cases} \dfrac{1}{\sqrt{N}}, & k = 0, 0 \le n \le N-1 \\ \sqrt{\dfrac{2}{N}} \cos \dfrac{\pi(2n+1)k}{2N}, & 1 \le k \le N-1, 0 \le n \le N-1 \end{cases} \tag{7.5.18}$$

If we define the following signal and coefficient vectors

$$\mathbf{x}_N = [\, x(0) \quad x(1) \quad \ldots \quad x(N-1) \,]^T \tag{7.5.19}$$

$$\mathbf{c}_N = [\, C(0) \quad C(1) \quad \ldots \quad C(N-1) \,]^T \tag{7.5.20}$$

the forward DCT (7.5.15) and the inverse DCT (7.5.16) can be written in matrix form as

$$\mathbf{c}_N = \mathbf{C}_N \mathbf{x}_N \tag{7.5.21}$$

$$\mathbf{x}_N = \mathbf{C}_N^T \mathbf{c}_N \tag{7.5.22}$$

It follows from (7.5.19) and (7.5.20) that $\mathbf{C}_N$ is a real orthogonal matrix, that is, it satisfies

$$\mathbf{C}_N^{-1} = \mathbf{C}_N^T \tag{7.5.23}$$

Orthogonality simplifies the computation of the inverse transform because it replaces matrix inversion by matrix transposition.

If we denote by $\mathbf{c}_N(k)$ the columns of $\mathbf{C}_N^T$, the inverse DCT can be written as

$$\mathbf{x}_N = \sum_{k=0}^{N-1} C(k)\mathbf{c}_N(k) \tag{7.5.24}$$

which represents the signal as a linear combination of the DCT cosine basis sequences. The value of the coefficient $C(k)$ measures the similarity of the signal with the $k$th basis vector.

**EXAMPLE 7.5.1**

Consider the discrete-time sinusoidal signal

$$x(n) = \cos(2\pi k_0 n/N), \quad 0 \le n \le N-1$$

In Figure 7.5.2 we show plots of the sequence $x(n)$, the absolute values of the $N$-point DFT $X(k)$ coefficients, and the $N$-point DCT coefficients for $k_0 = 5$ and $N = 32$. We note that, in contrast to the DFT, the DCT, although it shows a distinct peak at $2k_0$, also exhibits a significant amount of ripples at other frequencies. For this reason, the DCT is not useful for frequency analysis of signals and systems.

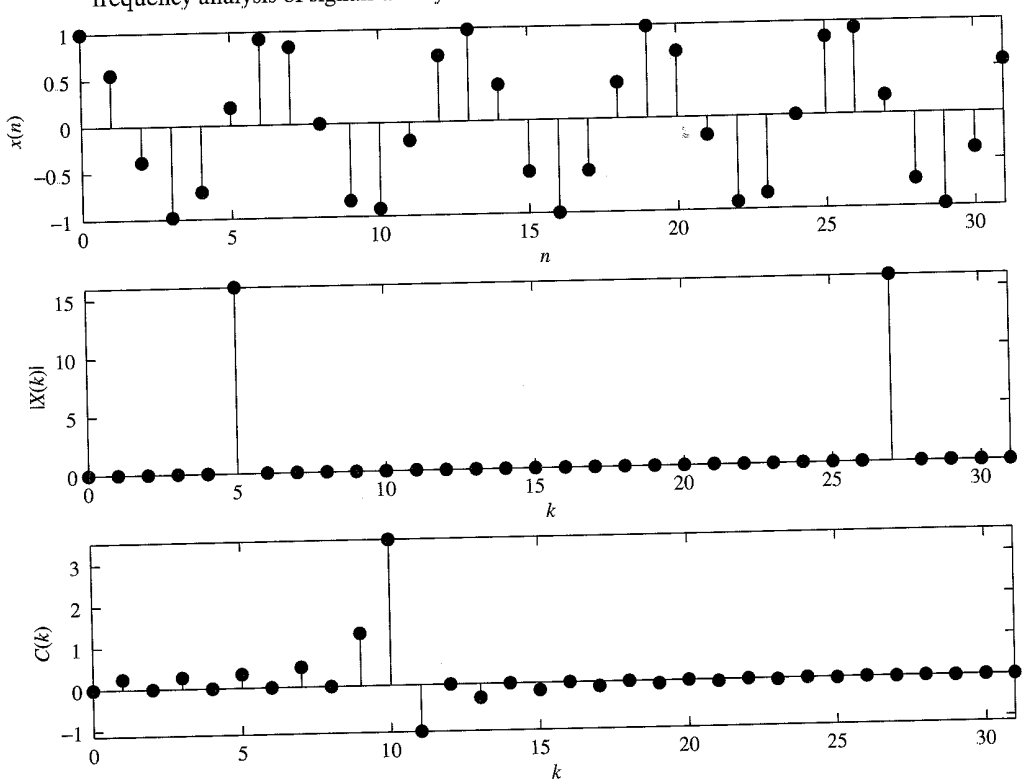



**Figure 7.5.2** A discrete-time sinusoidal signal and its DFT and DCT representations.

Using the orthogonality property (7.5.23), we can easily show that

$$\sum_{k=0}^{N-1} |C(k)|^2 = \mathbf{c}_N^T \mathbf{c}_N = \mathbf{x}_N^T \mathbf{C}_N^T \mathbf{C}_N \mathbf{x}_N = \mathbf{x}_N^T \mathbf{x}_N = \sum_{n=0}^{N-1} |x(n)|^2 = E_x \tag{7.5.25}$$

Thus, an orthogonal transformation preserves the signal energy, or equivalently, the length of the vector $\mathbf{x}$ in the $N$-dimensional vector space (generalized Parseval's theorem). This means that every orthogonal transformation is simply a rotation of the vector $\mathbf{x}$ in the $N$-dimensional vector space.

Most orthogonal transforms tend to pack a large fraction of the average energy of the signal into a relatively few components of the transform coefficients (energy compaction property). Since the total energy is preserved, many of the transform coefficients will contain very little energy. However, as we illustrate in the next example, different transforms have different energy compaction capabilities.

### EXAMPLE 7.5.2

We will compare the energy compaction capabilities of the DFT and DCT using the ramp sequence $x(n) = n$, $0 \le n \le N-1$, shown in Figure 7.5.3(a) for $N = 32$. Figures 7.5.3(d) and 7.5.3(f) show the absolute values of the DFT coefficients and the values of the DCT coefficients, respectively. Clearly, the DCT coefficients demonstrate better "energy packing" than the DFT ones. This implies that we should be able to represent the sequence $x(n)$ using a smaller number of DCT coefficients.

With the DCT we set the last $k_0$ coefficients to zero and take the inverse DCT to obtain an approximation $x_{\text{DCT}}(n)$ of the original sequence. However, since the DFT of a real sequence is complex, the information is carried in the first $N/2$ values. (For simplicity, we assume that $N$ is an even number.) Therefore, we should remove DFT coefficients in a way that preserves the complex-conjugate symmetry. This is done by first removing the coefficient $X(N/2)$, then the coefficients $X(N/2-1)$ and $X(N/2+1)$, and so forth. Clearly, we can remove only an odd number of DFT coefficients, that is, $k_0 = 1, 3, \ldots, N-1$. The reconstructed sequence using the DFT is denoted by $x_{\text{DFT}}(n)$.

The reconstruction error for the DCT, which is a function of $k_0$, is defined by

$$E_{\text{DCT}}(k_0) = \frac{1}{N} \sum_{n=0}^{N-1} |x(n) - x_{\text{DCT}}(n)|^2$$

A similar definition is used for the DFT. Figure 7.5.3(b) shows the reconstruction errors for the DFT and DCT as a function of the number $k_0$ of omitted coefficients. Figures 7.5.3(c) and 7.5.3(e) show the reconstructed signals when we retain $N - k_0 = 5$ coefficients. We see that we need fewer DCT coefficients than DFT coefficients to get a good approximation of the original signal. In this example, the DFT (due to its inherent periodicity) is trying to model a sawtooth wave. Therefore, it has to devote many high-frequency coefficients to approximate the discontinuities at the ends. In contrast, the DCT operates on the "even extension" of $x(n)$, which is a triangular wave with no discontinuities. As a result, the DCT can approximate better small blocks of signals that have fairly different values in the first and last samples.

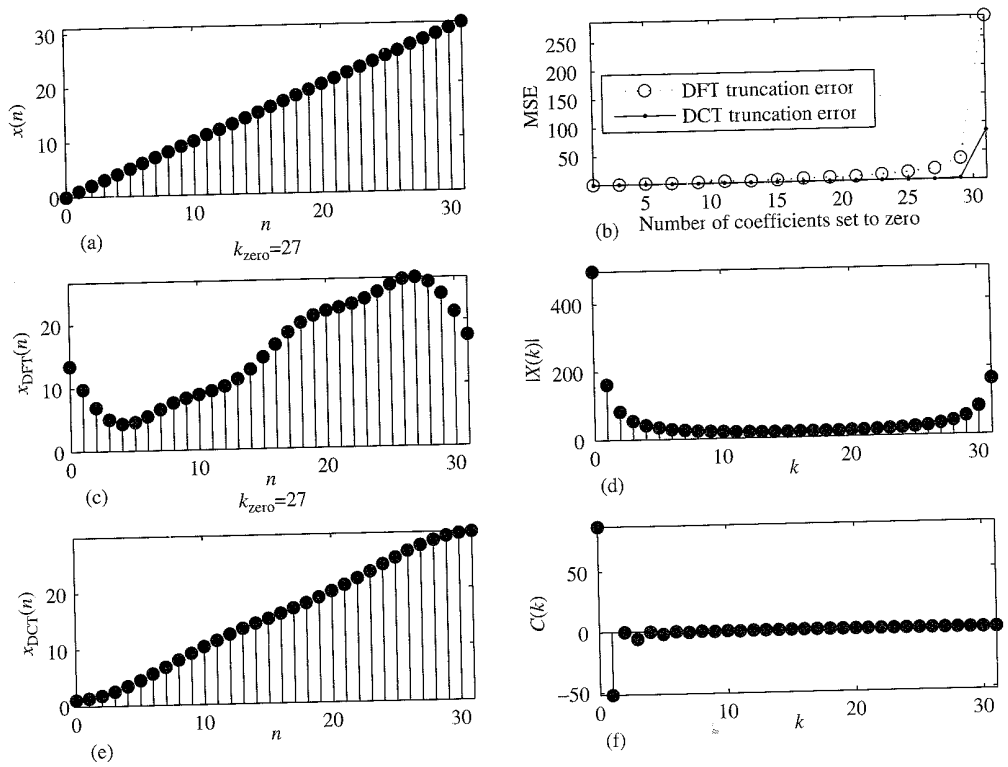



**Figure 7.5.3** A discrete time sinusoidal signal and its DFT and DCT representations.

From a statistical viewpoint, the optimum orthogonal transform for signal compression is the Karhunen–Loeve (KL) or Hotelling transform (Jayant and Noll, 1984). The KL transform has two optimality properties: (a) it minimizes the reconstruction error for any number of retained coefficients, and (b) it generates a set of uncorrelated transform coefficients. The KL transform is defined by the eigenvectors of the covariance matrix of the input sequence. The DCT provides a good approximation to the KL transform for signals that follow the difference equation $x(n) = ax(n-1)+w(n)$, where $w(n)$ is a white noise sequence, and $a$ $(0 < a < 1)$ is a constant coefficient with values close to one. Many signals, including natural images, have this characteristic. More details about orthogonal transforms and their applications can be found in Jayant and Noll (1984), Clarke (1985), Rao and Yip (1990), and Goyal (2001).

## 7.6 Summary and References

The major focus of this chapter was on the discrete Fourier transform, its properties and its applications. We developed the DFT by sampling the spectrum $X(\omega)$ of the sequence $x(n)$.

Frequency-domain sampling of the spectrum of a discrete-time signal is particularly important in the processing of digital signals. Of particular significance is the DFT, which was shown to uniquely represent a finite-duration sequence in the frequency domain. The existence of computationally efficient algorithms for the

DFT, which are described in Chapter 8, make it possible to digitally process signals in the frequency domain much faster than in the time domain. The processing methods in which the DFT is especially suitable include linear filtering as described in this chapter and correlation and spectrum analysis, which are treated in Chapters 8 and 4. A particularly lucid and concise treatment of the DFT and its application to frequency analysis is given in the book by Brigham (1988).

We also described the discrete cosine transform (DCT) in this chapter. An interesting treatment of the DCT from a linear algebra perspective is given in a paper by Strang (1999).

## Problems

**7.1** The first five points of the eight-point DFT of a real-valued sequence are $\{0.25, 0.125 - j0.3018, 0, 0.125 - j0.0518, 0\}$. Determine the remaining three points.

**7.2** Compute the eight-point circular convolution for the following sequences.

**(a)** $x_1(n) = \{1, 1, 1, 1, 0, 0, 0, 0\}$

$$x_2(n) = \sin \frac{3\pi}{8} n, \qquad 0 \le n \le 7$$

**(b)** $x_1(n) = (\frac{1}{4})^n, \qquad 0 \le n \le 7$

$$x_2(n) = \cos \frac{3\pi}{8} n, \qquad 0 \le n \le 7$$

**(c)** Compute the DFT of the two circular convolution sequences using the DFTs of $x_1(n)$ and $x_2(n)$.

**7.3** Let $X(k)$, $0 \le k \le N-1$, be the $N$-point DFT of the sequence $x(n)$, $0 \le n \le N-1$. We define

$$\hat{X}(k) = \begin{cases} X(k), & 0 \le k \le k_c, N - k_c \le k \le N-1 \\ 0, & k_c < k < N - k_c \end{cases}$$

and we compute the inverse $N$-point DFT of $\hat{X}(k)$, $0 \le k \le N-1$. What is the effect of this process on the sequence $x(n)$? Explain.

**7.4** For the sequences

$$x_1(n) = \cos \frac{2\pi}{N} n, \qquad x_2(n) = \sin \frac{2\pi}{N} n, \qquad 0 \le n \le N-1$$

determine the $N$-point:

**(a)** Circular convolution $x_1(n) \circledN x_2(n)$

**(b)** Circular correlation of $x_1(n)$ and $x_2(n)$

**(c)** Circular autocorrelation of $x_1(n)$

**(d)** Circular autocorrelation of $x_2(n)$

**7.5** Compute the quantity

$$\sum_{n=0}^{N-1} x_1(n) x_2(n)$$

for the following pairs of sequences.

**(a)** $x_1(n) = x_2(n) = \cos \frac{2\pi}{N} n, \qquad 0 \le n \le N-1$

**(b)** $x_1(n) = \cos \frac{2\pi}{N} n, \qquad x_2(n) = \sin \frac{2\pi}{N} n, \qquad 0 \le n \le N-1$

**(c)** $x_1(n) = \delta(n) + \delta(n-8), \qquad x_2(n) = u(n) - u(n-N)$

**7.6** Determine the $N$-point DFT of the Blackman window

$$w(n) = 0.42 - 0.5 \cos \frac{2\pi n}{N-1} + 0.08 \cos \frac{4\pi n}{N-1}, \qquad 0 \le n \le N-1$$

**7.7** If $X(k)$ is the DFT of the sequence $x(n)$, determine the $N$-point DFTs of the sequences

$$x_c(n) = x(n) \cos \frac{2\pi k_0 n}{N}, \qquad 0 \le n \le N-1$$

and

$$x_s(n) = x(n) \sin \frac{2\pi k_0 n}{N}, \qquad 0 \le n \le N-1$$

in terms of $X(k)$.

**7.8** Determine the circular convolution of the sequences

$$x_1(n) = \{\underset{\uparrow}{1}, 2, 3, 1\}$$

$$x_2(n) = \{\underset{\uparrow}{4}, 3, 2, 2\}$$

using the time-domain formula in (7.2.39).

**7.9** Use the four-point DFT and IDFT to determine the sequence

$$x_3(n) = x_1(n) \circledN x_2(n)$$

where $x_1(n)$ and $x_2(n)$ are the sequence given in Problem 7.8.

**7.10** Compute the energy of the $N$-point sequence

$$x(n) = \cos \frac{2\pi k_0 n}{N}, \qquad 0 \le n \le N-1$$

**7.11** Given the eight-point DFT of the sequence

$$x(n) = \begin{cases} 1, & 0 \le n \le 3 \\ 0, & 4 \le n \le 7 \end{cases}$$

compute the DFT of the sequences

**(a)** $x_1(n) = \begin{cases} 1, & n = 0 \\ 0, & 1 \le n \le 4 \\ 1, & 5 \le n \le 7 \end{cases}$

**(b)** $x_2(n) = \begin{cases} 0, & 0 \le n \le 1 \\ 1, & 2 \le n \le 5 \\ 0, & 6 \le n \le 7 \end{cases}$

**7.12** Consider a finite-duration sequence

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

**(a)** Sketch the sequence $s(n)$ with six-point DFT

$$S(k) = W_2^* X(k), \qquad k = 0, 1, \ldots, 6$$

**(b)** Determine the sequence $y(n)$ with six-point DFT $Y(k) = \Re|X(k)|$.

**(c)** Determine the sequence $v(n)$ with six-point DFT $V(k) = \Im|X(k)|$.

**7.13** Let $x_p(n)$ be a periodic sequence with fundamental period $N$. Consider the following DFTs:

$$x_p(n) \underset{N}{\overset{\text{DFT}}{\longleftrightarrow}} X_1(k)$$

$$x_p(n) \underset{3N}{\overset{\text{DFT}}{\longleftrightarrow}} X_3(k)$$

**(a)** What is the relationship between $X_1(k)$ and $X_3(k)$?

**(b)** Verify the result in part (a) using the sequence

$$x_p(n) = \{\cdots 1, 2, 1, \underset{\uparrow}{2}, 1, 2, 1, 2 \cdots\}$$

**7.14** Consider the sequences

$$x_1(n) = \{\underset{\uparrow}{0}, 1, 2, 3, 4\}, \qquad x_2(n) = \{\underset{\uparrow}{0}, 1, 0, 0, 0\}, \qquad s(n) = \{\underset{\uparrow}{1}, 0, 0, 0, 0\}$$

and their five-point DFTs.

**(a)** Determine a sequence $y(n)$ so that $Y(k) = X_1(k)X_2(k)$.

**(b)** Is there a sequence $x_3(n)$ such that $S(k) = X_1(k)X_3(k)$?

**7.15** Consider a causal LTI system with system function

$$H(z) = \frac{1}{1 - 0.5z^{-1}}$$

The output $y(n)$ of the system is known for $0 \le n \le 63$. Assuming that $H(z)$ is available, can you develop a 64-point DFT method to recover the sequence $x(n)$, $0 \le n \le 63$? Can you recover all values of $x(n)$ in this interval?

**7.16** The impulse response of an LTI system is given by $h(n) = \delta(n) - \frac{1}{4}\delta(n - k_0)$. To determine the impulse response $g(n)$ of the inverse system, an engineer computes the $N$-point DFT $H(k)$, $N = 4k_0$, of $h(n)$ and then defines $g(n)$ as the inverse DFT of $G(k) = 1/H(k)$, $k = 0, 1, 2, \ldots, N - 1$. Determine $g(n)$ and the convolution $h(n) \times g(n)$, and comment on whether the system with impulse response $g(n)$ is the inverse of the system with impulse response $h(n)$.

**7.17** Determine the eight-point DFT of the signal

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

and sketch its magnitude and phase.

**7.18** A linear time-invariant system with frequency response $H(\omega)$ is excited with the periodic input

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

Suppose that we compute the $N$-point DFT $Y(k)$ of the samples $y(n)$, $0 \le n \le N-1$ of the output sequence. How is $Y(k)$ related to $H(\omega)$?

**7.19** *DFT of real sequences with special symmetries*

**(a)** Using the symmetry properties of Section 7.2 (especially the decomposition properties), explain how we can compute the DFT of two real symmetric (even) and two real antisymmetric (odd) sequences simultaneously using an $N$-point DFT only.

**(b)** Suppose now that we are given four real sequences $x_i(n)$, $i = 1, 2, 3, 4$, that are all symmetric [i.e., $x_i(n) = x_i(N-n)$, $0 \le n \le N-1$]. Show that the sequences

$$s_i(n) = x_i(n+1) - x_i(n-1)$$

are antisymmetric [i.e., $s_i(n) = -s_i(N-n)$ and $s_i(0) = 0$].

**(c)** Form a sequence $x(n)$ using $x_1(n)$, $x_2(n)$, $s_3(n)$, and $s_4(n)$ and show how to compute the DFT $X_i(k)$ of $x_i(n)$, $i = 1, 2, 3, 4$ from the $N$-point DFT $X(k)$ of $x(n)$.

**(d)** Are there any frequency samples of $X_i(k)$ that cannot be recovered from $X(k)$? Explain.

**7.20** *DFT of real sequences with odd harmonics only* Let $x(n)$ be an $N$-point real sequence with $N$-point DFT $X(k)$ ($N$ even). In addition, $x(n)$ satisfies the following symmetry property:

$$x\left(n + \frac{N}{2}\right) = -x(n), \qquad n = 0, 1, \ldots, \frac{N}{2} - 1$$

that is, the upper half of the sequence is the negative of the lower half.

**(a)** Show that

$$X(k) = 0, \qquad k \text{ even}$$

that is, the sequence has a spectrum with odd harmonics.

**(b)** Show that the values of this odd-harmonic spectrum can be computed by evaluating the $N/2$-point DFT of a complex modulated version of the original sequence $x(n)$.

**7.21** Let $x_a(t)$ be an analog signal with bandwidth $B = 3$ kHz. We wish to use an $N = 2^m$-point DFT to compute the spectrum of the signal with a resolution less than or equal to 50 Hz. Determine **(a)** the minimum sampling rate, **(b)** the minimum number of required samples, and **(c)** the minimum length of the analog signal record.

**7.22** Consider the periodic sequence

$$x_p(n) = \cos \frac{2\pi}{10} n, \qquad -\infty < n < \infty$$

with frequency $f_0 = \frac{1}{10}$ and fundamental period $N = 10$. Determine the 10-point DFT of the sequence $x(n) = x_p(n)$, $0 \le n \le N-1$.

**7.23** Compute the $N$-point DFTs of the signals

**(a)** $x(n) = \delta(n)$

**(b)** $x(n) = \delta(n - n_0), \quad 0 < n_0 < N$

**(c)** $x(n) = a^n, \quad 0 \le n \le N-1$

**(d)** $x(n) = \begin{cases} 1, & 0 \le n \le N/2 - 1 \quad (N \text{ even}) \\ 0, & N/2 \le n \le N-1 \end{cases}$

**(e)** $x(n) = e^{j(2\pi/N)k_0 n}, \quad 0 \le n \le N-1$

**(f)** $x(n) = \cos \frac{2\pi}{N} k_0 n, \quad 0 \le n \le N-1$

**(g)** $x(n) = \sin \frac{2n}{N} k_0 n, \quad 0 \le n \le N-1$

**(h)** $x(n) = \begin{cases} 1, & n \text{ even} \\ 0, & n \text{ odd}, \quad 0 \le n \le N-1 \end{cases}$

**7.24** Consider the finite-duration signal

$$x(n) = \{1, 2, 3, 1\}$$

**(a)** Compute its four-point DFT by solving explicitly the 4-by-4 system of linear equations defined by the inverse DFT formula.

**(b)** Check the answer in part (a) by computing the four-point DFT, using its definition.

**7.25** **(a)** Determine the Fourier transform $X(\omega)$ of the signal

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

**(b)** Compute the six-point DFT $V(k)$ of the signal

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

**(c)** Is there any relation between $X(\omega)$ and $V(k)$? Explain.

**7.26** Prove the identity

$$\sum_{l=-\infty}^{\infty} \delta(n + lN) = \frac{1}{N} \sum_{k=0}^{N-1} e^{j(2\pi/N)kn}$$

(*Hint:* Find the DFT of the periodic signal in the left-hand side.)

**7.27** *Computation of the even and odd harmonics using the DFT.* Let $x(n)$ be an $N$-point sequence with an $N$-point DFT $X(k)$ ($N$ even).

**(a)** Consider the time-aliased sequence

$$y(n) = \begin{cases} \sum_{l=-\infty}^{\infty} x(n+lM), & 0 \le n \le M-1 \\ 0, & \text{elsewhere} \end{cases}$$

What is the relationship between the $M$-point DFT $Y(k)$ of $y(n)$ and the Fourier transform $X(\omega)$ of $x(n)$?

**(b)** Let

$$y(n) = \begin{cases} x(n) + x\left(n + \frac{N}{2}\right), & 0 \le n \le N-1 \\ 0, & \text{elsewhere} \end{cases}$$

and

$$y(n) \underset{N/2}{\overset{\text{DFT}}{\longleftrightarrow}} Y(k)$$

Show that $X(k) = Y(k/2)$, $k = 2, 4, \ldots, N-2$.

**(c)** Use the results in parts (a) and (b) to develop a procedure that computes the odd harmonics of $X(k)$ using an $N/2$-point DFT.

**7.28** *Frequency-domain sampling.* Consider the following discrete-time signal

$$x(n) = \begin{cases} a^{|n|}, & |n| \le L \\ 0, & |n| > L \end{cases}$$

where $a = 0.95$ and $L = 10$.

**(a)** Compute and plot the signal $x(n)$.

**(b)** Show that

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} = x(0) + 2\sum_{n-1}^{L} x(n)\cos\omega n$$

Plot $X(\omega)$ by computing it at $\omega = \pi k/100$, $k = 0, 1, \ldots, 100$.

**(c)** Compute

$$c_k = \frac{1}{N} X\left(\frac{2\pi}{N}K\right), \qquad k = 0, 1, \ldots, N-1$$

for $N = 30$.

**(d)** Determine and plot the signal

$$\tilde{x}(n) = \sum_{k=0}^{N-1} cke^{j(2\pi/N)kn}$$

What is the relation between the signals $x(n)$ and $\tilde{x}(n)$? Explain.

**(e)** Compute and plot the signal $\tilde{x}_1(n) = \sum_{l=-\infty}^{\infty} x(n-lN)$, $-L \le n \le L$ for $N = 30$. Compare the signals $\tilde{x}(n)$ and $\tilde{x}_1(n)$.

**(f)** Repeat parts (c) to (e) for $N = 15$.

**7.29** *Frequency-domain sampling* The signal $x(n) = a^{|n|}$, $-1 < a < 1$ has a Fourier transform

$$X(\omega) = \frac{1 - a^2}{1 - 2a\cos\omega + a^2}$$

**(a)** Plot $X(\omega)$ for $0 \le \omega \le 2\pi$, $a = 0.8$. Reconstruct and plot $X(\omega)$ from its samples $X(2\pi k/N)$, $0 \le k \le N - 1$ for

**(b)** $N = 20$

**(c)** $N = 100$

**(d)** Compare the spectra obtained in parts (b) and (c) with the original spectrum $X(\omega)$ and explain the differences.

**(e)** Illustrate the time-domain aliasing when $N = 20$.

**7.30** *Frequency analysis of amplitude-modulated discrete-time signal* The discrete-time signal

$$x(n) = \cos 2\pi f_1 n + \cos 2\pi f_2 n$$

where $f_1 = \frac{1}{18}$ and $f_2 = \frac{5}{128}$, modulates the amplitude of the carrier

$$x_c(n) = \cos 2\pi f_c n$$

where $f_c = \frac{50}{128}$. The resulting amplitude-modulated signal is

$$x_{\text{am}}(n) = x(n)\cos 2\pi f_c n$$

**(a)** Sketch the signals $x(n)$, $x_c(n)$, and $x_{\text{am}}(n)$, $0 \le n \le 255$.

**(b)** Compute and sketch the 128-point DFT of the signal $x_{\text{am}}(n)$, $0 \le n \le 127$.

**(c)** Compute and sketch the 128-point DFT of the signal $x_{\text{am}}(n)$, $0 \le n \le 99$.

**(d)** Compute and sketch the 256-point DFT of the signal $x_{\text{am}}(n)$, $0 \le n \le 179$.

**(e)** Explain the results obtained in parts (b) through (d), by deriving the spectrum of the amplitude-modulated signal and comparing it with the experimental results.

**7.31** The sawtooth waveform in Fig. P7.31 can be expressed in the form of a Fourier series as

$$x(t) = \frac{2}{\pi}\left(\sin \pi t - \frac{1}{2}\sin 2\pi t + \frac{1}{3}\sin 3\pi t - \frac{1}{4}\sin 4\pi t \cdots\right)$$

**(a)** Determine the Fourier series coefficients $c_k$.

**(b)** Use an $N$-point subroutine to generate samples of this signal in the time domain using the first six terms of the expansion for $N = 64$ and $N = 128$. Plot the signal $x(t)$ and the samples generated, and comment on the results.

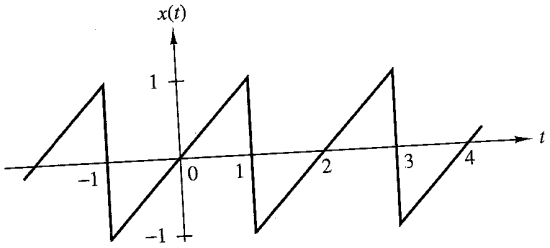



Figure P7.31

**7.32** Recall that the Fourier transform of $x(t) = e^{j\Omega_0 t}$ is $X(j\Omega) = 2\pi\delta(\Omega - \Omega_0)$ and the Fourier transform of

$$p(t) = \begin{cases} 1, & 0 \le t \le T_0 \\ 0, & \text{otherwise} \end{cases}$$

is

$$P(j\Omega) = T_0 \frac{\sin \Omega T_0/2}{\Omega T_0/2} e^{-j\Omega T_0/2}$$

**(a)** Determine the Fourier transform $Y(j\Omega)$ of

$$y(t) = p(t)e^{j\Omega_0 t}$$

and roughly sketch $|Y(j\Omega)|$ versus $\Omega$.

**(b)** Now consider the exponential sequence

$$x(n) = e^{j\omega_0 n}$$

where $\omega_0$ is some arbitrary frequency in the range $0 < \omega_0 < \pi$ radians. Give the most general condition that $\omega_0$ must satisfy in order for $x(n)$ to be periodic with period $P$ ($P$ is a positive integer).

**(c)** Let $y(n)$ be the finite-duration sequence

$$y(n) = x(n)w_N(n) = e^{j\omega_0 n} w_N(n)$$

where $w_N(n)$ is a finite-duration rectangular sequence of length $N$ and where $x(n)$ is not necessarily periodic. Determine $Y(\omega)$ and roughly sketch $|Y(\omega)|$ for $0 \le \omega \le 2\pi$. What effect does $N$ have in $|Y(\omega)|$? Briefly comment on the similarities and differences between $|Y(\omega)|$ and $|Y(j\Omega)|$.

**(d)** Suppose that

$$x(n) = e^{j(2\pi/P)n}, \qquad P \text{ a positive integer}$$

and

$$y(n) = w_N(n)x(n)$$

where $N = lP$, $l$ a positive integer. Determine and sketch the $N$-point DFT of $y(n)$. Relate your answer to the characteristics of $|Y(\omega)|$.

**(e)** Is the frequency sampling for the DFT in part (d) adequate for obtaining a rough approximation of $|Y(\omega)|$ directly from the magnitude of the DFT sequence $|Y(k)|$? If not, explain briefly how the sampling can be increased so that it will be possible to obtain a rough sketch of $|Y(\omega)|$ from an appropriate sequence $|Y(k)|$.

**7.33** Develop an algorithm that computes the DCT using the DFT as described in Sections 7.5.1 and 7.5.2.

**7.34** Use the algorithm developed in Problem 7.33 to reproduce the results in Example 7.5.2.

**7.35** Repeat Example 7.5.2 using the signal $x(n) = a^n \cos(2\pi f_0 n + \phi)$ with $a = 0.8$, $f_0 = 0.05$, and $N = 32$.

ELCoM
Electrical | Computer | Mechatronics
ELCOM-HU.COM

CHAPTER

8

# Efficient Computation of the DFT: Fast Fourier Transform Algorithms

As we have observed in the preceding chapter, the discrete fourier transform (DFT) plays an important role in many applications of digital signal processing, including linear filtering, correlation analysis, and spectrum analysis. A major reason for its importance is the existence of efficient algorithms for computing the DFT.

The main topic of this chapter is the description of computationally efficient algorithms for evaluating the DFT. Two different approaches are described. One is a divide-and-conquer approach in which a DFT of size $N$, where $N$ is a composite number, is reduced to the computation of smaller DFTs from which the larger DFT is computed. In particular, we present important computational algorithms, called fast Fourier transform (FFT) algorithms, for computing the DFT when the size $N$ is a power of 2 and when it is a power of 4.

The second approach is based on the formulation of the DFT as a linear filtering operation on the data. This approach leads to two algorithms, the Goertzel algorithm and the chirp-$z$ transform algorithm, for computing the DFT via linear filtering of the data sequence.

## 8.1 Efficient Computation of the DFT: FFT Algorithms

In this section we present several methods for computing the DFT efficiently. In view of the importance of the DFT in various digital signal processing applications, such as linear filtering, correlation analysis, and spectrum analysis, its efficient computation is a topic that has received considerable attention by many mathematicians, engineers, and applied scientists.

Basically, the computational problem for the DFT is to compute the sequence $\{X(k)\}$ of $N$ complex-valued numbers given another sequence of data $\{x(n)\}$ of length

$N$, according to the formula

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}, \qquad 0 \le k \le N-1 \tag{8.1.1}$$

where

$$W_N = e^{-j2\pi/N} \tag{8.1.2}$$

In general, the data sequence $x(n)$ is also assumed to be complex valued.

Similarly, the IDFT becomes

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-nk}, \qquad 0 \le n \le N-1 \tag{8.1.3}$$

Since the DFT and IDFT involve basically the same type of computations, our discussion of efficient computational algorithms for the DFT applies as well to the efficient computation of the IDFT.

We observe that for each value of $k$, direct computation of $X(k)$ involves $N$ complex multiplications ($4N$ real multiplications) and $N-1$ complex additions ($4N-2$ real additions). Consequently, to compute all $N$ values of the DFT requires $N^2$ complex multiplications and $N^2 - N$ complex additions.

Direct computation of the DFT is basically inefficient, primarily because it does not exploit the symmetry and periodicity properties of the phase factor $W_N$. In particular, these two properties are:

$$\text{Symmetry property:} \qquad W_N^{k+N/2} = -W_N^k \tag{8.1.4}$$

$$\text{Periodicity property:} \qquad W_N^{k+N} = W_N^k \tag{8.1.5}$$

The computationally efficient algorithms described in this section, known collectively as fast Fourier transform (FFT) algorithms, exploit these two basic properties of the phase factor.

### 8.1.1 Direct Computation of the DFT

For a complex-valued sequence $x(n)$ of $N$ points, the DFT may be expressed as

$$X_R(k) = \sum_{n=0}^{N-1} \left[ x_R(n) \cos \frac{2\pi kn}{N} + x_I(n) \sin \frac{2\pi kn}{N} \right] \tag{8.1.6}$$

$$X_I(k) = -\sum_{n=0}^{N-1} \left[ x_R(n) \sin \frac{2\pi kn}{N} - x_I(n) \cos \frac{2\pi kn}{N} \right] \tag{8.1.7}$$

The direct computation of (8.1.6) and (8.1.7) requires:

1. $2N^2$ evaluations of trigonometric functions.
2. $4N^2$ real multiplications.
3. $4N(N-1)$ real additions.
4. A number of indexing and addressing operations.

These operations are typical of DFT computational algorithms. The operations in items 2 and 3 result in the DFT values $X_R(k)$ and $X_I(k)$. The indexing and addressing operations are necessary to fetch the data $x(n), 0 \le n \le N-1$, and the phase factors and to store the results. The variety of DFT algorithms optimize each of these computational processes in a different way.

### 8.1.2 Divide-and-Conquer Approach to Computation of the DFT

The development of computationally efficient algorithms for the DFT is made possible if we adopt a divide-and-conquer approach. This approach is based on the decomposition of an $N$-point DFT into successively smaller DFTs. This basic approach leads to a family of computationally efficient algorithms known collectively as FFT algorithms.

To illustrate the basic notions, let us consider the computation of an $N$-point DFT, where $N$ can be factored as a product of two integers, that is,

$$N = LM \tag{8.1.8}$$

The assumption that $N$ is not a prime number is not restrictive, since we can pad any sequence with zeros to ensure a factorization of the form (8.1.8).

Now the sequence $x(n), 0 \le n \le N-1$, can be stored either in a one-dimensional array indexed by $n$ or as a two-dimensional array indexed by $l$ and $m$, where $0 \le l \le L-1$ and $0 \le m \le M-1$ as illustrated in Fig. 8.1.1. Note that $l$ is the row index and $m$ is the column index. Thus, the sequence $x(n)$ can be stored in a rectangular

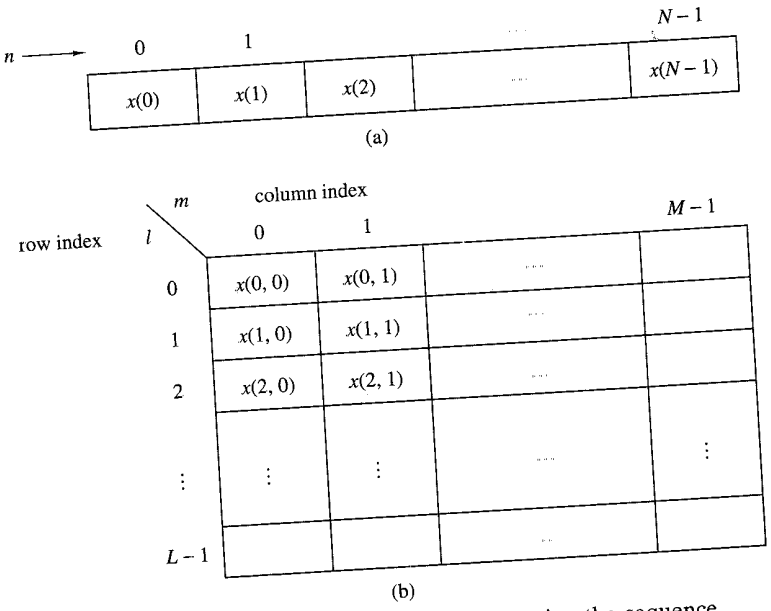



**Figure 8.1.1** Two dimensional data array for storing the sequence $x(n)$, $0 \le n \le N-1$.

array in a variety of ways, each of which depends on the mapping of index $n$ to the indexes $(l, m)$.

For example, suppose that we select the mapping

$$n = Ml + m \tag{8.1.9}$$

This leads to an arrangement in which the first row consists of the first $M$ elements of $x(n)$, the second row consists of the next $M$ elements of $x(n)$, and so on, as illustrated in Fig. 8.1.2(a). On the other hand, the mapping

$$n = l + mL \tag{8.1.10}$$

stores the first $L$ elements of $x(n)$ in the first column, the next $L$ elements in the second column, and so on, as illustrated in Fig. 8.1.2(b).

Row-wise $\quad n = Ml + m$

| $l \backslash m$ | 0 | 1 | 2 | | $M-1$ |
|---|---|---|---|---|---|
| 0 | $x(0)$ | $x(1)$ | $x(2)$ | ... | $x(M-1)$ |
| 1 | $x(M)$ | $x(M+1)$ | $x(M+2)$ | ... | $x(2M-1)$ |
| 2 | $x(2M)$ | $x(2M+1)$ | $x(2M+2)$ | ... | $x(3M-1)$ |
| | $\vdots$ | $\vdots$ | $\vdots$ | ... | $\vdots$ |
| $L-1$ | $x((L-1)M)$ | $x((L-1)M+1)$ | $x((L-1)M+2)$ | ... | $x(LM-1)$ |

(a)

Column-wise $\quad n = l + mL$

| $l \backslash m$ | 0 | 1 | 2 | | $M-1$ |
|---|---|---|---|---|---|
| 0 | $x(0)$ | $x(L)$ | $x(2L)$ | ... | $x((M-1)L)$ |
| 1 | $x(1)$ | $x(L+1)$ | $x(2L+1)$ | ... | $x((M-1)L+1)$ |
| 2 | $x(2)$ | $x(L+2)$ | $x(2L+2)$ | ... | $x((M-1)L+2)$ |
| | $\vdots$ | $\vdots$ | $\vdots$ | ... | $\vdots$ |
| $L-1$ | $x(L-1)$ | $x(2L-1)$ | $x(3L-1)$ | ... | $x(LM-1)$ |

(b)

**Figure 8.1.2** Two arrangements for the data arrays.

A similar arrangement can be used to store the computed DFT values. In particular, the mapping is from the index $k$ to a pair of indices $(p, q)$, where $0 \le p \le L-1$ and $0 \le q \le M-1$. If we select the mapping

$$k = Mp + q \tag{8.1.11}$$

the DFT is stored on a row-wise basis, where the first row contains the first $M$ elements of the DFT $X(k)$, the second row contains the next set of $M$ elements, and so on. On the other hand, the mapping

$$k = qL + p \tag{8.1.12}$$

results in a column-wise storage of $X(k)$, where the first $L$ elements are stored in the first column, the second set of $L$ elements are stored in the second column, and so on.

Now suppose that $x(n)$ is mapped into the rectangular array $x(l, m)$ and $X(k)$ is mapped into a corresponding rectangular array $X(p, q)$. Then the DFT can be expressed as a double sum over the elements of the rectangular array multiplied by the corresponding phase factors. To be specific, let us adopt a column-wise mapping for $x(n)$ given by (8.1.10) and the row-wise mapping for the DFT given by (8.1.11). Then

$$X(p, q) = \sum_{m=0}^{M-1} \sum_{l=0}^{L-1} x(l, m) W_N^{(Mp+q)(mL+l)} \tag{8.1.13}$$

But

$$W_N^{(Mp+q)(mL+l)} = W_N^{MLmp} W_N^{mLq} W_N^{Mpl} W_N^{lq} \tag{8.1.14}$$

However, $W_N^{Nmp} = 1$, $W_N^{mqL} = W_{N/L}^{mq} = W_M^{mq}$, and $W_N^{Mpl} = W_{N/M}^{pl} = W_L^{pl}$.

With these simplications, (8.1.13) can be expressed as

$$X(p, q) = \sum_{l=0}^{L-1} \left\{ W_N^{lq} \left[ \sum_{m=0}^{M-1} x(l, m) W_M^{mq} \right] \right\} W_L^{lp} \tag{8.1.15}$$

The expression in (8.1.15) involves the computation of DFTs of length $M$ and length $L$. To elaborate, let us subdivide the computation into three steps:

1. First, we compute the $M$-point DFTs

$$F(l, q) \equiv \sum_{m=0}^{M-1} x(l, m) W_M^{mq}, \qquad 0 \le q \le M-1 \tag{8.1.16}$$

for each of the rows $l = 0, 1, \ldots, L-1$.

2. Second, we compute a new rectangular array $G(l, q)$ defined as

$$G(l, q) = W_N^{lq} F(l, q), \qquad \begin{array}{l} 0 \le l \le L-1 \\ 0 \le q \le M-1 \end{array} \tag{8.1.17}$$

3. Finally, we compute the $L$-point DFTs

$$X(p,q) = \sum_{l=0}^{L-1} G(l,q) W_L^{lp} \tag{8.1.18}$$

for each column $q = 0, 1, \ldots, M-1$, of the array $G(l,q)$.

On the surface it may appear that the computational procedure outlined above is more complex than the direct computation of the DFT. However, let us evaluate the computational complexity of (8.1.15). The first step involves the computation of $L$ DFTs, each of $M$ points. Hence this step requires $LM^2$ complex multiplications and $LM(M-1)$ complex additions. The second step requires $LM$ complex multiplications. Finally, the third step in the computation requires $ML^2$ complex multiplications and $ML(L-1)$ complex additions. Therefore, the computational complexity is

$$\begin{array}{ll} \text{Complex multiplications:} & N(M+L+1) \\ \text{Complex additions:} & N(M+L-2) \end{array} \tag{8.1.19}$$

where $N = ML$. Thus the number of multiplications has been reduced from $N^2$ to $N(M+L+1)$ and the number of additions has been reduced from $N(N-1)$ to $N(M+L-2)$.

For example, suppose that $N = 1000$ and we select $L = 2$ and $M = 500$. Then, instead of having to perform $10^6$ complex multiplications via direct computation of the DFT, this approach leads to 503,000 complex multiplications. This represents a reduction by approximately a factor of 2. The number of additions is also reduced by about a factor of 2.

When $N$ is a highly composite number, that is, $N$ can be factored into a product of prime numbers of the form

$$N = r_1 r_2 \cdots r_\nu \tag{8.1.20}$$

then the decomposition above can be repeated $(\nu - 1)$ more times. This procedure results in smaller DFTs, which, in turn, leads to a more efficient computational algorithm.

In effect, the first segmentation of the sequence $x(n)$ into a rectangular array of $M$ columns with $L$ elements in each column resulted in DFTs of sizes $L$ and $M$. Further decomposition of the data in effect involves the segmentation of each row (or column) into smaller rectangular arrays which result in smaller DFTs. This procedure terminates when $N$ is factored into its prime factors.

### EXAMPLE 8.1.1

To illustrate this computational procedure, let us consider the computation of an $N = 15$ point DFT. Since $N = 5 \times 3 = 15$, we select $L = 5$ and $M = 3$. In other words, we store the 15-point sequence $x(n)$ column-wise as follows:

$$\begin{array}{llll} \text{Row 1:} & x(0,0) = x(0) & x(0,1) = x(5) & x(0,2) = x(10) \\ \text{Row 2:} & x(1,0) = x(1) & x(1,1) = x(6) & x(1,2) = x(11) \\ \text{Row 3:} & x(2,0) = x(2) & x(2,1) = x(7) & x(2,2) = x(12) \\ \text{Row 4:} & x(3,0) = x(3) & x(3,1) = x(8) & x(3,2) = x(13) \\ \text{Row 5:} & x(4,0) = x(4) & x(4,1) = x(9) & x(4,2) = x(14) \end{array}$$

Now, we compute the three-point DFTs for each of the five rows. This leads to the following $5 \times 3$ array:

$$\begin{array}{ccc} F(0,0) & F(0,1) & F(0,2) \\ F(1,0) & F(1,1) & F(1,2) \\ F(2,0) & F(2,1) & F(2,2) \\ F(3,0) & F(3,1) & F(3,2) \\ F(4,0) & F(4,1) & F(4,2) \end{array}$$

The next step is to multiply each of the terms $F(l,q)$ by the phase factors $W_N^{lq} = W_{15}^{lq}$, $0 \le l \le 4$ and $0 \le q \le 2$. This computation results in the $5 \times 3$ array:

| Column 1 | Column 2 | Column 3 |
|---|---|---|
| $G(0,0)$ | $G(0,1)$ | $G(0,2)$ |
| $G(1,0)$ | $G(1,1)$ | $G(1,2)$ |
| $G(2,0)$ | $G(2,1)$ | $G(2,2)$ |
| $G(3,0)$ | $G(3,1)$ | $G(3,2)$ |
| $G(4,0)$ | $G(4,1)$ | $G(4,2)$ |

The final step is to compute the five-point DFTs for each of the three columns. This computation yields the desired values of the DFT in the form

$$\begin{array}{ccc} X(0,0) = X(0) & X(0,1) = X(1) & X(0,2) = X(2) \\ X(1,0) = X(3) & X(1,1) = X(4) & X(1,2) = X(5) \\ X(2,0) = X(6) & X(2,1) = X(7) & X(2,2) = X(8) \\ X(3,0) = X(9) & X(3,1) = X(10) & X(3,2) = X(11) \\ X(4,0) = X(12) & X(4,1) = X(13) & X(4,2) = X(14) \end{array}$$

Figure 8.1.3 illustrates the steps in the computation.

It is interesting to view the segmented data sequence and the resulting DFT in terms of one-dimensional arrays. When the input sequence $x(n)$ and the output DFT $X(k)$ in the two-dimensional arrays are read across from row 1 through row 5, we obtain the following sequences:

INPUT ARRAY

$$x(0)\ x(5)\ x(10)\ x(1)\ x(6)\ x(11)\ x(2)\ x(7)\ x(12)\ x(3)\ x(8)\ x(13)\ x(4)\ x(9)\ x(14)$$

OUTPUT ARRAY

$$X(0)\ X(1)\ X(2)\ X(3)\ X(4)\ X(5)\ X(6)\ X(7)\ X(8)\ X(9)\ X(10)\ X(11)\ X(12)\ X(13)\ X(14)$$

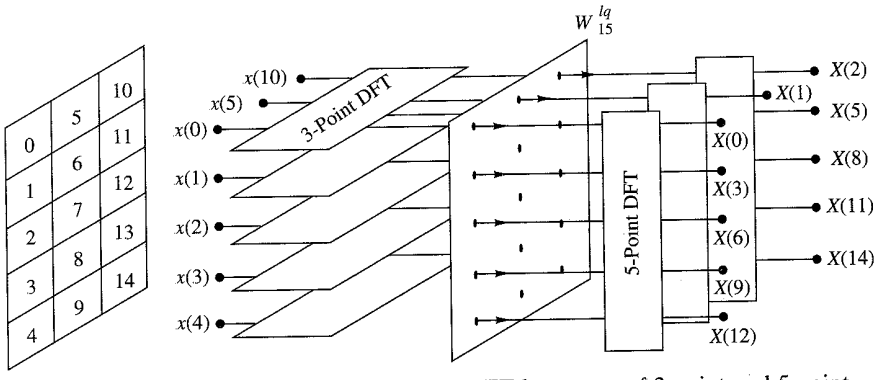



**Figure 8.1.3** Computation of $N = 15$-point DFT by means of 3-point and 5-point DFTs.

We observe that the input data sequence is shuffled from the normal order in the computation of the DFT. On the other hand, the output sequence occurs in normal order. In this case the rearrangement of the input data array is due to the segmentation of the one-dimensional array into a rectangular array and the order in which the DFTs are computed. This shuffling of either the input data sequence or the output DFT sequence is a characteristic of most FFT algorithms.

---

To summarize, the algorithm that we have introduced involves the following computations:

**Algorithm 1**

1. Store the signal column-wise.
2. Compute the $M$-point DFT of each row.
3. Multiply the resulting array by the phase factors $W_N^{lq}$.
4. Compute the $L$-point DFT of each column
5. Read the resulting array row-wise.

An additional algorithm with a similar computational structure can be obtained if the input signal is stored row-wise and the resulting transformation is column-wise. In this case we select

$$\begin{aligned} n &= Ml + m \\ k &= qL + p \end{aligned} \tag{8.1.21}$$

This choice of indices leads to the formula for the DFT in the form

$$\begin{aligned} X(p,q) &= \sum_{m=0}^{M-1} \sum_{l=0}^{L-1} x(l,m) W_N^{pm} W_L^{pl} W_M^{qm} \\ &= \sum_{m=0}^{M-1} W_M^{mq} \left[ \sum_{l=0}^{L-1} x(l,m) W_L^{lp} \right] W_N^{mp} \end{aligned} \tag{8.1.22}$$

Thus we obtain a second algorithm.

**Algorithm 2**

1. Store the signal row-wise.
2. Compute the $L$-point DFT at each column.
3. Multiply the resulting array by the factors $W_N^{pm}$.
4. Compute the $M$-point DFT of each row.
5. Read the resulting array column-wise.

The two algorithms given above have the same complexity. However, they differ in the arrangement of the computations. In the following sections we exploit the divide-and-conquer approach to derive fast algorithms when the size of the DFT is restricted to be a power of 2 or a power of 4.

### 8.1.3 Radix-2 FFT Algorithms

In the preceding section we described four algorithms for efficient computation of the DFT based on the divide-and-conquer approach. Such an approach is applicable when the number $N$ of data points is not a prime. In particular, the approach is very efficient when $N$ is highly composite, that is, when $N$ can be factored as $N = r_1 r_2 r_3 \cdots r_\nu$, where the $\{r_j\}$ are prime.

Of particular importance is the case in which $r_1 = r_2 = \cdots = r_\nu \equiv r$, so that $N = r^\nu$. In such a case the DFTs are of size $r$, so that the computation of the $N$-point DFT has a regular pattern. The number $r$ is called the radix of the FFT algorithm.

In this section we describe radix-2 algorithms, which are by far the most widely used FFT algorithms. Radix-4 algorithms are described in the following section.

Let us consider the computation of the $N = 2^\nu$ point DFT by the divide-and-conquer approach specified by (8.1.16) through (8.1.18). We select $M = N/2$ and $L = 2$. This selection results in a split of the $N$-point data sequence into two $N/2$-point data sequences $f_1(n)$ and $f_2(n)$, corresponding to the even-numbered and odd-numbered samples of $x(n)$, respectively, that is,

$$\begin{aligned} f_1(n) &= x(2n) \\ f_2(n) &= x(2n+1), \qquad n = 0, 1, \ldots, \frac{N}{2} - 1 \end{aligned} \tag{8.1.23}$$

Thus $f_1(n)$ and $f_2(n)$ are obtained by decimating $x(n)$ by a factor of 2, and hence the resulting FFT algorithm is called a decimation-in-time algorithm.

Now the $N$-point DFT can be expressed in terms of the DFTs of the decimated sequences as follows:

$$\begin{aligned} X(k) &= \sum_{n=0}^{N-1} x(n) W_N^{kn}, \qquad k = 0, 1, \ldots, N-1 \\ &= \sum_{n \text{ even}} x(n) W_N^{kn} + \sum_{n \text{ odd}} x(n) W_N^{kn} \\ &= \sum_{m=0}^{(N/2)-1} x(2m) W_N^{2mk} + \sum_{m=0}^{(N/2)-1} x(2m+1) W_N^{k(2m+1)} \end{aligned} \tag{8.1.24}$$

But $W_N^2 = W_{N/2}$. With this substitution, (8.1.24) can be expressed as

$$\begin{aligned} X(k) &= \sum_{m=0}^{(N/2)-1} f_1(m) W_{N/2}^{km} + W_N^k \sum_{m=0}^{(N/2)-1} f_2(m) W_{N/2}^{km} \\ &= F_1(k) + W_N^k F_2(k), \qquad k = 0, 1, \ldots, N-1 \end{aligned} \tag{8.1.25}$$

where $F_1(k)$ and $F_2(k)$ are the $N/2$-point DFTs of the sequences $f_1(m)$ and $f_2(m)$, respectively.

Since $F_1(k)$ and $F_2(k)$ are periodic, with period $N/2$, we have $F_1(k+N/2) = F_1(k)$ and $F_2(k+N/2) = F_2(k)$. In addition, the factor $W_N^{k+N/2} = -W_N^k$. Hence (8.1.25) can be expressed as

$$X(k) = F_1(k) + W_N^k F_2(k), \qquad k = 0, 1, \ldots, \frac{N}{2} - 1 \tag{8.1.26}$$

$$X\left(k + \frac{N}{2}\right) = F_1(k) - W_N^k F_2(k), \qquad k = 0, 1, \ldots, \frac{N}{2} - 1 \tag{8.1.27}$$

We observe that the direct computation of $F_1(k)$ requires $(N/2)^2$ complex multiplications. The same applies to the computation of $F_2(k)$. Furthermore, there are $N/2$ additional complex multiplications required to compute $W_N^k F_2(k)$. Hence the computation of $X(k)$ requires $2(N/2)^2 + N/2 = N^2/2 + N/2$ complex multiplications. This first step results in a reduction of the number of multiplications from $N^2$ to $N^2/2 + N/2$, which is about a factor of 2 for $N$ large.

To be consistent with our previous notation, we may define

$$G_1(k) = F_1(k) \quad , \quad k = 0, 1, \ldots, \frac{N}{2} - 1$$

$$G_2(k) = W_N^k F_2(k), \quad k = 0, 1, \ldots, \frac{N}{2} - 1$$

Then the DFT $X(k)$ may be expressed as

$$\begin{aligned} X(k) &= G_1(k) + G_2(k), \quad k = 0, 1, \ldots, \frac{N}{2} - 1 \\ X(k + \frac{N}{2}) &= G_1(k) - G_2(k), \quad k = 0, 1, \ldots, \frac{N}{2} - 1 \end{aligned} \tag{8.1.28}$$

This computation is illustrated in Fig. 8.1.4.

Having performed the decimation-in-time once, we can repeat the process for each of the sequences $f_1(n)$ and $f_2(n)$. Thus $f_1(n)$ would result in the two $N/4$-point sequences

$$\begin{aligned} v_{11}(n) &= f_1(2n), \qquad n = 0, 1, \ldots, \frac{N}{4} - 1 \\ v_{12}(n) &= f_1(2n+1), \quad n = 0, 1, \ldots, \frac{N}{4} - 1 \end{aligned} \tag{8.1.29}$$

and $f_2(n)$ would yield

$$\begin{aligned} v_{21}(n) &= f_2(2n), \qquad n = 0, 1, \ldots, \frac{N}{4} - 1 \\ v_{22}(n) &= f_2(2n+1), \quad n = 0, 1, \ldots, \frac{N}{4} - 1 \end{aligned} \tag{8.1.30}$$

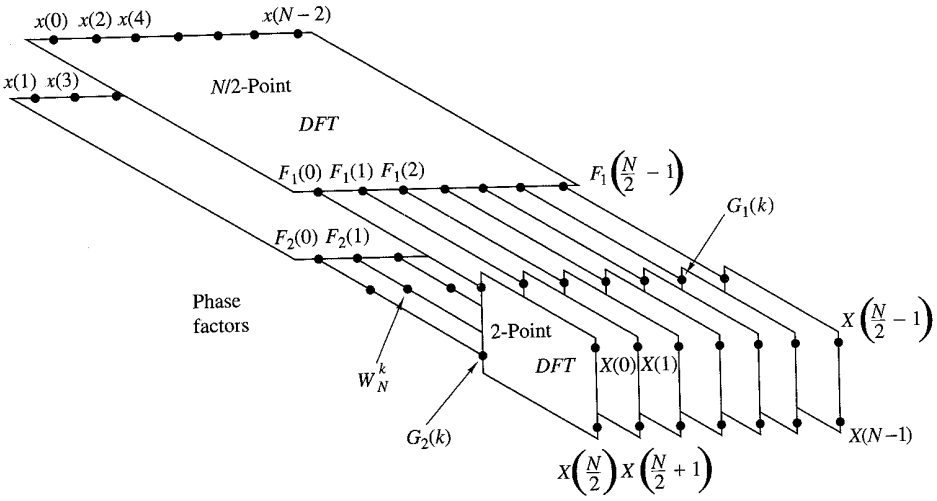



**Figure 8.1.4** First step in the decimation-in-time algorithm.

By computing $N/4$-point DFTs, we would obtain the $N/2$-point DFTs $F_1(k)$ and $F_2(k)$ from the relations

$$F_1(k) = V_{11}(k) + W_{N/2}^k V_{12}(k), \quad k = 0, 1, \ldots, \frac{N}{4} - 1$$

$$F_1\left(k + \frac{N}{4}\right) = V_{11}(k) - W_{N/2}^k V_{12}(k), \quad k = 0, 1, \ldots, \frac{N}{4} - 1 \tag{8.1.31}$$

$$F_2(k) = V_{21}(k) + W_{N/2}^k V_{22}(k), \quad k = 0, 1, \ldots, \frac{N}{4} - 1$$

$$F_2\left(k + \frac{N}{4}\right) = V_{21}(k) - W_{N/2}^k V_{22}(k), \quad k = 0, \ldots, \frac{N}{4} - 1 \tag{8.1.32}$$

where the $\{V_{ij}(k)\}$ are the $N/4$-point DFTs of the sequences $\{v_{ij}(n)\}$.

We observe that the computation of $\{V_{ij}(k)\}$ requires $4(N/4)^2$ multiplications and hence the computation of $F_1(k)$ and $F_2(k)$ can be accomplished with $N^2/4+N/2$ complex multiplications. An additional $N/2$ complex multiplications are required to compute $X(k)$ from $F_1(k)$ and $F_2(k)$. Consequently, the total number of multiplications is reduced approximately by a factor of 2 again to $N^2/4 + N$.

The decimation of the data sequence can be repeated again and again until the resulting sequences are reduced to one-point sequences. For $N = 2^\nu$, this decimation can be performed $\nu = \log_2 N$ times. Thus the total number of complex multiplications is reduced to $(N/2)\log_2 N$. The number of complex additions is $N \log_2 N$. Table 8.1 presents a comparison of the number of complex multiplications in the FFT and in the direct computation of the DFT.

For illustrative purposes, Fig. 8.1.5 depicts the computation of an $N = 8$-point DFT. We observe that the computation is performed in three stages, beginning with the computations of four two-point DFTs, then two four-point DFTs, and finally, one

**TABLE 8.1** Comparison of Computational Complexity for the Direct Computation of the DFT Versus the FFT Algorithm

| Number of Points, $N$ | Complex Multiplications in Direct Computation, $N^2$ | Complex Multiplications in FFT Algorithm, $(N/2)\log_2 N$ | Speed Improvement Factor |
|---|---|---|---|
| 4 | 16 | 4 | 4.0 |
| 8 | 64 | 12 | 5.3 |
| 16 | 256 | 32 | 8.0 |
| 32 | 1,024 | 80 | 12.8 |
| 64 | 4,096 | 192 | 21.3 |
| 128 | 16,384 | 448 | 36.6 |
| 256 | 65,536 | 1,024 | 64.0 |
| 512 | 262,144 | 2,304 | 113.8 |
| 1,024 | 1,048,576 | 5,120 | 204.8 |

eight-point DFT. The combination of the smaller DFTs to form the larger DFT is illustrated in Fig. 8.1.6 for $N = 8$.

Observe that the basic computation performed at every stage, as illustrated in Fig. 8.1.6, is to take two complex numbers, say the pair $(a, b)$, multiply $b$ by $W_N^r$, and then add and subtract the product from $a$ to form two new complex numbers $(A, B)$. This basic computation, which is shown in Fig. 8.1.7, is called a *butterfly* because the flow graph resembles a butterfly.

In general, each butterfly involves one complex multiplication and two complex additions. For $N = 2^\nu$, there are $N/2$ butterflies per stage of the computation process and $\log_2 N$ stages. Therefore, as previously indicated, the total number of complex multiplications is $(N/2)\log_2 N$ and complex additions is $N \log_2 N$.

Once a butterfly operation is performed on a pair of complex numbers $(a, b)$ to produce $(A, B)$, there is no need to save the input pair $(a, b)$. Hence we can store the result $(A, B)$ in the same locations as $(a, b)$. Consequently, we require a fixed amount of storage, namely, $2N$ storage registers, in order to store the results

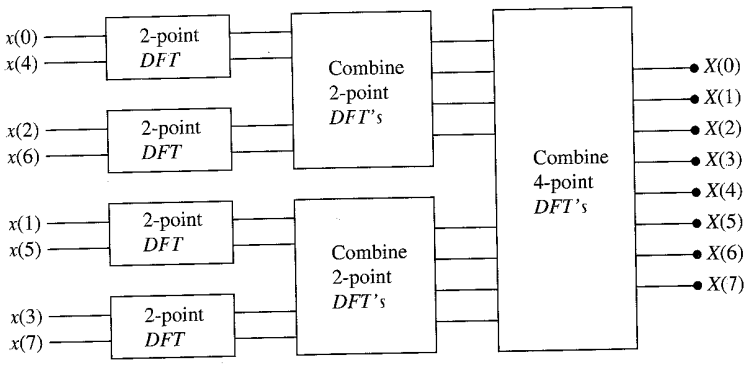



**Figure 8.1.5** Three stages in the computation of an $N = 8$-point DFT.

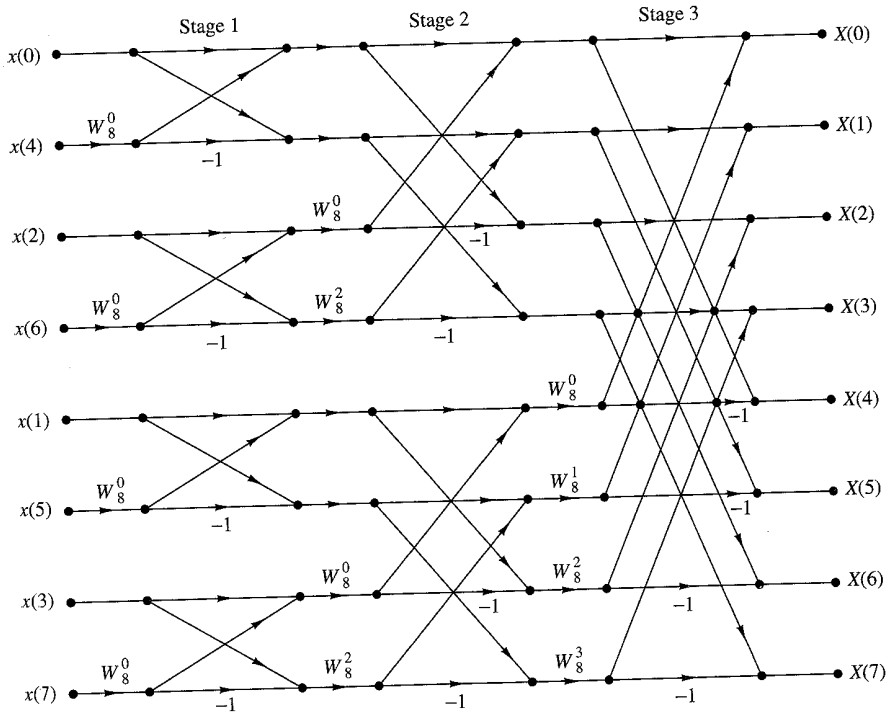



**Figure 8.1.6** Eight-point decimation-in-time FFT algorithm.

($N$ complex numbers) of the computations at each stage. Since the same $2N$ storage locations are used throughout the computation of the $N$-point DFT, we say that *the computations are done in place.*

A second important observation is concerned with the order of the input data sequence after it is decimated $(\nu - 1)$ times. For example, if we consider the case where $N = 8$, we know that the first decimation yields the sequence $x(0), x(2), x(4), x(6), x(1), x(3), x(5), x(7)$, and the second decimation results in the sequence $x(0), x(4), x(2), x(6), x(1), x(5), x(3), x(7)$. This *shuffling* of the input data sequence has a well-defined order as can be ascertained from observing Fig. 8.1.8, which illustrates the decimation of the eight-point sequence. By expressing the index $n$, in the sequence $x(n)$, in binary form, we note that the order of the decimated data sequence is easily obtained by reading the binary representation of the index $n$ in reverse order. Thus the data point $x(3) \equiv x(011)$ is placed in position $m = 110$ or $m = 6$ in the decimated array. Thus we say that the data $x(n)$ after decimation is stored in bit-reversed order.

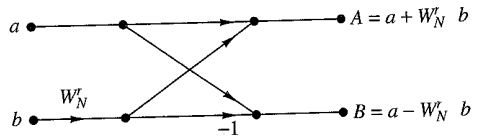
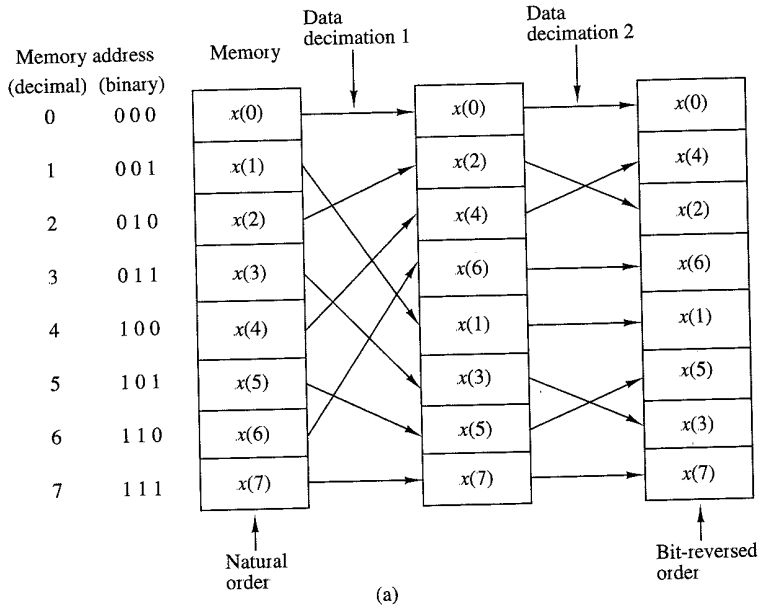



**Figure 8.1.7**
Basic butterfly computation in the decimation-in-time FFT algorithm.




(a)

| $(n_2n_1n_0)$ | $\rightarrow$ | $(n_1n_0n_2)$ | $\rightarrow$ | $(n_0n_1n_2)$ |
|---|---|---|---|---|
| (0 0 0) | $\rightarrow$ | (0 0 0) | $\rightarrow$ | (0 0 0) |
| (0 0 1) | $\rightarrow$ | (0 1 0) | $\rightarrow$ | (1 0 0) |
| (0 1 0) | $\rightarrow$ | (1 0 0) | $\rightarrow$ | (0 1 0) |
| (0 1 1) | $\rightarrow$ | (1 1 0) | $\rightarrow$ | (1 1 0) |
| (1 0 0) | $\rightarrow$ | (0 0 1) | $\rightarrow$ | (0 0 1) |
| (1 0 1) | $\rightarrow$ | (0 1 1) | $\rightarrow$ | (1 0 1) |
| (1 1 0) | $\rightarrow$ | (1 0 1) | $\rightarrow$ | (0 1 1) |
| (1 1 1) | $\rightarrow$ | (1 1 1) | $\rightarrow$ | (1 1 1) |

(b)

**Figure 8.1.8** Shuffling of the data and bit reversal.

With the input data sequence stored in bit-reversed order and the butterfly computations performed in place, the resulting DFT sequence $X(k)$ is obtained in natural order (i.e., $k = 0, 1, \ldots, N-1$). On the other hand, we should indicate that it is possible to arrange the FFT algorithm such that the input is left in natural order and the resulting output DFT will occur in bit-reversed order. Furthermore, we can impose the restriction that both the input data $x(n)$ and the output DFT $X(k)$ be in natural order, and derive an FFT algorithm in which the computations are not done in place. Hence such an algorithm requires additional storage.

Another important radix-2 FFT algorithm, called the decimation-in-frequency algorithm, is obtained by using the divide-and-conquer approach described in Section 8.1.2 with the choice of $M = 2$ and $L = N/2$. This choice of parameters implies a column-wise storage of the input data sequence. To derive the algorithm, we begin by splitting the DFT formula into two summations, of which one involves the sum over the first $N/2$ data points and the second the sum over the last $N/2$ data points.

Thus we obtain

$$X(k) = \sum_{n=0}^{(N/2)-1} x(n)W_N^{kn} + \sum_{n=N/2}^{N-1} x(n)W_N^{kn}$$
$$= \sum_{n=0}^{(N/2)-1} x(n)W_N^{kn} + W_N^{Nk/2} \sum_{n=0}^{(N/2)-1} x\left(n+\frac{N}{2}\right) W_N^{kn} \tag{8.1.33}$$

Since $W_N^{kN/2} = (-1)^k$, the expression (8.1.33) can be rewritten as

$$X(k) = \sum_{n=0}^{(N/2)-1} \left[x(n) + (-1)^k x\left(n+\frac{N}{2}\right)\right] W_N^{kn} \tag{8.1.34}$$

Now, let us split (decimate) $X(k)$ into the even- and odd-numbered samples. Thus we obtain

$$X(2k) = \sum_{n=0}^{(N/2)-1} \left[x(n) + x\left(n+\frac{N}{2}\right)\right] W_{N/2}^{kn}, \qquad k = 0, 1, \ldots, \frac{N}{2}-1 \tag{8.1.35}$$

and

$$X(2k+1) = \sum_{n=0}^{(N/2)-1} \left\{\left[x(n) - x\left(n+\frac{N}{2}\right)\right] W_N^{n}\right\} W_{N/2}^{kn}, \qquad k = 0, 1, \ldots, \frac{N}{2}-1 \tag{8.1.36}$$

where we have used the fact that $W_N^2 = W_{N/2}$.

If we define the $N/2$-point sequences $g_1(n)$ and $g_2(n)$ as

$$g_1(n) = x(n) + x\left(n+\frac{N}{2}\right)$$
$$g_2(n) = \left[x(n) - x\left(n+\frac{N}{2}\right)\right] W_N^{n}, \qquad n = 0, 1, 2, \ldots, \frac{N}{2}-1 \tag{8.1.37}$$

then

$$X(2k) = \sum_{n=0}^{(N/2)-1} g_1(n) W_{N/2}^{kn}$$
$$X(2k+1) = \sum_{n=0}^{(N/2)-1} g_2(n) W_{N/2}^{kn} \tag{8.1.38}$$

The computation of the sequences $g_1(n)$ and $g_2(n)$ according to (8.1.37) and the subsequent use of these sequences to compute the $N/2$-point DFTs are depicted in Fig. 8.1.9. We observe that the basic computation in this figure involves the butterfly operation illustrated in Fig. 8.1.10.

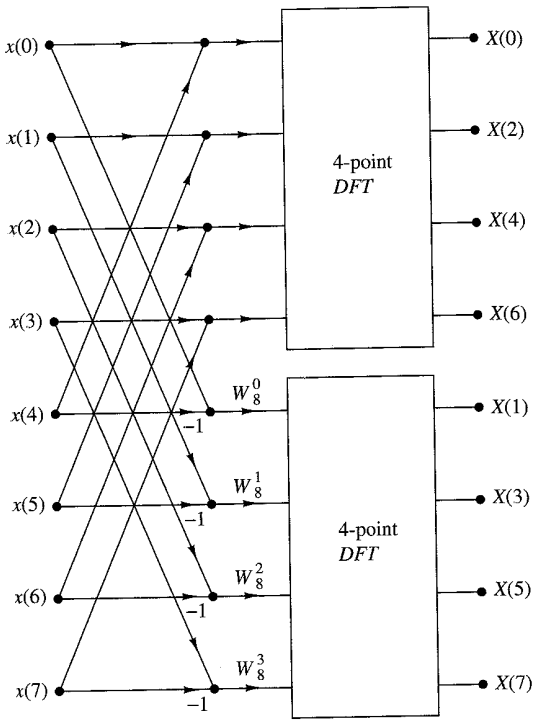



**Figure 8.1.9**
First stage of the decimation-in-frequency FFT algorithm.

This computational procedure can be repeated through decimation of the $N/2$-point DFTs, $X(2k)$ and $X(2k+1)$. The entire process involves $\nu = \log_2 N$ stages of decimation, where each stage involves $N/2$ butterflies of the type shown in Fig. 8.1.10. Consequently, the computation of the $N$-point DFT via the decimation-in-frequency FFT algorithm requires $(N/2)\log_2 N$ complex multiplications and $N \log_2 N$ complex additions, just as in the decimation-in-time algorithm. For illustrative purposes, the eight-point decimation-in-frequency algorithm is given in Fig. 8.1.11.

We observe from Fig. 8.1.11 that the input data $x(n)$ occurs in natural order, but the output DFT occurs in bit-reversed order. We also note that the computations are performed in place. However, it is possible to reconfigure the decimation-in-frequency algorithm so that the input sequence occurs in bit-reversed order while the output DFT occurs in normal order. Furthermore, if we abandon the requirement that the computations be done in place, it is also possible to have both the input data and the output DFT in normal order.

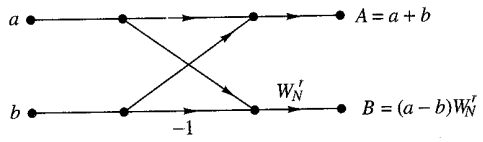



**Figure 8.1.10**
Basic butterfly computation in the decimation-in-frequency FFT algorithm.

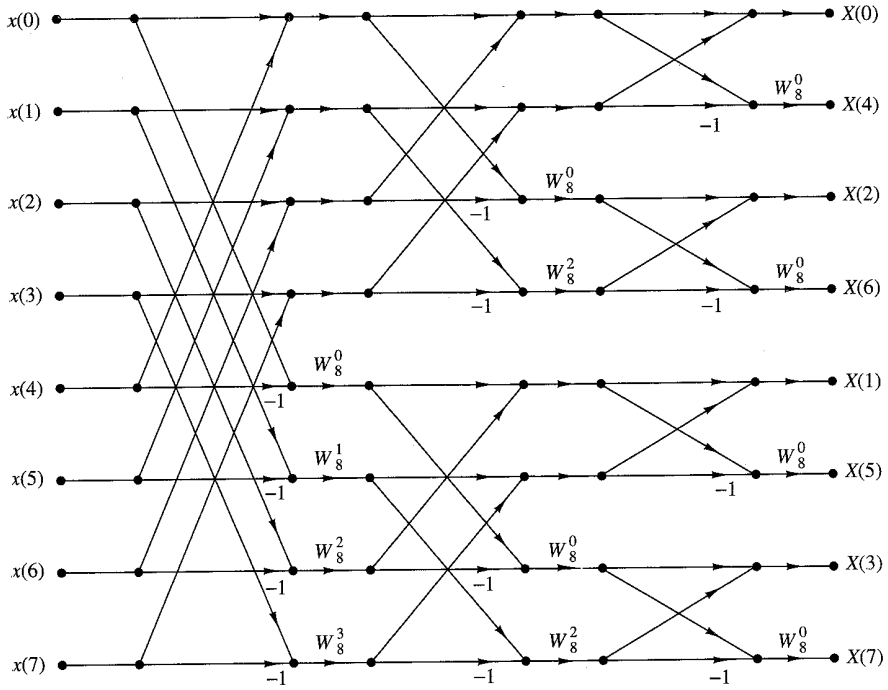



**Figure 8.1.11** $N = 8$-point decimation-in-frequency FFT algorithm.

### 8.1.4 Radix-4 FFT Algorithms

When the number of data points $N$ in the DFT is a power of 4 (i.e., $N = 4^\nu$), we can, of course, always use a radix-2 algorithm for the computation. However, for this case, it is more efficient computationally to employ a radix-4 FFT algorithm.

Let us begin by describing a radix-4 decimation-in-time FFT algorithm, which is obtained by selecting $L = 4$ and $M = N/4$ in the divide-and-conquer approach described in Section 8.1.2. For this choice of $L$ and $M$, we have $l, p = 0, 1, 2, 3$; $m, q = 0, 1, \ldots, N/4 - 1$; $n = 4m + l$; and $k = (N/4)p + q$. Thus we split or decimate the $N$-point input sequence into four subsequences, $x(4n)$, $x(4n+1)$, $x(4n+2)$, $x(4n+3)$, $n = 0, 1, \ldots, N/4 - 1$.

By applying (8.1.15) we obtain

$$X(p,q) = \sum_{l=0}^{3} \left[ W_N^{lq} F(l,q) \right] W_4^{lp}, \qquad p = 0, 1, 2, 3 \tag{8.1.39}$$

where $F(l,q)$ is given by (8.1.16), that is,

$$F(l,q) = \sum_{m=0}^{(N/4)-1} x(l,m) W_{N/4}^{mq}, \qquad \begin{array}{l} l = 0, 1, 2, 3, \\ q = 0, 1, 2, \ldots, \dfrac{N}{4} - 1 \end{array} \tag{8.1.40}$$

and

$$x(l, m) = x(4m + l) \tag{8.1.41}$$

$$X(p, q) = X\left(\frac{N}{4}p + q\right) \tag{8.1.42}$$

Thus, the four $N/4$-point DFTs obtained from (8.1.40) are combined according to (8.1.39) to yield the $N$-point DFT. The expression in (8.1.39) for combining the $N/4$-point DFTs defines a radix-4 decimation-in-time butterfly, which can be expressed in matrix form as

$$\begin{bmatrix} X(0,q) \\ X(1,q) \\ X(2,q) \\ X(3,q) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -j & -1 & j \\ 1 & -1 & 1 & -1 \\ 1 & j & -1 & -j \end{bmatrix} \begin{bmatrix} W_N^0 F(0,q) \\ W_N^q F(1,q) \\ W_N^{2q} F(2,q) \\ W_N^{3q} F(3,q) \end{bmatrix} \tag{8.1.43}$$

The radix-4 butterfly is depicted in Fig. 8.1.12(a) and in a more compact form in Fig. 8.1.12(b). Note that since $W_N^0 = 1$, each butterfly involves three complex multiplications, and 12 complex additions.

This decimation-in-time procedure can be repeated recursively $\nu$ times. Hence the resulting FFT algorithm consists of $\nu$ stages, where each stage contains $N/4$ butterflies. Consequently, the computational burden for the algorithm is $3\nu N/4 = (3N/8)\log_2 N$ complex multiplications and $(3N/2)\log_2 N$ complex additions. We note that the number of multiplications is reduced by 25%, but the number of additions has increased by 50% from $N \log_2 N$ to $(3N/2)\log_2 N$.

It is interesting to note, however, that by performing the additions in two steps, it is possible to reduce the number of additions per butterfly from 12 to 8. This can

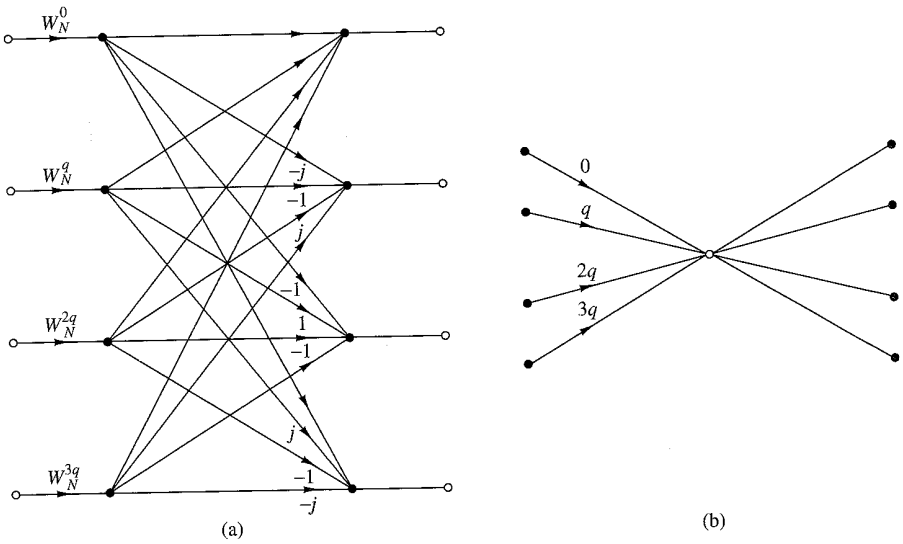



**Figure 8.1.12** Basic butterfly computation in a radix-4 FFT algorithm.

be accomplished by expressing the matrix of the linear transformation in (8.1.43) as a product of two matrices as follows:

$$\begin{bmatrix} X(0,q) \\ X(1,q) \\ X(2,q) \\ X(3,q) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & -j \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & j \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} W_N^0 F(0,q) \\ W_N^q F(1,q) \\ W_N^{2q} F(2,q) \\ W_N^{3q} F(3,q) \end{bmatrix} \tag{8.1.44}$$

Now each matrix multiplication involves four additions for a total of eight additions. Thus the total number of complex additions is reduced to $N \log_2 N$, which is identical to the radix-2 FFT algorithm. The computational savings results from the 25% reduction in the number of complex multiplications.

An illustration of a radix-4 decimation-in-time FFT algorithm is shown in Fig. 8.1.13 for $N = 16$. Note that in this algorithm, the input sequence is in normal order while the output DFT is shuffled. In the radix-4 FFT algorithm, where the decimation is by a factor of 4, the order of the decimated sequence can be determined by reversing the order of the number that represents the index $n$ in a quaternary number system (i.e., the number system based on the digits 0, 1, 2, 3).

A radix-4 decimation-in-frequency FFT algorithm can be obtained by selecting $L = N/4$, $M = 4$; $l, p = 0, 1, \ldots, N/4 - 1$; $m, q = 0, 1, 2, 3$; $n = (N/4)m + l$; and

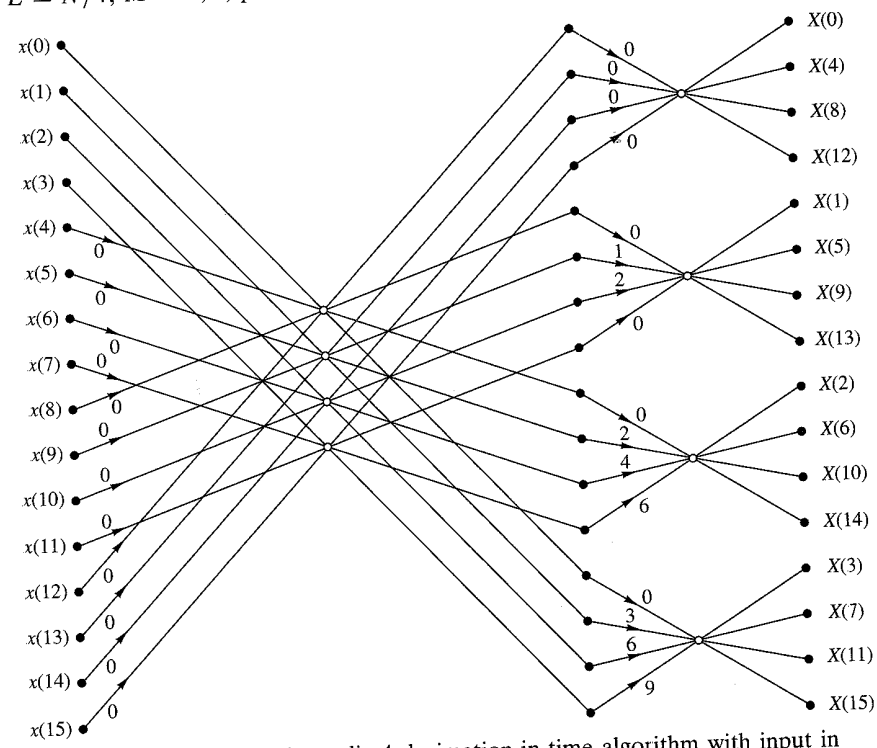



**Figure 8.1.13** Sixteen-point radix-4 decimation-in-time algorithm with input in normal order and output in digit-reversed order. The integer multipliers shown on the graph represent the exponents on $W_{16}$.

$k = 4p + q$. With this choice of parameters, the general equation given by (8.1.15) can be expressed as

$$X(p, q) = \sum_{l=0}^{(N/4)-1} G(l, q) W_{N/4}^{lp} \tag{8.1.45}$$

where

$$G(l, q) = W_N^{lq} F(l, q), \qquad \begin{array}{l} q = 0, 1, 2, 3 \\ l = 0, 1, \ldots, \dfrac{N}{4} - 1 \end{array} \tag{8.1.46}$$

and

$$F(l, q) = \sum_{m=0}^{3} x(l, m) W_4^{mq}, \qquad \begin{array}{l} q = 0, 1, 2, 3 \\ l = 0, 1, 2, 3, \ldots, \dfrac{N}{4} - 1 \end{array} \tag{8.1.47}$$

We note that $X(p, q) = X(4p + q)$, $q = 0, 1, 2, 3$. Consequently, the $N$-point DFT is decimated into four $N/4$-point DFTs and hence we have a decimation-in-frequency FFT algorithm. The computations in (8.1.46) and (8.1.47) define the basic radix-4 butterfly for the decimation-in-frequency algorithm. Note that the multiplications by the factors $W_N^{lq}$ occur after the combination of the data points $x(l, m)$, just as in the case of the radix-2 decimation-in-frequency algorithm.

A 16-point radix-4 decimation-in-frequency FFT algorithm is shown in Fig. 8.1.14. Its input is in normal order and its output is in digit-reversed order. It has exactly the same computational complexity as the decimation-in-time radix-4 FFT algorithm.

For illustrative purposes, let us rederive the radix-4 decimation-in-frequency algorithm by breaking the $N$-point DFT formula into four smaller DFTs. We have

$$\begin{aligned} X(k) &= \sum_{n=0}^{N-1} x(n) W_N^{kn} \\ &= \sum_{n=0}^{N/4-1} x(n) W_N^{kn} + \sum_{n=N/4}^{N/2-1} x(n) W_N^{kn} + \sum_{n=N/2}^{3N/4-1} x(n) W_N^{kn} + \sum_{n=3N/4}^{N-1} x(n) W_N^{kn} \\ &= \sum_{n=0}^{N/4-1} x(n) W_N^{kn} + W_N^{Nk/4} \sum_{n=0}^{N/4-1} x\left(n + \frac{N}{4}\right) W_N^{kn} \\ &\quad + W_N^{kN/2} \sum_{n=0}^{N/4-1} x\left(n + \frac{N}{2}\right) W_N^{nk} + W_N^{3kN/4} \sum_{n=0}^{N/4-1} x\left(n + \frac{3N}{4}\right) W_N^{kn} \end{aligned} \tag{8.1.48}$$

From the definition of the phase factors, we have

$$W_N^{kN/4} = (-j)^k, \qquad W_N^{Nk/2} = (-1)^k, \qquad W_N^{3Nk/4} = (j)^k \tag{8.1.49}$$

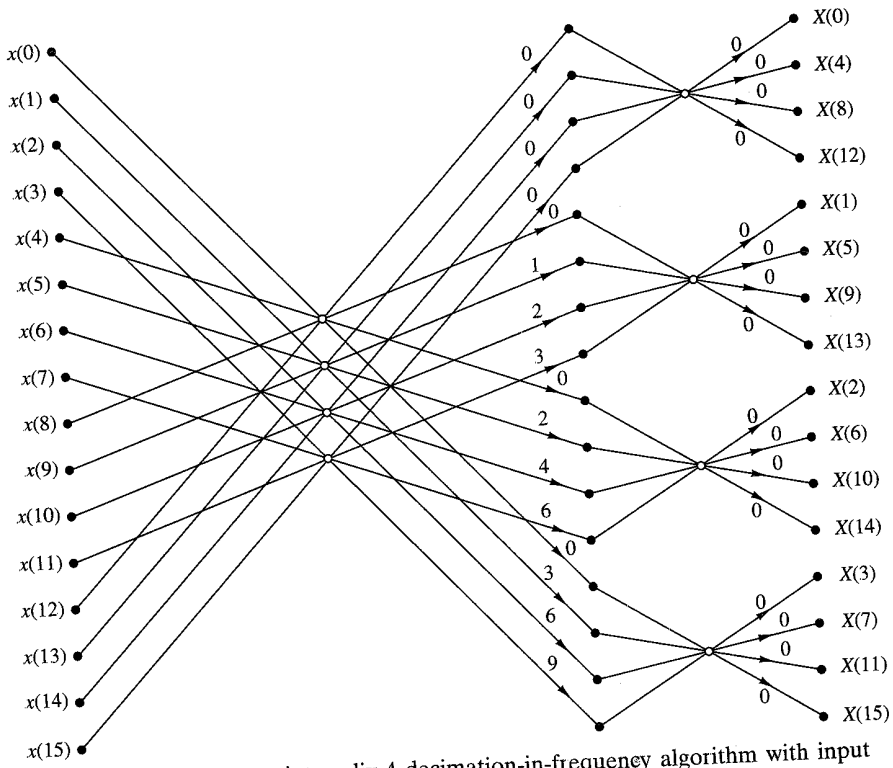



**Figure 8.1.14** Sixteen-point, radix-4 decimation-in-frequency algorithm with input in normal order and output in digit-reversed order.

After substitution of (8.1.49) into (8.1.48), we obtain

$$X(k) = \sum_{n=0}^{N/4-1} \left[ x(n) + (-j)^k x\left(n + \frac{N}{4}\right) + (-1)^k x\left(n + \frac{N}{4}\right) + (j)^k x\left(n + \frac{3N}{4}\right) \right] W_N^{nk} \tag{8.1.50}$$

The relation in (8.1.50) is not an $N/4$-point DFT because the phase factor depends on $N$ and not on $N/4$. To convert it into an $N/4$-point DFT, we subdivide the DFT sequence into four $N/4$-point subsequences, $X(4k)$, $X(4k+1)$, $X(4k+2)$, and $X(4k+3)$, $k = 0, 1, \ldots, N/4-1$. Thus we obtain the radix-4 decimation-in-frequency DFT as

$$X(4k) = \sum_{n=0}^{N/4-1} \left[ x(n) + x\left(n + \frac{N}{4}\right) + x\left(n + \frac{N}{2}\right) + x\left(n + \frac{3N}{4}\right) \right] W_N^0 W_{N/4}^{kn} \tag{8.1.51}$$

$$X(4k+1) = \sum_{n=0}^{N/4-1} \left[ x(n) - jx\left(n + \frac{N}{4}\right) - x\left(n + \frac{N}{2}\right) + jx\left(n + \frac{3N}{4}\right) \right] W_N^n W_{N/4}^{kn} \tag{8.1.52}$$

$$X(4k+2) = \sum_{n=0}^{N/4-1} \left[ x(n) - x\left(n + \frac{N}{4}\right) + x\left(n + \frac{N}{2}\right) - x\left(n + \frac{3N}{4}\right) \right] W_N^{2n} W_{N/4}^{kn} \tag{8.1.53}$$

$$X(4k+3) = \sum_{n=0}^{N/4-1} \left[ x(n) + jx\left(n + \frac{N}{4}\right) - x\left(n + \frac{N}{2}\right) - jx\left(n + \frac{3N}{4}\right) \right] W_N^{3n} W_{N/4}^{kn} \tag{8.1.54}$$

where we have used the property $W_N^{4kn} = W_{N/4}^{kn}$. Note that the input to each $N/4$-point DFT is a linear combination of four signal samples scaled by a phase factor. This procedure is repeated $\nu$ times, where $\nu = \log_4 N$.

### 8.1.5 Split-Radix FFT Algorithms

An inspection of the radix-2 decimation-in-frequency flowgraph shown in Fig. 8.1.11 indicates that the even-numbered points of the DFT can be computed independently of the odd-numbered points. This suggests the possibility of using different computational methods for independent parts of the algorithm with the objective of reducing the number of computations. The split-radix FFT (SRFFT) algorithms exploit this idea by using both a radix-2 and a radix-4 decomposition in the same FFT algorithm.

We illustrate this approach with a decimation-in-frequency SRFFT algorithm due to Duhamel (1986). First, we recall that in the radix-2 decimation-in-frequency FFT algorithm, the even-numbered samples of the $N$-point DFT are given as

$$X(2k) = \sum_{n=0}^{N/2-1} \left[ x(n) + x\left(n + \frac{N}{2}\right) \right] W_{N/2}^{nk}, \qquad k = 0, 1, \ldots, \frac{N}{2} - 1 \tag{8.1.55}$$

Note that these DFT points can be obtained from an $N/2$-point DFT without any additional multiplications. Consequently, a radix-2 suffices for this computation.

The odd-numbered samples $\{X(2k+1)\}$ of the DFT require the premultiplication of the input sequence with the phase factors $W_N^n$. For these samples a radix-4 decomposition produces some computational efficiency because the four-point DFT has the largest multiplication-free butterfly. Indeed, it can be shown that using a radix greater than 4 does not result in a significant reduction in computational complexity.

If we use a radix-4 decimation-in-frequency FFT algorithm for the odd-numbered samples of the $N$-point DFT, we obtain the following $N/4$-point DFTs:

$$X(4k+1) = \sum_{n=0}^{N/4-1} \{[x(n) - x(n+N/2)] - j[x(n+N/4) - x(n+3N/4)]\} W_N^n W_{N/4}^{kn} \tag{8.1.56}$$

$$X(4k+3) = \sum_{n=0}^{N/4-1} \{[x(n) - x(n+N/2)] + j[x(n+N/4) - x(n+3N/4)]\} W_N^{3n} W_{N/4}^{kn} \tag{8.1.57}$$

Thus the $N$-point DFT is decomposed into one $N/2$-point DFT without additional phase factors and two $N/4$-point DFTs with phase factors. The $N$-point DFT is obtained by successive use of these decompositions up to the last stage. Thus we obtain a decimation-in-frequency SRFFT algorithm.

Figure 8.1.15 shows the flow graph for an in-place 32-point decimation-in-frequency SRFFT algorithm. At stage A of the computation for $N = 32$, the top 16 points constitute the sequence

$$g_0(n) = x(n) + x(n+N/2), \qquad 0 \le n \le 15 \tag{8.1.58}$$

This is the sequence required for the computation of $X(2k)$. The next 8 points constitute the sequence

$$g_1(n) = x(n) - x(n+N/2), \qquad 0 \le n \le 7 \tag{8.1.59}$$

The bottom eight points constitute the sequence $jg_2(n)$, where

$$g_2(n) = x(n+N/4) - x(n+3N/4), \qquad 0 \le n \le 7 \tag{8.1.60}$$

The sequences $g_1(n)$ and $g_2(n)$ are used in the computation of $X(4k+1)$ and $X(4k+3)$. Thus, at stage A we have completed the first decimation for the radix-2 component of the algorithm. At stage B, the bottom eight points constitute the computation of $[g_1(n) + jg_2(n)]W_{32}^{3n}$, $0 \le n \le 7$, which is used to compute $X(4k+3)$, $0 \le k \le 7$. The next eight points from the bottom constitute the computation of $[g_1(n) - jg_2(n)]$ $W_{32}^n$, $0 \le n \le 7$, which is used to compute $X(4k+1)$, $0 \le k \le 7$. Thus at stage B, we have completed the first decimation for the radix-4 algorithm, which results in two 8-point sequences. Hence the basic butterfly computation for the SRFFT algorithm has the "L-shaped" form illustrated in Fig. 8.1.16.

Now we repeat the steps in the computation above. Beginning with the top 16 points at stage A, we repeat the decomposition for the 16-point DFT. In other words,

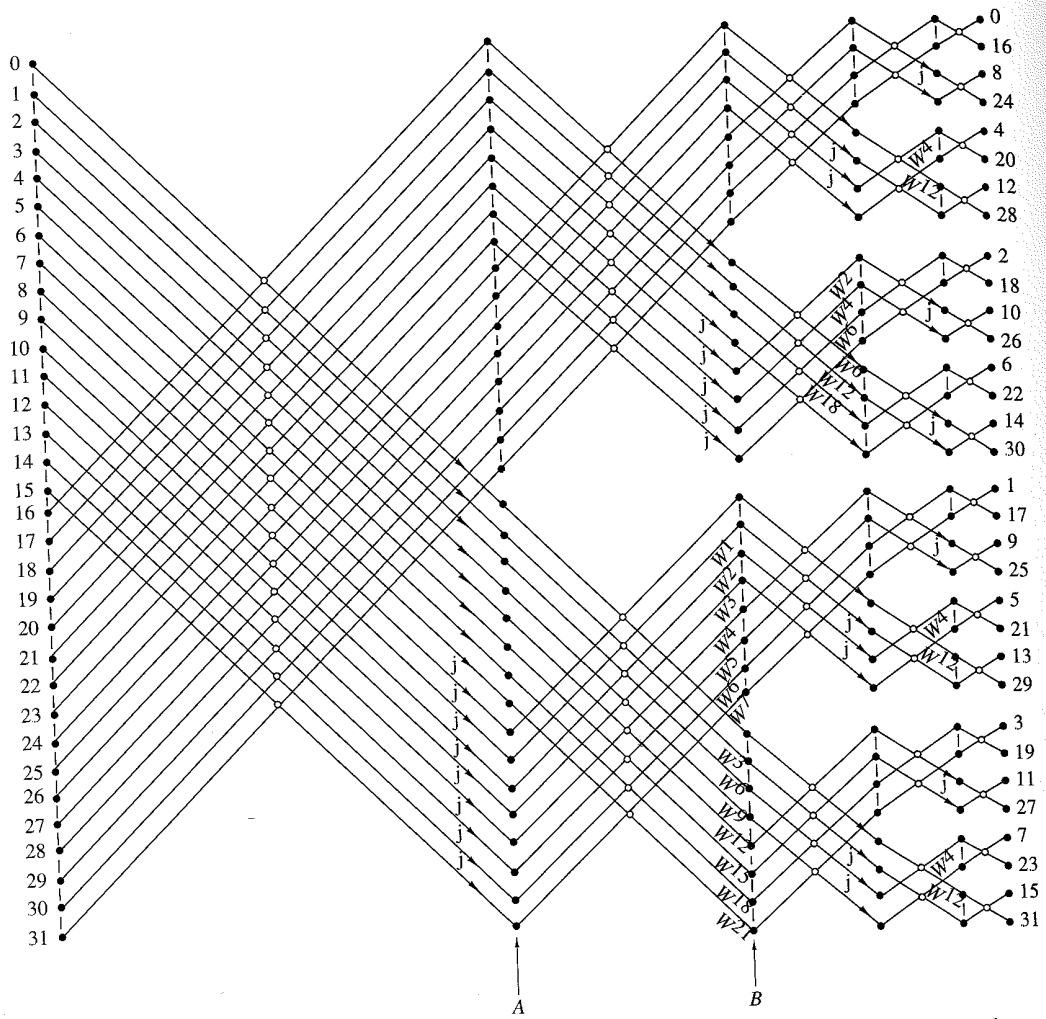



**Figure 8.1.15** Length 32 split-radix FFT algorithms from paper by Duhamel (1986); reprinted with permission from the IEEE.

we decompose the computation into an eight-point, radix-2 DFT and two four-point, radix-4 DFTs. Thus at stage B, the top eight points constitute the sequence (with $N = 16$)

$$g_0'(n) = g_0(n) + g_0(n + N/2), \qquad 0 \le n \le 7 \tag{8.1.61}$$

and the next eight points constitute the two four-point sequences $g_1'(n)$ and $jg_2'(n)$, where

$$\begin{aligned} g_1'(n) &= g_0(n) - g_0(n + N/2), \qquad 0 \le n \le 3 \\ g_2'(n) &= g_0(n + N/4) - g_0(n + 3N/4), \qquad 0 \le n \le 3 \end{aligned} \tag{8.1.62}$$

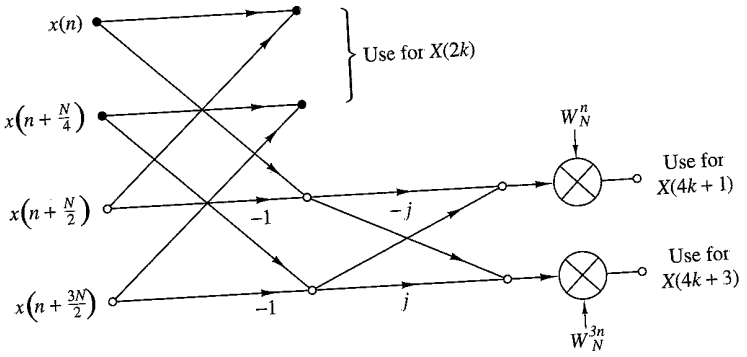



**Figure 8.1.16** Butterfly for SRFFT algorithm.

The bottom 16 points of stage B are in the form of two eight-point DFTs. Hence each eight-point DFT is decomposed into a four-point, radix-2 DFT and a four-point, radix-4 DFT. In the final stage, the computations involve the combination of two-point sequences.

Table 8.2 presents a comparison of the number of *nontrivial* real multiplications and additions required to perform an $N$-point DFT with complex-valued data, using a radix-2, radix-4, radix-8, and a split-radix FFT. Note that the SRFFT algorithm requires the lowest number of multiplication and additions. For this reason, it is preferable in many practical applications.

Another type of SRFFT algorithm has been developed by Price (1990). Its relation to Duhamel's algorithm described previously can be seen by noting that the radix-4 DFT terms $X(4k+1)$ and $X(4k+3)$ involve the $N/4$-point DFTs of the sequences $[g_1(n) - jg_2(n)]W_N^n$ and $[g_1(n) + jg_2(n)]W_N^{3n}$, respectively. In effect, the sequences $g_1(n)$ and $g_2(n)$ are multiplied by the factor (vector) $(1, -j) = (1, W_{32}^8)$

**TABLE 8.2** Number of Nontrivial Real Multiplications and Additions to Compute an $N$-point Complex DFT

| | Real Multiplications | | | | Real Additions | | | |
|---|---|---|---|---|---|---|---|---|
| $N$ | Radix 2 | Radix 4 | Radix 8 | Split Radix | Radix 2 | Radix 4 | Radix 8 | Split Radix |
| 16 | 24 | 20 | | 20 | 152 | 148 | | 148 |
| 32 | 88 | | | 68 | 408 | | | 388 |
| 64 | 264 | 208 | 204 | 196 | 1,032 | 976 | 972 | 964 |
| 128 | 712 | | | 516 | 2,504 | | | 2,308 |
| 256 | 1,800 | 1,392 | | 1,284 | 5,896 | 5,488 | | 5,380 |
| 512 | 4,360 | | 3,204 | 3,076 | 13,566 | | 12,420 | 12,292 |
| 1,024 | 10,248 | 7,856 | | 7,172 | 30,728 | 28,336 | | 27,652 |

*Source:* Extracted from Duhamel (1986).

and by $W_N^n$ for the computation of $X(4k+1)$, while the computation of $X(4k+3)$ involves the factor $(1, j) = (1, W_{32}^{-8})$ and $W_N^{3n}$. Instead, one can rearrange the computation so that the factor for $X(4k+3)$ is $(-j, -1) = -(W_{32}^{-8}, 1)$. As a result of this phase rotation, the phase factors in the computation of $X(4k+3)$ become exactly the same as those for $X(4k+1)$, except that they occur in mirror image order. For example, at stage B of Fig. 8.1.15, the phase factors $W^{21}, W^{18}, \ldots, W^3$ are replaced by $W^1, W^2, \ldots, W^7$, respectively. This mirror-image symmetry occurs at every subsequent stage of the algorithm. As a consequence, the number of phase factors that must be computed and stored is reduced by a factor of 2 in comparison to Duhamel's algorithm. The resulting algorithm is called the "mirror" FFT (MFFT) algorithm.

An additional factor-of-2 savings in storage of phase factors can be obtained by introducing a 90° phase offset at the midpoint of each factor array, which can be removed if necessary at the output of the SRFFT computation. The incorporation of this improvement into the SRFFT (or the MFFT) results in another algorithm, also due to Price (1990), called the "phase" FFT (PFFT) algorithm.

### 8.1.6 Implementation of FFT Algorithms

Now that we have described the basic radix-2 and radix-4 FFT algorithms, let us consider some of the implementation issues. Our remarks apply directly to radix-2 algorithms, although similar comments may be made about radix-4 and higher-radix algorithms.

Basically, the radix-2 FFT algorithm consists of taking two data points at a time from memory, performing the butterfly computations and returning the resulting numbers to memory. This procedure is repeated many times ($(N \log_2 N)/2$ times) in the computation of an $N$-point DFT.

The butterfly computations require the phase factors $\{W_N^k\}$ at various stages in either natural or bit-reversed order. In an efficient implementation of the algorithm, the phase factors are computed once and stored in a table, either in normal order or in bit-reversed order, depending on the specific implementation of the algorithm.

Memory requirement is another factor that must be considered. If the computations are performed in place, the number of memory locations required is $2N$ since the numbers are complex. However, we can instead double the memory to 4N, thus simplifying the indexing and control operations in the FFT algorithms. In this case we simply alternate in the use of the two sets of memory locations from one stage of the FFT algorithm to the other. Doubling of the memory also allows us to have both the input sequence and the output sequence in normal order.

There are a number of other implementation issues regarding indexing, bit reversal, and the degree of parallelism in the computations. To a large extent, these issues are a function of the specific algorithm and the type of implementation, namely, a hardware or software implementation. In implementations based on a fixed-point arithmetic, or floating-point arithmetic on small machines, there is also the issue of round-off errors in the computation. This topic is considered in Section 8.4.

Although the FFT algorithms described previously were presented in the context of computing the DFT efficiently, they can also be used to compute the IDFT, which is

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k)W_N^{-nk} \tag{8.1.63}$$

The only difference between the two transforms is the normalization factor $1/N$ and the sign of the phase factor $W_N$. Consequently, an FFT algorithm for computing the DFT can be converted to an FFT algorithm for computing the IDFT by changing the sign on all the phase factors and dividing the final output of the algorithm by $N$.

In fact, if we take the decimation-in-time algorithm that we described in Section 8.1.3, reverse the direction of the flow graph, change the sign on the phase factors, interchange the output and input, and finally, divide the output by $N$, we obtain a decimation-in-frequency FFT algorithm for computing the IDFT. On the other hand, if we begin with the decimation-in-frequency FFT algorithm described in Section 8.1.3 and repeat the changes described above, we obtain a decimation-in-time FFT algorithm for computing the IDFT. Thus it is a simple matter to devise FFT algorithms for computing the IDFT.

Finally, we note that the emphasis in our discussion of FFT algorithms was on radix-2, radix-4, and split-radix algorithms. These are by far the most widely used in practice. When the number of data points is not a power of 2 or 4, it is a simple matter to pad the sequence $x(n)$ with zeros such that $N = 2^\nu$ or $N = 4^\nu$.

The measure of complexity for FFT algorithms that we have emphasized is the required number of arithmetic operations (multiplications and additions). Although this is a very important benchmark for computational complexity, there are other issues to be considered in practical implementation of FFT algorithms. These include the architecture of the processor, the available instruction set, the data structures for storing phase factors, and other considerations.

For general-purpose computers, where the cost of the numerical operations dominates, radix-2, radix-4, and split-radix FFT algorithms are good candidates. However, in the case of special-purpose digital signal processors, featuring single-cycle multiply-and-accumulate operation, bit-reversed addressing, and a high degree of instruction parallelism, the structural regularity of the algorithm is equally as important as arithmetic complexity. Hence for DSP processors, radix-2 or radix-4 decimation-in-frequency FFT algorithms are preferable in terms of speed and accuracy. The irregular structure of the SRFFT may render it less suitable for implementation on digital signal processors. Structural regularity is also important in the implementation of FFT algorithms on vector processors, multiprocessors, and in VLSI. Interprocessor communication is an important consideration in such implementations on parallel processors.

In conclusion, we have presented several important considerations in the implementation of FFT algorithms. Advances in digital signal processing technology, in hardware and software, will continue to influence the choice among FFT algorithms for various practical applications.

## 8.2 Applications of FFT Algorithms

The FFT algorithms described in the preceding section find application in a variety of areas, including linear filtering, correlation, and spectrum analysis. Basically, the FFT algorithm is used as an efficient means to compute the DFT and the IDFT.

In this section we consider the use of the FFT algorithm in linear filtering and in the computation of the crosscorrelation of two sequences. The use of the FFT in spectrum estimation is considered in Chapter 14. In addition we illustrate how to enhance the efficiency of the FFT algorithm by forming complex-valued sequences from real-valued sequences prior to the computation of the DFT.

### 8.2.1 Efficient Computation of the DFT of Two Real Sequences

The FFT algorithm is designed to perform complex multiplications and additions, even though the input data may be real valued. The basic reason for this situation is that the phase factors are complex and hence, after the first stage of the algorithm, all variables are basically complex valued.

In view of the fact that the algorithm can handle complex-valued input sequences, we can exploit this capability in the computation of the DFT of two real-valued sequences.

Suppose that $x_1(n)$ and $x_2(n)$ are two real-valued sequences of length $N$, and let $x(n)$ be a complex-valued sequence defined as

$$x(n) = x_1(n) + jx_2(n), \qquad 0 \le n \le N-1 \tag{8.2.1}$$

The DFT operation is linear and hence the DFT of $x(n)$ can be expressed as

$$X(k) = X_1(k) + jX_2(k) \tag{8.2.2}$$

The sequences $x_1(n)$ and $x_2(n)$ can be expressed in terms of $x(n)$ as follows:

$$x_1(n) = \frac{x(n) + x^*(n)}{2} \tag{8.2.3}$$

$$x_2(n) = \frac{x(n) - x^*(n)}{2j} \tag{8.2.4}$$

Hence the DFTs of $x_1(n)$ and $x_2(n)$ are

$$X_1(k) = \frac{1}{2}\{\text{DFT}[x(n)] + \text{DFT}[x^*(n)]\} \tag{8.2.5}$$

$$X_2(k) = \frac{1}{2j}\{\text{DFT}[x(n)] - \text{DFT}[x^*(n)]\} \tag{8.2.6}$$

Recall that the DFT of $x^*(n)$ is $X^*(N-k)$. Therefore,

$$X_1(k) = \frac{1}{2}[X(k) + X^*(N-k)] \tag{8.2.7}$$

$$X_2(k) = \frac{1}{j2}[X(k) - X^*(N-k)] \tag{8.2.8}$$

Thus, by performing a single DFT on the complex-valued sequence $x(n)$, we have obtained the DFT of the two real sequences with only a small amount of additional computation that is involved in computing $X_1(k)$ and $X_2(k)$ from $X(k)$ by use of (8.2.7) and (8.2.8).

### 8.2.2 Efficient Computation of the DFT of a $2N$-Point Real Sequence

Suppose that $g(n)$ is a real-valued sequence of $2N$ points. We now demonstrate how to obtain the $2N$-point DFT of $g(n)$ from computation of one $N$-point DFT involving complex-valued data. First, we define

$$\begin{aligned} x_1(n) &= g(2n) \\ x_2(n) &= g(2n+1) \end{aligned} \tag{8.2.9}$$

Thus we have subdivided the $2N$-point real sequence into two $N$-point real sequences. Now we can apply the method described in the preceding section.

Let $x(n)$ be the $N$-point complex-valued sequence

$$x(n) = x_1(n) + jx_2(n) \tag{8.2.10}$$

From the results of the preceding section, we have

$$\begin{aligned} X_1(k) &= \frac{1}{2}[X(k) + X^*(N-k)] \\ X_2(k) &= \frac{1}{2j}[X(k) - X^*(N-k)] \end{aligned} \tag{8.2.11}$$

Finally, we must express the $2N$-point DFT in terms of the two $N$-point DFTs, $X_1(k)$ and $X_2(k)$. To accomplish this, we proceed as in the decimation-in-time FFT algorithm, namely,

$$\begin{aligned} G(k) &= \sum_{n=0}^{N-1} g(2n) W_{2N}^{2nk} + \sum_{n=0}^{N-1} g(2n+1) W_{2N}^{(2n+1)k} \\ &= \sum_{n=0}^{N-1} x_1(n) W_N^{nk} + W_{2N}^k \sum_{n=0}^{N-1} x_2(n) W_N^{nk} \end{aligned}$$

Consequently,

$$\begin{aligned} G(k) &= X_1(k) + W_2^k N X_2(k), \qquad k = 0, 1, \ldots, N-1 \\ G(k+N) &= X_1(k) - W_2^k N X_2(k), \qquad k = 0, 1, \ldots, N-1 \end{aligned} \tag{8.2.12}$$

Thus we have computed the DFT of a $2N$-point real sequence from one $N$-point DFT and some additional computation as indicated by (8.2.11) and (8.2.12).

### 8.2.3 Use of the FFT Algorithm in Linear Filtering and Correlation

An important application of the FFT algorithm is in FIR linear filtering of long data sequences. In Chapter 7 we described two methods, the overlap-add and the overlap-save methods for filtering a long data sequence with an FIR filter, based on the use of the DFT. In this section we consider the use of these two methods in conjunction with the FFT algorithm for computing the DFT and the IDFT.

Let $h(n)$, $0 \le n \le M-1$, be the unit sample response of the FIR filter and let $x(n)$ denote the input data sequence. The block size of the FFT algorithm is $N$, where $N = L + M - 1$ and $L$ is the number of new data samples being processed by the filter. We assume that for any given value of $M$, the number $L$ of data samples is selected so that $N$ is a power of 2. For purposes of this discussion, we consider only radix-2 FFT algorithms.

The $N$-point DFT of $h(n)$, which is padded by $L-1$ zeros, is denoted as $H(k)$. This computation is performed once via the FFT and the resulting $N$ complex numbers are stored. To be specific we assume that the decimation-in-frequency FFT algorithm is used to compute $H(k)$. This yields $H(k)$ in bit-reversed order, which is the way it is stored in memory.

In the overlap-save method, the first $M-1$ data points of each data block are the last $M-1$ data points of the previous data block. Each data block contains $L$ new data points, such that $N = L + M - 1$. The $N$-point DFT of each data block is performed by the FFT algorithm. If the decimation-in-frequency algorithm is employed, the input data block requires no shuffling and the values of the DFT occur in bit-reversed order. Since this is exactly the order of $H(k)$, we can multiply the DFT of the data, say $X_m(k)$, with $H(k)$, and thus the result

$$Y_m(k) = H(k)X_m(k)$$

is also in bit-reversed order.

The inverse DFT (IDFT) can be computed by use of an FFT algorithm that takes the input in bit-reversed order and produces an output in normal order. Thus there is no need to shuffle any block of data in computing either the DFT or the IDFT.

If the overlap-add method is used to perform the linear filtering, the computational method using the FFT algorithm is basically the same. The only difference is that the $N$-point data blocks consist of $L$ new data points and $M-1$ additional zeros. After the IDFT is computed for each data block, the $N$-point filtered blocks are overlapped as indicated in Section 7.3.2, and the $M-1$ overlapping data points between successive output records are added together.

Let us assess the computational complexity of the FFT method for linear filtering. For this purpose, the one-time computation of $H(k)$ is insignificant and can be ignored. Each FFT requires $(N/2)\log_2 N$ complex multiplications and $N\log_2 N$ additions. Since the FFT is performed twice, once for the DFT and once for the IDFT, the computational burden is $N\log_2 N$ complex multiplications and $2N\log_2 N$ additions. There are also $N$ complex multiplications and $N-1$ additions required to compute $Y_m(k)$. Therefore, we have $(N\log_2 2N)/L$ complex multiplications per output data point and approximately $(2N\log_2 2N)/L$ additions per output data point.

The overlap-add method requires an incremental increase of $(M-1)/L$ in the number of additions.

By way of comparison, a direct-form realization of the FIR filter involves $M$ real multiplications per output point if the filter is not linear phase, and $M/2$ if it is linear phase (symmetric). Also, the number of additions is $M-1$ per output point (see Sec. 10.2).

It is interesting to compare the efficiency of the FFT algorithm with the direct form realization of the FIR filter. Let us focus on the number of multiplications, which are more time consuming than additions. Suppose that $M = 128 = 2^7$ and $N = 2^\nu$. Then the number of complex multiplications per output point for an FFT size of $N = 2^\nu$ is

$$c(\nu) = \frac{N \log_2 2N}{L} = \frac{2^\nu(\nu+1)}{N-M+1}$$
$$\approx \frac{2^\nu(\nu+1)}{2^\nu - 2^7}$$

The values of $c(\nu)$ for different values of $\nu$ are given in Table 8.3. We observe that there is an optimum value of $\nu$ which minimizes $c(\nu)$. For the FIR filter of size $M = 128$, the optimum occurs at $\nu = 10$.

We should emphasize that $c(\nu)$ represents the number of complex multiplications for the FFT-based method. The number of real multiplications is four times this number. However, even if the FIR filter has linear phase (see Sec. 10.2), the number of computations per output point is still less with the FFT-based method. Furthermore, the efficiency of the FFT method can be improved by computing the DFT of two successive data blocks simultaneously, according to the method just described. Consequently, the FFT-based method is indeed superior from a computational point of view when the filter length is relatively large.

The computation of the cross correlation between two sequences by means of the FFT algorithm is similar to the linear FIR filtering problem just described. In practical applications involving crosscorrelation, at least one of the sequences has finite duration and is akin to the impulse response of the FIR filter. The second sequence may be a long sequence which contains the desired sequence corrupted by additive noise. Hence the second sequence is akin to the input to the FIR filter.

**TABLE 8.3** Computational Complexity

| Size of FFT $\nu = \log_2 N$ | $c(\nu)$ Number of Complex Multiplications per Output Point |
|---|---|
| 9 | 13.3 |
| 10 | 12.6 |
| 11 | 12.8 |
| 12 | 13.4 |
| 14 | 15.1 |

By time reversing the first sequence and computing its DFT, we have reduced the cross correlation to an equivalent convolution problem (i.e., a linear FIR filtering problem). Therefore, the methodology we developed for linear FIR filtering by use of the FFT applies directly.

## 8.3 A Linear Filtering Approach to Computation of the DFT

The FFT algorithm takes $N$ points of input data and produces an output sequence of $N$ points corresponding to the DFT of the input data. As we have shown, the radix-2 FFT algorithm performs the computation of the DFT in $(N/2)\log_2 N$ multiplications and $N\log_2 N$ additions for an $N$-point sequence.

There are some applications where only a selected number of values of the DFT are desired, but the entire DFT is not required. In such a case, the FFT algorithm may no longer be more efficient than a direct computation of the desired values of the DFT. In fact, when the desired number of values of the DFT is less than $\log_2 N$, a direct computation of the desired values is more efficient.

The direct computation of the DFT can be formulated as a linear filtering operation on the input data sequence. As we will demonstrate, the linear filter takes the form of a parallel bank of resonators where each resonator selects one of the frequencies $\omega_k = 2\pi k/N$, $k = 0, 1, \ldots, N-1$, corresponding to the $N$ frequencies in the DFT.

There are other applications in which we require the evaluation of the $z$-transform of a finite-duration sequence at points other than the unit circle. If the set of desired points in the $z$-plane possesses some regularity, it is possible to also express the computation of the $z$-transform as a linear filtering operation. In this connection, we introduce another algorithm, called the chirp-$z$ transform algorithm, which is suitable for evaluating the $z$-transform of a set of data on a variety of contours in the $z$-plane. This algorithm is also formulated as a linear filtering of a set of input data. As a consequence, the FFT algorithm can be used to compute the chirp-$z$ transform and thus to evaluate the $z$-transform at various contours in the $z$-plane, including the unit circle.

### 8.3.1 The Goertzel Algorithm

The Goertzel algorithm exploits the periodicity of the phase factors $\{W_N^k\}$ and allows us to express the computation of the DFT as a linear filtering operation. Since $W_N^{-kN} = 1$, we can multiply the DFT by this factor. Thus

$$X(k) = W_N^{-kN}\sum_{m=0}^{N-1} x(m)W_N^{km} = \sum_{m=0}^{N-1} x(m)W_N^{-k(N-m)} \tag{8.3.1}$$

We note that (8.3.1) is in the form of a convolution. Indeed, if we define the sequence $y_k(n)$ as

$$y_k(n) = \sum_{m=0}^{N-1} x(m)W_N^{-k(n-m)} \tag{8.3.2}$$

then it is clear that $y_k(n)$ is the convolution of the finite-duration input sequence $x(n)$ of length $N$ with a filter that has an impulse response

$$h_k(n) = W_N^{-kn} u(n) \tag{8.3.3}$$

The output of this filter at $n = N$ yields the value of the DFT at the frequency $\omega_k = 2\pi k/N$. That is,

$$X(k) = y_k(n)|_{n=N} \tag{8.3.4}$$

as can be verified by comparing (8.3.1) with (8.3.2).

The filter with impulse response $h_k(n)$ has the system function

$$H_k(z) = \frac{1}{1 - W_N^{-k} z^{-1}} \tag{8.3.5}$$

This filter has a pole on the unit circle at the frequency $\omega_k = 2\pi k/N$. Thus, the entire DFT can be computed by passing the block of input data into a parallel bank of $N$ single-pole filters (resonators), where each filter has a pole at the corresponding frequency of the DFT.

Instead of performing the computation of the DFT as in (8.3.2), via convolution, we can use the difference equation corresponding to the filter given by (8.3.5) to compute $y_k(n)$ recursively. Thus we have

$$y_k(n) = W_N^{-k} y_k(n-1) + x(n), \qquad y_k(-1) = 0 \tag{8.3.6}$$

The desired output is $X(k) = y_k(N)$, for $k = 0, 1, \ldots, N-1$. To perform this computation, we can compute once and store the phase factors $W_N^{-k}$.

The complex multiplications and additions inherent in (8.3.6) can be avoided by combining the pairs of resonators possessing complex-conjugate poles. This leads to two-pole filters with system functions of the form

$$H_k(z) = \frac{1 - W_N^k z^{-1}}{1 - 2\cos(2\pi k/N) z^{-1} + z^{-2}} \tag{8.3.7}$$

The direct form II realization of the system illustrated in Fig. 8.3.1 is described by the difference equations

$$v_k(n) = 2\cos\frac{2\pi k}{N} v_k(n-1) - v_k(n-2) + x(n) \tag{8.3.8}$$

$$y_k(n) = v_k(n) - W_N^k v_k(n-1) \tag{8.3.9}$$

with initial conditions $v_k(-1) = v_k(-2) = 0$.

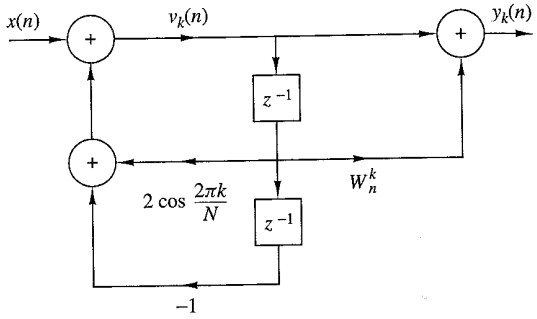



**Figure 8.3.1**
Direct form II realization of two-pole resonator for computing the DFT.

The recursive relation in (8.3.8) is iterated for $n = 0, 1, \ldots, N$, but the equation in (8.3.9) is computed only once at time $n = N$. Each iteration requires one real multiplication and two additions. Consequently, for a real input sequence $x(n)$, this algorithm requires $N + 1$ real multiplications to yield not only $X(k)$ but also, due to symmetry, the value of $X(N - k)$.

The Goertzel algorithm is particularly attractive when the DFT is to be computed at a relatively small number $M$ of values, where $M \leq \log_2 N$. Otherwise, the FFT algorithm is a more efficient method.

### 8.3.2 The Chirp-$z$ Transform Algorithm

The DFT of an $N$-point data sequence $x(n)$ has been viewed as the $z$-transform of $x(n)$ evaluated at $N$ equally spaced points on the unit circle. It has also been viewed as $N$ equally spaced samples of the Fourier transform of the data sequence $x(n)$. In this section we consider the evaluation of $X(z)$ on other contours in the $z$-plane, including the unit circle.

Suppose that we wish to compute the values of the $z$-transform of $x(n)$ at a set of points $\{z_k\}$. Then,

$$X(z_k) = \sum_{n=0}^{N-1} x(n) z_k^{-n}, \qquad k = 0, 1, \ldots, L - 1 \tag{8.3.10}$$

For example, if the contour is a circle of radius $r$ and the $z_k$ are $N$ equally spaced points, then

$$\begin{aligned} z_k &= r e^{j2\pi kn/N}, \qquad k = 0, 1, 2, \ldots, N - 1 \\ X(z_k) &= \sum_{n=0}^{N-1} [x(n) r^{-n}] e^{-j2\pi kn/N}, \qquad k = 0, 1, 2, \ldots, N - 1 \end{aligned} \tag{8.3.11}$$

In this case the FFT algorithm can be applied on the modified sequence $x(n)r^{-n}$.

More generally, suppose that the points $z_k$ in the $z$-plane fall on an arc which begins at some point

$$z_0 = r_0 e^{j\theta_0}$$

and spirals either in toward the origin or out away from the origin such that the points $\{z_k\}$ are defined as

$$z_k = r_0 e^{j\theta_0}(R_0 e^{j\phi_0})^k, \qquad k = 0, 1, \ldots, L-1 \tag{8.3.12}$$

Note that if $R_0 < 1$, the points fall on a contour that spirals toward the origin, and if $R_0 > 1$, the contour spirals away from the origin. If $R_0 = 1$, the contour is a circular arc of radius $r_0$. If $r_0 = 1$ and $R_0 = 1$, the contour is an arc of the unit circle. The latter contour would allow us to compute the frequency content of the sequence $x(n)$ at a dense set of $L$ frequencies in the range covered by the arc without having to compute a large DFT, that is, a DFT of the sequence $x(n)$ padded with many zeros to obtain the desired resolution in frequency. Finally, if $r_0 = R_0 = 1$, $\theta_0 = 0$, $\phi_0 = 2\pi/N$, and $L = N$, the contour is the entire unit circle and the frequencies are those of the DFT. The various contours are illustrated in Fig. 8.3.2.

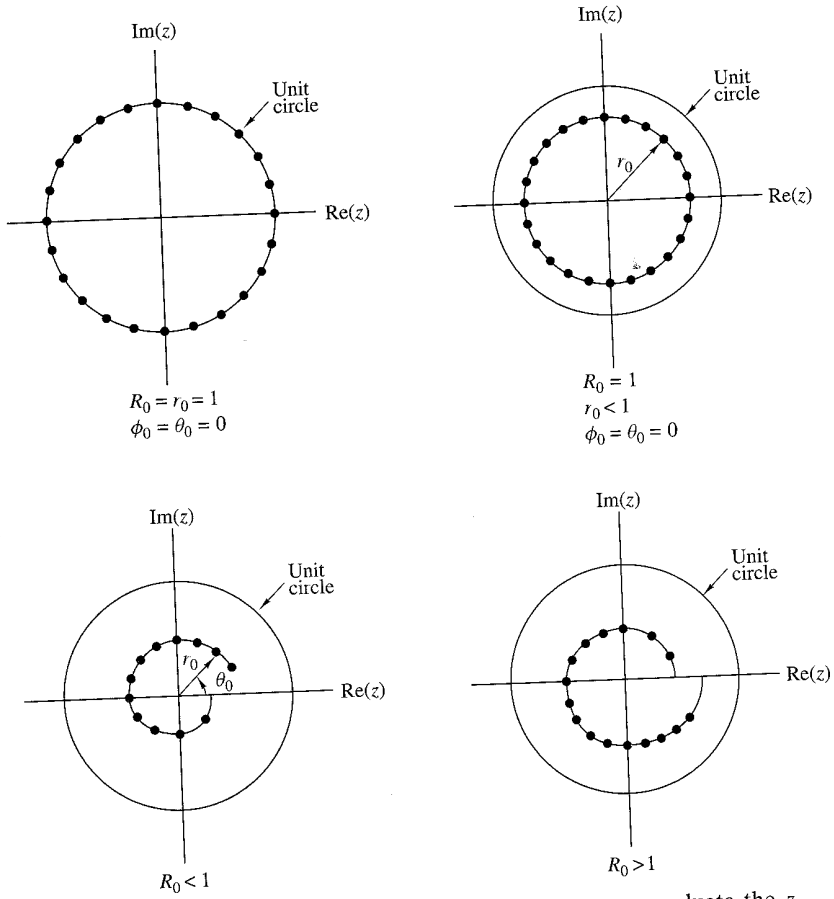



**Figure 8.3.2** Some examples of contours on which we may evaluate the $z$-transform.

When points $\{z_k\}$ in (8.3.12) are substituted into the expression for the $z$-transform, we obtain

$$\begin{aligned} X(z_k) &= \sum_{n=0}^{N-1} x(n) z_k^{-n} \\ &= \sum_{n=0}^{N-1} x(n)(r_0 e^{j\theta_0})^{-n} V^{-nk} \end{aligned} \tag{8.3.13}$$

where, by definition,

$$V = R_0 e^{j\phi_0} \tag{8.3.14}$$

We can express (8.3.13) in the form of a convolution, by noting that

$$nk = \tfrac{1}{2}[n^2 + k^2 - (k-n)^2] \tag{8.3.15}$$

Substitution of (8.3.15) into (8.3.13) yields

$$X(z_k) = V^{-k^2/2} \sum_{n=0}^{N-1} [x(n)(r_0 e^{j\theta_0})^{-n} V^{-n^2/2}] V^{(k-n)^2/2} \tag{8.3.16}$$

Let us define a new sequence $g(n)$ as

$$g(n) = x(n)(r_0 e^{j\theta_0})^{-n} V^{-n^2/2} \tag{8.3.17}$$

Then (8.3.16) can be expressed as

$$X(z_k) = V^{-k^2/2} \sum_{n=0}^{N-1} g(n) V^{(k-n)^2/2} \tag{8.3.18}$$

The summation in (8.3.18) can be interpreted as the convolution of the sequence $g(n)$ with the impulse response $h(n)$ of a filter, where

$$h(n) = V^{n^2/2} \tag{8.3.19}$$

Consequently, (8.3.18) may be expressed as

$$\begin{aligned} X(z_k) &= V^{-k^2/2} y(k) \\ &= \frac{y(k)}{h(k)}, \qquad k = 0, 1, \ldots, L-1 \end{aligned} \tag{8.3.20}$$

where $y(k)$ is the output of the filter

$$y(k) = \sum_{n=0}^{N-1} g(n) h(k-n), \qquad k = 0, 1, \ldots, L-1 \tag{8.3.21}$$

We observe that both $h(n)$ and $g(n)$ are complex-valued sequences.

The sequence $h(n)$ with $R_0 = 1$ has the form of a complex exponential with argument $\omega n = n^2\phi_0/2 = (n\phi_0/2)n$. The quantity $n\phi_0/2$ represents the frequency of the complex exponential signal, which increases linearly with time. Such signals are used in radar systems and are called *chirp signals*. Hence the $z$-transform evaluated as in (8.3.18) is called the *chirp-z transform*.

The linear convolution in (8.3.21) is most efficiently done by use of the FFT algorithm. The sequence $g(n)$ is of length $N$. However, $h(n)$ has infinite duration. Fortunately, only a portion $h(n)$ is required to compute the $L$ values of $X(z)$.

Since we will compute the convolution in (8.3.21) via the FFT, let us consider the circular convolution of the $N$-point sequence $g(n)$ with an $M$-point section of $h(n)$, where $M > N$. In such a case, we know that the first $N-1$ points contain aliasing and that the remaining $M-N+1$ points are identical to the result that would be obtained from a linear convolution of $h(n)$ with $g(n)$. In view of this, we should select a DFT of size

$$M = L + N - 1$$

which would yield $L$ valid points and $N-1$ points corrupted by aliasing.

The section of $h(n)$ that is needed for this computation corresponds to the values of $h(n)$ for $-(N-1) \le n \le (L-1)$, which is of length $M = L+N-1$, as observed from (8.3.21). Let us define the sequence $h_1(n)$ of length $M$ as

$$h_1(n) = h(n-N+1), \qquad n = 0, 1, \ldots, M-1 \tag{8.3.22}$$

and compute its $M$-point DFT via the FFT algorithm to obtain $H_1(k)$. From $x(n)$ we compute $g(n)$ as specified by (8.3.17), pad $g(n)$ with $L-1$ zeros, and compute its $M$-point DFT to yield $G(k)$. The IDFT of the product $Y_1(k) = G(k)H_1(k)$ yields the $M$-point sequence $y_1(n)$, $n = 0, 1, \ldots, M-1$. The first $N-1$ points of $y_1(n)$ are corrupted by aliasing and are discarded. The desired values are $y_1(n)$ for $N-1 \le n \le M-1$, which correspond to the range $0 \le n \le L-1$ in (8.3.21), that is,

$$y(n) = y_1(n+N-1), \qquad n = 0, 1, \ldots, L-1 \tag{8.3.23}$$

Alternatively, we can define a sequence $h_2(n)$ as

$$h_2(n) = \begin{cases} h(n), & 0 \le n \le L-1 \\ h(n-N-L+1), & L \le n \le M-1 \end{cases} \tag{8.3.24}$$

The $M$-point DFT of $h_2(n)$ yields $H_2(k)$, which when multiplied by $G(k)$ yields $Y_2(k) = G(k)H_2(k)$. The IDFT of $Y_2(k)$ yields the sequence $y_2(n)$ for $0 \le n \le M-1$. Now the desired values of $y_2(n)$ are in the range $0 \le n \le L-1$, that is,

$$y(n) = y_2(n), \qquad n = 0, 1, \ldots, L-1 \tag{8.3.25}$$

Finally, the complex values $X(z_k)$ are computed by dividing $y(k)$ by $h(k)$, $k = 0, 1, \ldots, L-1$, as specified by (8.3.20).

In general, the computational complexity of the chirp-$z$ transform algorithm described above is of the order of $M \log_2 M$ complex multiplications, where $M = N + L - 1$. This number should be compared with the product, $N \cdot L$, the number of computations required by direct evaluation of the $z$-transform. Clearly, if $L$ is small, direct computation is more efficient. However, if $L$ is large, then the chirp-$z$ transform algorithm is more efficient.

The chirp-$z$ transform method has been implemented in hardware to compute the DFT of signals. For the computation of the DFT, we select $r_0 = R_0 = 1$, $\theta_0 = 0$, $\phi_0 = 2\pi/N$, and $L = N$. In this case

$$\begin{aligned} V^{-n^2/2} &= e^{-j\pi n^2/N} \\ &= \cos\frac{\pi n^2}{N} - j\sin\frac{\pi n^2}{N} \end{aligned} \tag{8.3.26}$$

The chirp filter with impulse response

$$\begin{aligned} h(n) &= V^{n^2/2} \\ &= \cos\frac{\pi n^2}{N} + j\sin\frac{\pi n^2}{N} \\ &= h_r(n) + jh_i(n) \end{aligned} \tag{8.3.27}$$

has been implemented as a pair of FIR filters with coefficients $h_r(n)$ and $h_i(n)$, respectively. Both *surface acoustic wave* (SAW) devices and *charge coupled devices*

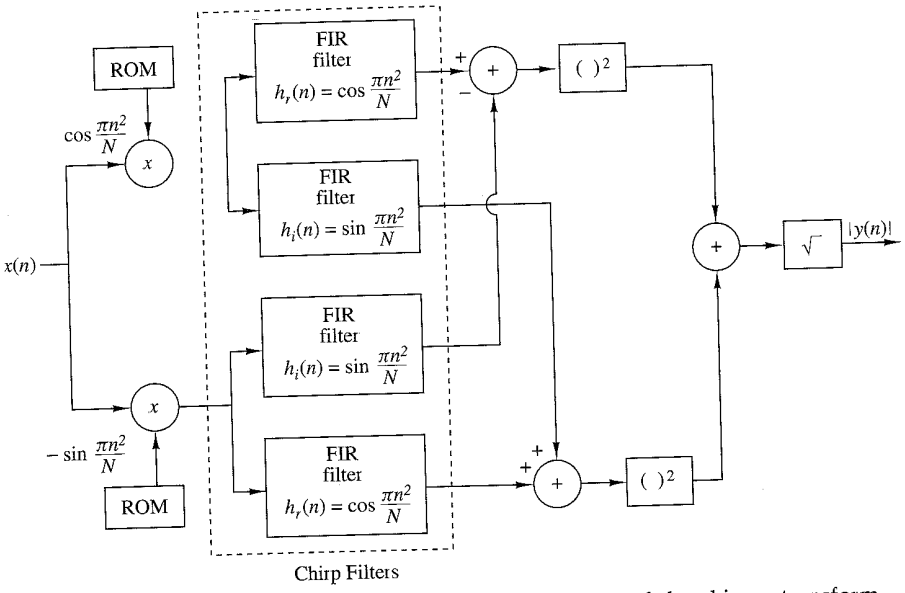



**Figure 8.3.3** Block diagram illustrating the implementation of the chirp-$z$ transform for computing the DFT (magnitude only).

(CCD) have been used in practice for the FIR filters. The cosine and sine sequences given in (8.3.26) needed for the premultiplications and postmultiplications are usually stored in a read-only memory (ROM). Furthermore, we note that if only the magnitude of the DFT is desired, the postmultiplications are unnecessary. In this case,

$$|X(z_k)| = |y(k)|, \qquad k = 0, 1, \ldots, N-1 \tag{8.3.28}$$

as illustrated in Fig. 8.3.3. Thus the linear FIR filtering approach using the chirp-$z$ transform has been implemented for the computation of the DFT.

## 8.4 Quantization Effects in the Computation of the DFT[1]

As we have observed in our previous discussions, the DFT plays an important role in many digital signal processing applications, including FIR filtering, the computation of the correlation between signals, and spectral analysis. For this reason it is important for us to know the effect of quantization errors in its computation. In particular, we shall consider the effect of round-off errors due to the multiplications performed in the DFT with fixed-point arithmetic.

The model that we shall adopt for characterizing round-off errors in multiplication is the additive white noise model that we use in the statistical analysis of round-off errors in IIR and FIR filters (see Fig. 9.6.8). Although the statistical analysis is performed for rounding, the analysis can be easily modified to apply to truncation in two's-complement arithmetic (see Sec. 9.4.3).

Of particular interest is the analysis of round-off errors in the computation of the DFT via the FFT algorithm. However, we shall first establish a benchmark by determining the round-off errors in the direct computation of the DFT.

### 8.4.1 Quantization Errors in the Direct Computation of the DFT

Given a finite-duration sequence $\{x(n)\}$, $0 \le n \le N-1$, the DFT of $\{x(n)\}$ is defined as

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}, \qquad k = 0, 1, \ldots, N-1 \tag{8.4.1}$$

where $W_N = e^{-j2\pi/N}$. We assume that in general, $\{x(n)\}$ is a complex-valued sequence. We also assume that the real and imaginary components of $\{x(n)\}$ and $\{W_N^{kn}\}$ are represented by $b$ bits. Consequently, the computation of the product $x(n)W_N^{kn}$ requires four real multiplications. Each real multiplication is rounded from $2b$ bits to $b$ bits, and hence there are four quantization errors for each complex-valued multiplication.

In the direct computation of the DFT, there are $N$ complex-valued multiplications for each point in the DFT. Therefore, the total number of real multiplications in the computation of a single point in the DFT is $4N$. Consequently, there are $4N$ quantization errors.

[1] It is recommended that the reader review Section 9.5 prior to reading this section.

Let us evaluate the variance of the quantization errors in a fixed-point computation of the DFT. First, we make the following assumptions about the statistical properties of the quantization errors.

1. The quantization errors due to rounding are uniformly distributed random variables in the range $(-\Delta/2, \Delta/2)$ where $\Delta = 2^{-b}$.
2. The $4N$ quantization errors are mutually uncorrelated.
3. The $4N$ quantization errors are uncorrelated with the sequence $\{x(n)\}$.

Since each of the quantization errors has a variance

$$\sigma_e^2 = \frac{\Delta^2}{12} = \frac{2^{-2b}}{12} \tag{8.4.2}$$

the variance of the quantization errors from the $4N$ multiplications is

$$\begin{aligned} \sigma_q^2 &= 4N\sigma_e^2 \\ &= \frac{N}{3} \cdot 2^{-2b} \end{aligned} \tag{8.4.3}$$

Hence the variance of the quantization error is proportional to the size of DFT. Note that when $N$ is a power of 2 (i.e., $N = 2^\nu$), the variance can be expressed as

$$\sigma_q^2 = \frac{2^{-2(b-\nu/2)}}{3} \tag{8.4.4}$$

This expression implies that every fourfold increase in the size $N$ of the DFT requires an additional bit in computational precision to offset the additional quantization errors.

To prevent overflow, the input sequence to the DFT requires scaling. Clearly, an upper bound on $|X(k)|$ is

$$|X(k)| \le \sum_{n=0}^{N-1} |x(n)| \tag{8.4.5}$$

If the dynamic range in addition is $(-1, 1)$, then $|X(k)| < 1$ requires that

$$\sum_{n=0}^{N-1} |x(n)| < 1 \tag{8.4.6}$$

If $|x(n)|$ is initially scaled such that $|x(n)| < 1$ for all $n$, then each point in the sequence can be divided by $N$ to ensure that (8.4.6) is satisfied.

The scaling implied by (8.4.6) is extremely severe. For example, suppose that the signal sequence $\{x(n)\}$ is white and, after scaling, each value $|x(n)|$ of the sequence is uniformly distributed in the range $(-1/N, 1/N)$. Then the variance of the signal sequence is

$$\sigma_x^2 = \frac{(2/N)^2}{12} = \frac{1}{3N^2} \tag{8.4.7}$$

and the variance of the output DFT coefficients $|X(k)|$ is

$$\begin{aligned}\sigma_X^2 &= N\sigma_x^2 \\ &= \frac{1}{3N}\end{aligned} \tag{8.4.8}$$

Thus the signal-to-noise power ratio is

$$\frac{\sigma_X^2}{\sigma_q^2} = \frac{2^{2b}}{N^2} \tag{8.4.9}$$

We observe that the scaling is responsible for reducing the SNR by $N$ and the combination of scaling and quantization errors results in a total reduction that is proportional to $N^2$. Hence scaling the input sequence $\{x(n)\}$ to satisfy (8.4.6) imposes a severe penalty on the signal-to-noise ratio in the DFT.

**EXAMPLE 8.4.1**

Use (8.4.9) to determine the number of bits required to compute the DFT of a 1024-point sequence with an SNR of 30 dB.

**Solution.** The size of the sequence is $N = 2^{10}$. Hence the SNR is

$$10\log_{10}\frac{\sigma_X^2}{\sigma_q^2} = 10\log_{10}2^{2b-20}$$

For an SNR of 30 dB, we have

$$3(2b-20) = 30$$

$$b = 15 \text{ bits}$$

Note that the 15 bits is the precision for both multiplication and addition.

Instead of scaling the input sequence $\{x(n)\}$, suppose we simply require that $|x(n)| < 1$. Then we must provide a sufficiently large dynamic range for addition such that $|X(k)| < N$. In such a case, the variance of the sequence $\{|x(n)|\}$ is $\sigma_x^2 = \frac{1}{3}$, and hence the variance of $|X(k)|$ is

$$\sigma_X^2 = N\sigma_x^2 = \frac{N}{3} \tag{8.4.10}$$

Consequently, the SNR is

$$\frac{\sigma_X^2}{\sigma_q^2} = 2^{2b} \tag{8.4.11}$$

If we repeat the computation in Example 8.4.1, we find that the number of bits required to achieve an SNR of 30 dB is $b = 5$ bits. However, we need an additional 10 bits for the accumulator (the adder) to accommodate the increase in the dynamic range for addition. Although we did not achieve any reduction in the dynamic range for addition, we have managed to reduce the precision in multiplication from 15 bits to 5 bits, which is highly significant.

### 8.4.2 Quantization Errors in FFT Algorithms

As we have shown, the FFT algorithms require significantly fewer multiplications than the direct computation of the DFT. In view of this we might conclude that the computation of the DFT via an FFT algorithm will result in smaller quantization errors. Unfortunately, that is not the case, as we will demonstrate.

Let us consider the use of fixed-point arithmetic in the computation of a radix-2 FFT algorithm. To be specific, we select the radix-2, decimation-in-time algorithm illustrated in Fig. 8.4.1 for the case $N = 8$. The results on quantization errors that we obtain for this radix-2 FFT algorithm are typical of the results obtained with other radix-2 and higher radix algorithms.

We observe that each butterfly computation involves one complex-valued multiplication or, equivalently, four real multiplications. We ignore the fact that some butterflies contain a trivial multiplication by $\pm 1$. If we consider the butterflies that affect the computation of any one value of the DFT, we find that, in general, there are $N/2$ in the first stage of the FFT, $N/4$ in the second stage, $N/8$ in the third state, and so on, until the last stage, where there is only one. Consequently, the number of

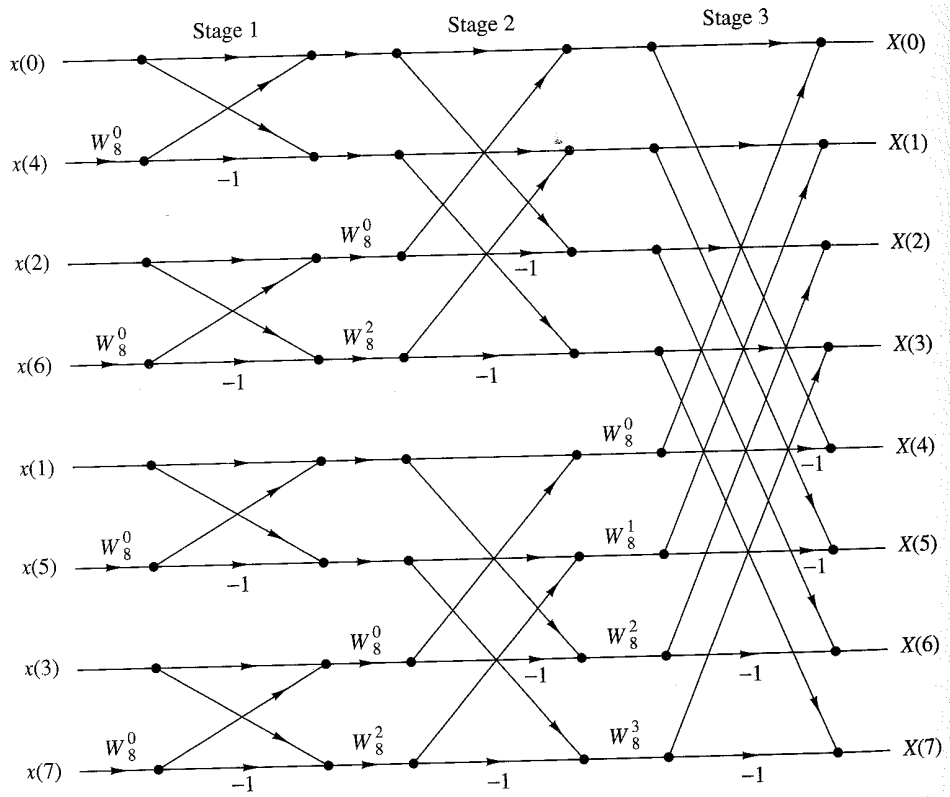



**Figure 8.4.1** Decimation-in-time FFT algorithm.

butterflies per output point is

$$2^{\nu-1}+2^{\nu-2}+\cdots+2+1=2^{\nu-1}\left[1+\left(\tfrac{1}{2}\right)+\cdots+\left(\tfrac{1}{2}\right)^{\nu-1}\right]$$
$$=2^{\nu}[1-\left(\tfrac{1}{2}\right)^{\nu}]=N-1 \tag{8.4.12}$$

For example, the butterflies that affect the computation of $X(3)$ in the eight-point FFT algorithm of Fig. 8.4.1 are illustrated in Fig. 8.4.2.

The quantization errors introduced in each butterfly propagate to the output. Note that the quantization errors introduced in the first stage propagate through $(\nu-1)$ stages, those introduced in the second stage propagate through $(\nu-2)$ stages, and so on. As these quantization errors propagate through a number of subsequent stages, they are phase shifted (phase rotated) by the phase factors $W_N^{kn}$. These phase rotations do not change the statistical properties of the quantization errors and, in particular, the variance of each quantization error remains invariant.

If we assume that the quantization errors in each butterfly are uncorrelated with the errors in other butterflies, then there are $4(N-1)$ errors that affect the output of each point of the FFT. Consequently, the variance of the total quantization error at the output is

$$\sigma_q^2=4(N-1)\frac{\Delta^2}{12}\approx\frac{N\Delta^2}{3} \tag{8.4.13}$$

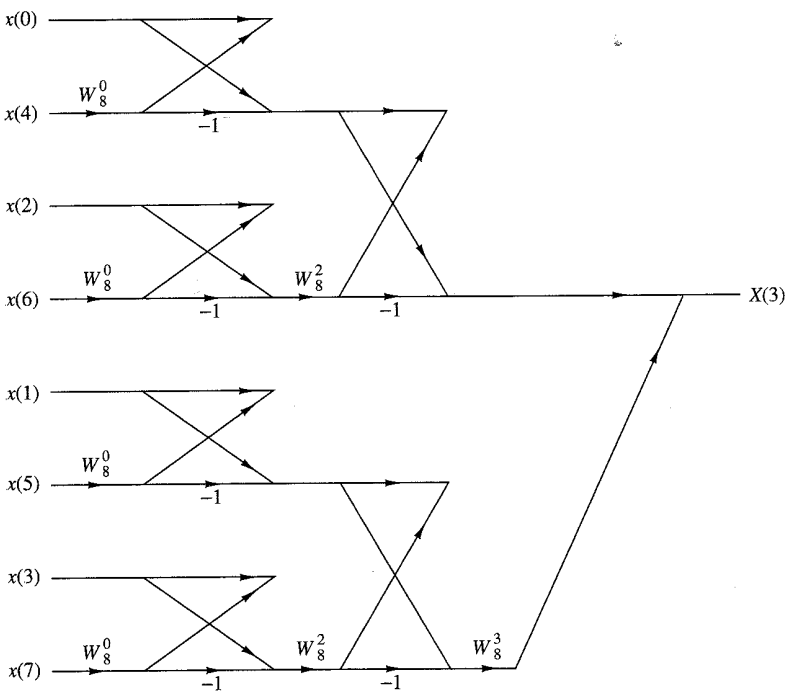



**Figure 8.4.2** Butterflies that affect the computation of $X(3)$.

where $\Delta = 2^{-b}$. Hence

$$\sigma_q^2 = \frac{N}{3} \cdot 2^{-2b} \tag{8.4.14}$$

This is exactly the same result that we obtained for the direct computation of the DFT.

The result in (8.4.14) should not be surprising. In fact, the FFT algorithm does not reduce the number of multiplications required to compute a single point of the DFT. It does, however, exploit the periodicities in $W_N^{kn}$ and thus reduces the number of multiplications in the computation of the entire block of $N$ points in the DFT.

As in the case of the direct computation of the DFT, we must scale the input sequence to prevent overflow. Recall that if $|x(n)| < 1/N$, $0 \le n \le N-1$, then $|X(k)| < 1$ for $0 \le k \le N-1$. Thus overflow is avoided. With this scaling, the relations in (8.4.7), (8.4.8), and (8.4.9), obtained previously for the direct computation of the DFT, apply to the FFT algorithm as well. Consequently, the same SNR is obtained for the FFT.

Since the FFT algorithm consists of a sequence of stages, where each stage contains butterflies that involve pairs of points, it is possible to devise a different scaling strategy that is not as severe as dividing each input point by $N$. This alternative scaling strategy is motivated by the observation that the intermediate values $|X_n(k)|$ in the $n = 1, 2, \ldots, \nu$ stages of the FFT algorithm satisfy the conditions (see Problem 8.35)

$$\begin{aligned} \max[|X_{n+1}(k)|, |X_{n+1}(l)|] &\ge \max[|X_n(k)|, |X_n(l)|] \\ \max[|X_{n+1}(k)|, |X_{n+1}(l)|] &\le 2\max[|X_n(k)|, |X_n(l)|] \end{aligned} \tag{8.4.15}$$

In view of these relations, we can distribute the total scaling of $1/N$ into each of the stages of the FFT algorithm. In particular, if $|x(n)| < 1$, we apply a scale factor of $\frac{1}{2}$ in the first stage so that $|x(n)| < \frac{1}{2}$. Then the output of each subsequent stage in the FFT algorithm is scaled by $\frac{1}{2}$, so that after $\nu$ stages we have achieved an overall scale factor of $(\frac{1}{2})^\nu = 1/N$. Thus overflow in the computation of the DFT is avoided.

This scaling procedure does not affect the signal level at the output of the FFT algorithm, but it significantly reduces the variance of the quantization errors at the output. Specifically, each factor of $\frac{1}{2}$ reduces the variance of a quantization error term by a factor of $\frac{1}{4}$. Thus the $4(N/2)$ quantization errors introduced in the first stage are reduced in variance by $(\frac{1}{4})^{\nu-1}$, the $4(N/4)$ quantization errors introduced in the second stage are reduced in variance by $(\frac{1}{4})^{\nu-2}$, and so on. Consequently, the total variance of the quantization errors at the output of the FFT algorithm is

$$\begin{aligned} \sigma_q^2 &= \frac{\Delta^2}{12}\left\{4\left(\frac{N}{2}\right)\left(\frac{1}{4}\right)^{\nu-1} + 4\left(\frac{N}{4}\right)\left(\frac{1}{4}\right)^{\nu-2} + 4\left(\frac{N}{8}\right)\left(\frac{1}{4}\right)^{\nu-3} + \cdots + 4\right\} \\ &= \frac{\Delta^2}{3}\left\{\left(\frac{1}{2}\right)^{\nu-1} + \left(\frac{1}{2}\right)^{\nu-2} + \cdots + \frac{1}{2} + 1\right\} \\ &= \frac{2\Delta^2}{3}\left[1 - \left(\frac{1}{2}\right)^{\nu}\right] \approx \frac{2}{3} \cdot 2^{-2b} \end{aligned} \tag{8.4.16}$$

where the factor $(\frac{1}{2})^\nu$ is negligible.

We now observe that (8.4.16) is no longer proportional to $N$. On the other hand, the signal has the variance $\sigma_X^2 = 1/3N$, as given in (8.4.8). Hence the SNR is

$$\begin{aligned}\frac{\sigma_X^2}{\sigma_q^2} &= \frac{1}{2N} \cdot 2^{2b} \\ &= 2^{2b-v-1}\end{aligned} \tag{8.4.17}$$

Thus, by distributing the scaling of $1/N$ uniformly throughout the FFT algorithm, we have achieved an SNR that is inversely proportional to $N$ instead of $N^2$.

**EXAMPLE 8.4.2**

Determine the number of bits required to compute an FFT of 1024 points with an SNR of 30 dB when the scaling is distributed as described above.

**Solution.** The size of the FFT is $N = 2^{10}$. Hence the SNR according to (8.4.17) is

$$\begin{aligned}10 \log_{10} 2^{2b-v-1} &= 30 \\ 3(2b - 11) &= 30 \\ b &= \frac{21}{2} (11 \text{ bits})\end{aligned}$$

This can be compared with the 15 bits required if all the scaling is performed in the first stage of the FFT algorithm.

## 8.5 Summary and References

The focus of this chapter was on the efficient computation of the DFT. We demonstrated that by taking advantage of the symmetry and periodicity properties of the exponential factors $W_N^{kn}$, we can reduce the number of complex multiplications needed to compute the DFT from $N^2$ to $N \log_2 N$ when $N$ is a power of 2. As we indicated, any sequence can be augmented with zeros, such that $N = 2^v$.

For decades, FFT-type algorithms were of interest to mathematicians who were concerned with computing values of Fourier series by hand. However, it was not until Cooley and Tukey (1965) published their well-known paper that the impact and significance of the efficient computation of the DFT was recognized. Since then the Cooley–Tukey FFT algorithm and its various forms, for example, the algorithms of Singleton (1967, 1969), have had a tremendous influence on the use of the DFT in convolution, correlation, and spectrum analysis. For a historical perspective on the FFT algorithm, the reader is referred to the paper by Cooley et al. (1967).

The split-radix FFT (SRFFT) algorithm described in Section 8.1.5 is due to Duhamel and Hollmann (1984, 1986). The "mirror" FFT (MFFT) and "phase" FFT (PFFT) algorithms were described to the authors by R. Price. The exploitation of symmetry properties in the data to reduce the computation time is described in a paper by Swarztrauber (1986).

Over the years, a number of tutorial papers have been published on FFT algorithms. We cite the early papers by Brigham and Morrow (1967), Cochran et al. (1967), Bergland (1969), and Cooley et al. (1967, 1969).

The recognition that the DFT can be arranged and computed as a linear convolution is also highly significant. Goertzel (1968) indicated that the DFT can be computed via linear filtering, although the computational savings of this approach is rather modest, as we have observed. More significant is the work of Bluestein (1970), who demonstrated that the computation of the DFT can be formulated as a chirp linear filtering operation. This work led to the development of the chirp-$z$ transform algorithm by Rabiner et al. (1969).

In addition to the FFT algorithms described in this chapter, there are other efficient algorithms for computing the DFT, some of which further reduce the number of multiplications, but usually require more additions. Of particular importance is an algorithm due to Rader and Brenner (1976), the class of prime factor algorithms, such as the Good algorithm (1971), and the Winograd algorithm (1976, 1978). For a description of these and related algorithms, the reader may refer to the text by Blahut (1985).

## Problems

**8.1** Show that each of the numbers

$$e^{j(2\pi/N)k}, \qquad 0 \le k \le N-1$$

corresponds to an $N$th root of unity. Plot these numbers as phasors in the complex plane and illustrate, by means of this figure, the orthogonality property

$$\sum_{n=0}^{N-1} e^{j(2\pi/N)kn} e^{-j(2\pi/N)ln} = \begin{cases} N, & \text{if } k = l \\ 0, & \text{if } k \neq l \end{cases}$$

**8.2** **(a)** Show that the phase factors can be computed recursively by

$$W_N^{ql} = W_N^q W_N^{q(l-1)}$$

**(b)** Perform this computation once using single-precision floating-point arithmetic and once using only four significant digits. Note the deterioration due to the accumulation of round-off errors in the latter case.

**(c)** Show how the results in part (b) can be improved by resetting the result to the correct value $-j$, each time $ql = N/4$.

**8.3** Let $x(n)$ be a real-valued $N$-point ($N = 2^\nu$) sequence. Develop a method to compute an $N$-point DFT $X'(k)$, which contains only the odd harmonics [i.e., $X'(k) = 0$ if $k$ is even] by using only a real $N/2$-point DFT.

**8.4** A designer has available a number of eight-point FFT chips. Show explicitly how he should interconnect three such chips in order to compute a 24-point DFT.

**8.5** The $z$-transform of the sequence $x(n) = u(n) - u(n-7)$ is sampled at five points on the unit circle as follows:

$$x(k) = X(z)|_z = e^{j2\pi k/5}, \qquad k = 0, 1, 2, 3, 4$$

Determine the inverse DFT $x'(n)$ of $X(k)$. Compare it with $x(n)$ and explain the results.

**8.6** Consider a finite-duration sequence $x(n)$, $0 \le n \le 7$, with $z$-transform $X(z)$. We wish to compute $X(z)$ at the following set of values:

$$z_k = 0.8e^{j[(2\pi k/8)+(\pi/8)]}, \qquad 0 \le k \le 7$$

**(a)** Sketch the points $\{z_k\}$ in the complex plane.

**(b)** Determine a sequence $s(n)$ such that its DFT provides the desired samples of $X(z)$.

**8.7** Derive the radix-2 decimation-in-time FFT algorithm given by (8.1.26) and (8.1.27) as a special case of the more general algorithmic procedure given by (8.1.16) through (8.1.18).

**8.8** Compute the eight-point DFT of the sequence

$$x(n) = \begin{cases} 1, & 0 \le n \le 7 \\ 0, & \text{otherwise} \end{cases}$$

by using the decimation-in-frequency FFT algorithm described in the text.

**8.9** Derive the signal flow graph for the $N = 16$-point, radix-4 decimation-in-time FFT algorithm in which the input sequence is in normal order and the computations are done in place.

**8.10** Derive the signal flow graph for the $N = 16$-point, radix-4 decimation-in-frequency FFT algorithm in which the input sequence is in digit-reversed order and the output DFT is in normal order.

**8.11** Compute the eight-point DFT of the sequence

$$x(n) = \left\{\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0, 0, 0, 0\right\}$$

using the in-place radix-2 decimation-in-time and radix-2 decimation-in-frequency algorithms. Follow exactly the corresponding signal flow graphs and keep track of all the intermediate quantities by putting them on the diagrams.

**8.12** Compute the 16-point DFT of the sequence

$$x(n) = \cos\frac{\pi}{2}n, \qquad 0 \le n \le 15$$

using the radix-4 decimation-in-time algorithm.

**8.13** Consider the eight-point decimation-in-time (DIT) flow graph in Fig. 8.1.6.

**(a)** What is the gain of the "signal path" that goes from $x(7)$ to $X(2)$?

**(b)** How many paths lead from the input to a given output sample? Is this true for every output sample?

**(c)** Compute $X(3)$ using the operations dictated by this flow graph.

**8.14** Draw the flow graph for the decimation-in-frequency (DIF) SRFFT algorithm for $N = 16$. What is the number of nontrivial multiplications?

**8.15** Derive the algorithm and draw the $N = 8$ flow graph for the DIT SRFFT algorithm. Compare your flow graph with the DIF radix-2 FFT flow graph shown in Fig. 8.1.11.

**8.16** Show that the product of two complex numbers $(a+jb)$ and $(c+jd)$ can be performed with three real multiplications and five additions using the algorithm

$$x_R = (a-b)d + (c-d)a$$

$$x_I = (a-b)d + (c+d)b$$

where

$$x = x_R + jx_I = (a+jb)(c+jd)$$

**8.17** Explain how the DFT can be used to compute $N$ equispaced samples of the $z$-transform of an $N$-point sequence, on a circle of radius $r$.

**8.18** A real-valued $N$-point sequence $x(n)$ is called DFT bandlimited if its DFT $X(k) = 0$ for $k_0 \le k \le N - k_0$. We insert $(L-1)N$ zeros in the middle of $X(k)$ to obtain the following $LN$-point DFT:

$$X'(k) = \begin{cases} X(k), & 0 \le k \le k_0 - 1 \\ 0, & k_0 \le k \le LN - k_0 \\ X(k+N-LN), & LN - k_0 + 1 \le k \le LN - 1 \end{cases}$$

Show that

$$Lx'(Ln) = x(n), \qquad 0 \le n \le N-1$$

where

$$x'(n) \underset{LN}{\overset{DFT}{\longleftrightarrow}} X'(k)$$

Explain the meaning of this type of processing by working out an example with $N = 4$, $L = 1$, and $X(k) = \{1, 0, 0, 1\}$.

**8.19** Let $X(k)$ be the $N$-point DFT of the sequence $x(n)$, $0 \le n \le N-1$. What is the $N$-point DFT of the sequence $s(n) = X(n)$, $0 \le n \le N-1$?

**8.20** Let $X(k)$ be the $N$-point DFT of the sequence $x(n)$, $0 \le n \le N-1$. We define a $2N$-point sequence $y(n)$ as

$$y(n) = \begin{cases} x\left(\frac{n}{2}\right), & n \text{ even} \\ 0, & n \text{ odd} \end{cases}$$

Express the $2N$-point DFT of $y(n)$ in terms of $X(k)$.

**8.21** **(a)** Determine the $z$-transform $W(z)$ of the Hanning window $w(n) = \left(1 - \cos\frac{2\pi n}{N-1}\right)/2$.

**(b)** Determine a formula to compute the $N$-point DFT $X_w(k)$ of the signal $x_w(n) = w(n)x(n)$, $0 \le n \le N-1$, from the $N$-point DFT $X(k)$ of the signal $x(n)$.

**8.22** Create a DFT coefficient table that uses only $N/4$ memory locations to store the first quadrant of the sine sequence (assume $N$ even).

**8.23** Determine the computational burden of the algorithm given by (8.2.12) and compare it with the computational burden required in the $2N$-point DFT of $g(n)$. Assume that the FFT algorithm is a radix-2 algorithm.

**8.24** Consider an IIR system described by the difference equation

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

Describe a procedure that computes the frequency response $H\left(\frac{2\pi}{N}k\right)$, $k = 0, 1, \ldots, N-1$ using the FFT algorithm ($N = 2^\nu$).

**8.25** Develop a radix-3 decimation-in-time FFT algorithm for $N = 3^\nu$ and draw the corresponding flow graph for $N = 9$. What is the number of required complex multiplications? Can the operations be performed in place?

**8.26** Repeat Problem 8.25 for the DIF case.

**8.27** *FFT input and output pruning* In many applications we wish to compute only a few points $M$ of the $N$-point DFT of a finite-duration sequence of length $L$ (i.e., $M << N$ and $L << N$).

**(a)** Draw the flow graph of the radix-2 DIF FFT algorithm for $N = 16$ and eliminate [i.e., prune] all signal paths that originate from zero inputs assuming that only $x(0)$ and $x(1)$ are nonzero.

**(b)** Repeat part (a) for the radix-2 DIT algorithm.

**(c)** Which algorithm is better if we wish to compute all points of the DFT? What happens if we want to compute only the points $X(0)$, $X(1)$, $X(2)$, and $X(3)$? Establish a rule to choose between DIT and DIF pruning depending on the values of $M$ and $L$.

**(d)** Give an estimate of saving in computations in terms of $M$, $L$, and $N$.

**8.28** *Parallel computation of the DFT* Suppose that we wish to compute an $N = 2^p 2^\nu$-point DFT using $2^p$ digital signal processors (DSPs). For simplicity we assume that $p = \nu = 2$. In this case each DSP carries out all the computations that are necessary to compute $2^\nu$ DFT points.

**(a)** Using the radix-2 DIF flow graph, show that to avoid data shuffling, the entire sequence $x(n)$ should be loaded to the memory of each DSP.

**(b)** Identify and redraw the portion of the flow graph that is executed by the DSP that computes the DFT samples $X(2)$, $X(10)$, $X(6)$, and $X(14)$.

**(c)** Show that, if we use $M = 2^p$ DSPs, the computation speed-up $S$ is given by

$$S = M\frac{\log_2 N}{\log_2 N - \log_2 M + 2(M-1)}$$

**8.29** Develop an inverse radix-2 DIT FFT algorithm starting with the definition. Draw the flow graph for computation and compare with the corresponding flow graph for the direct FFT. Can the IFFT flow graph be obtained from the one for the direct FFT?

**8.30** Repeat Problem 8.29 for the DIF case.

**8.31** Show that an FFT on data with Hermitian symmetry can be derived by reversing the flow graph of an FFT for real data.

**8.32** Determine the system function $H(z)$ and the difference equation for the system that uses the Goertzel algorithm to compute the DFT value $X(N-k)$.

**8.33** **(a)** Suppose that $x(n)$ is a finite-duration sequence of $N = 1024$ points. It is desired to evaluate the $z$-transform $X(z)$ of the sequence at the points

$$z_k = e^{j(2\pi/1024)k}, \qquad k = 0, 100, 200, \ldots, 1000$$

by using the most efficient method or algorithm possible. Describe an algorithm for performing this computation efficiently. Explain how you arrived at your answer by giving the various options or algorithms that can be used.

**(b)** Repeat part (a) if $X(z)$ is to be evaluated at

$$z_k = 2(0.9)^k e^{j[(2\pi/5000)k+\pi/2]}, \qquad k = 0, 1, 2, \ldots, 999$$

**8.34** Repeat the analysis for the variance of the quantization error, carried out in Section 8.4.2, for the decimation-in-frequency radix-2 FFT algorithm.

**8.35** The basic butterfly in the radix-2 decimation-in-time FFT algorithm is

$$X_{n+1}(k) = X_n(k) + W_N^m X_n(l)$$
$$X_{n+1}(l) = X_n(k) - W_N^m X_n(l)$$

**(a)** If we require that $|X_n(k)| < \frac{1}{2}$ and $|X_n(l)| < \frac{1}{2}$, show that

$$|\mathrm{Re}[X_{[}X_{n+1}(k)]| < 1, \qquad |\mathrm{Re}[X_{n+1}(l)]| < 1$$
$$|\mathrm{Im}[X_{[}X_{n+1}(k)]| < 1, \qquad |\mathrm{Im}[X_{n+1}(l)]| < 1$$

Thus overflow does not occur.

**(b)** Prove that

$$\max[|X_{n+1}(k)|, |X_{n+1}(l)|] \geq \max[|X_n(k)|, |X_n(l)|]$$
$$\max[|X_{n+1}(k)|, |X_{n+1}(l)|] \leq 2\max[|X_n(k)|, |X_n(l)|]$$

**8.36** *Computation of the DFT* Use an FFT subroutine to compute the following DFTs and plot the magnitudes $|X(k)|$ of the DFTs.

**(a)** The 64-point DFT of the sequence

$$x(n) = \begin{cases} 1, & n = 0, 1, \ldots, 15 \quad (N_1 = 16) \\ 0, & \text{otherwise} \end{cases}$$

**(b)** The 64-point DFT of the sequence

$$x(n) = \begin{cases} 1, & n = 0, 1, \ldots, 7 \quad (N_1 = 8) \\ 0, & \text{otherwise} \end{cases}$$

**(c)** The 128-point DFT of the sequence in part (a).

**(d)** The 64-point DFT of the sequence

$$x(n) = \begin{cases} 10e^{j(\pi/8)n}, & n = 0, 1, \ldots, 63 \quad (N_1 = 64) \\ 0, & \text{otherwise} \end{cases}$$

Answer the following questions.

1. What is the frequency interval between successive samples for the plots in parts (a), (b), (c), and (d)?
2. What is the value of the spectrum at zero frequency (dc value) obtained from the plots in parts (a), (b), (c), (d)?
   From the formula

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j(2\pi/N)nk}$$

   compute the theoretical values for the dc value and check these with the computer results.
3. In plots (a), (b), and (c), what is the *frequency interval* between successive nulls in the spectrum? What is the relationship between $N_1$ of the sequence $x(n)$ and the frequency interval between successive nulls?
4. Explain the difference between the plots obtained from parts (a) and (c).

**8.37** *Identification of pole positions in a system* Consider the system described by the difference equation

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

**(a)** Let $r = 0.9$ and $x(n) = \delta(n)$. Generate the output sequence $y(n)$ for $0 \le n \le 127$. Compute the $N = 128$-point DFT $\{Y(k)\}$ and plot $\{|Y(k)|\}$.

**(b)** Compute the $N = 128$-point DFT of the sequence

$$w(n) = (0.92)^{-n} y(n)$$

where $y(n)$ is the sequence generated in part (a). Plot the DFT values $|W(k)|$. What can you conclude from the plots in parts (a) and (b)?

**(c)** Let $r = 0.5$ and repeat part (a).

**(d)** Repeat part (b) for the sequence

$$w(n) = (0.55)^{-n} y(n)$$

where $y(n)$ is the sequence generated in part (c). What can you conclude from the plots in parts (c) and (d)?

**(e)** Now let the sequence generated in part (c) be corrupted by a sequence of "measurement" noise which is Gaussian with zero mean and variance $\sigma^2 = 0.1$. Repeat parts (c) and (d) for the noise-corrupted signal.