

Introduction to Orthogonal Transforms

with Applications in Data Processing and Analysis

Introduction to Orthogonal Transforms

with Applications in Data Processing and Analysis

Ruye Wang

March 28, 2011



CAMBRIDGE UNIVERSITY PRESS
Cambridge, New York, Melbourne, Madrid, Cape Town, Singapore, São Paulo
Cambridge University Press
The Edinburgh Building, Cambridge CB2 2RU, UK
Published in the United States of America by Cambridge University Press, New York

www.cambridge.org
Information on this title: www.cambridge.org/9780521XXXXXX

© Cambridge university Press 2011

This publication is in copyright. Subject to statutory exception
and to the provisions of relevant collective licensing agreements,
no reproduction of any part may take place without
the written permission of Cambridge University Press.

First published 2011

Printed in the United Kingdom at the University Press, Cambridge

A catalogue record for this publication is available from the British Library

Library of Congress Cataloguing in Publication data

ISBN-13 978-0-521-XXXXX-X hardback
ISBN-10 0-521-XXXXX-X hardback

Cambridge University Press has no responsibility for the persistence or accuracy of URLs for
external or third-party internet websites referred to in this publication, and does not
guarantee that any content on such websites is, or will remain, accurate or appropriate.

To my parents

Contents

<i>Preface</i>	<i>page</i> xi
<i>Notation</i>	xix
1 Signals and Systems	1
1.1 Continuous and Discrete Signals	1
1.2 Unit Step and Nascent Delta Functions	4
1.3 Relationship Between Complex Exponentials and Delta Functions	7
1.4 Attributes of Signals	9
1.5 Signal Arithmetics and Transformations	11
1.6 Linear and Time Invariant Systems	15
1.7 Signals Through LTI Systems (Continuous)	17
1.8 Signals Through LTI Systems (Discrete)	20
1.9 Continuous and Discrete Convolutions	23
1.10 Homework Problems	27
2 Vector Spaces and Signal Representation	32
2.1 Inner Product Space	32
2.1.1 Vector Space	32
2.1.2 Inner Product Space	34
2.1.3 Bases of Vector Space	41
2.1.4 Signal Representation by Orthogonal Bases	45
2.1.5 Signal Representation by Standard Bases	50
2.1.6 An Example: the Fourier Transforms	53
2.2 Unitary Transformation and Signal Representation	55
2.2.1 Linear Transformation	55
2.2.2 Eigenvalue problems	57
2.2.3 Eigenvectors of D^2 as Fourier Basis	59
2.2.4 Unitary Transformations	62
2.2.5 Unitary Transformations in N-D Space	64
2.3 Projection Theorem and Signal Approximation	68
2.3.1 Projection Theorem and Pseudo-Inverse	68
2.3.2 Signal Approximation	74
2.4 Frames and Biorthogonal Bases	78

2.4.1	Frames	78
2.4.2	Signal Expansion by Frames and Riesz Bases	80
2.4.3	Frames in Finite-Dimensional Space	87
2.5	Kernel Function and Mercer's Theorem	90
2.6	Summary	96
2.7	Homework Problems	98
3	Continuous-Time Fourier Transform	102
3.1	The Fourier Series Expansion of Periodic Signals	102
3.1.1	Formulation of The Fourier Expansion	102
3.1.2	Physical Interpretation	104
3.1.3	Properties of The Fourier Series Expansion	105
3.1.4	The Fourier Expansion of Typical Functions	107
3.2	The Fourier Transform of Non-Periodic Signals	114
3.2.1	Formulation	114
3.2.2	Relation to The Fourier Expansion	118
3.2.3	Properties of The Fourier Transform	120
3.2.4	Fourier Spectra of Typical Functions	127
3.2.5	The Uncertainty Principle	133
3.3	Homework Problems	137
4	Discrete-Time Fourier Transform	140
4.1	Discrete-Time Fourier Transform	140
4.1.1	Fourier Transform of Discrete Signals	140
4.1.2	Properties of DTFT	145
4.1.3	Discrete Time Fourier Transform of Typical Functions	150
4.1.4	The Sampling Theorem	154
4.1.5	Reconstruction by Interpolation	162
4.2	Discrete Fourier Transform	166
4.2.1	Formulation of DFT	166
4.2.2	Array Representation	171
4.2.3	Properties of DFT	176
4.2.4	DFT Computation and Fast Fourier Transform (FFT)	183
4.2.5	Four different forms of Fourier transform	188
4.3	Two-Dimensional Fourier Transform	192
4.3.1	Two-Dimensional Signals and Their Spectra	192
4.3.2	Fourier Transform of Typical 2-D Functions	196
4.3.3	Four Forms of 2-D Fourier Transform	200
4.3.4	Computation of the 2-D DFT	201
4.4	Homework Problems	208
5	Applications of the Fourier Transforms	212
5.1	LTI Systems in Time and Frequency Domains	212

5.2	Solving Differential and Difference Equations	215
5.3	Magnitude and Phase Filtering	221
5.4	Implementation of 1-D Filtering	226
5.5	Implementation of 2-D Filtering	236
5.6	Hilbert Transform and Analytic Signals	242
5.7	Radon Transform and Image Restoration from Projections	247
5.8	Orthogonal Frequency Division Modulation (OFDM)	255
5.9	Homework Problems	258
6	The Laplace and z-Transforms	262
6.1	The Laplace Transform	262
6.1.1	From Fourier Transform to Laplace Transform	262
6.1.2	The Region of Convergence	265
6.1.3	Properties of the Laplace Transform	266
6.1.4	Laplace Transform of Typical Signals	269
6.1.5	Analysis of Continuous LTI Systems by Laplace Transform	271
6.1.6	First Order System	277
6.1.7	Second Order System	279
6.1.8	The Unilateral Laplace Transform	291
6.2	The z-Transform	295
6.2.1	From Discrete Time Fourier Transform to z-Transform	295
6.2.2	Region of Convergence	298
6.2.3	Properties of the z-Transform	300
6.2.4	z-Transform of Typical Signals	305
6.2.5	Analysis of Discrete LTI Systems by z-Transform	306
6.2.6	First and Second Order Systems	312
6.2.7	The Unilateral z-Transform	315
6.3	Homework Problems	320
7	Fourier Related Orthogonal Transforms	324
7.1	The Hartley Transform	324
7.1.1	Continuous Hartley Transform	324
7.1.2	Properties of the Hartley Transform	326
7.1.3	Hartley Transform of Typical Signals	328
7.1.4	Discrete Hartley Transform	330
7.1.5	2-D Hartley Transform	334
7.2	The Discrete Sine and Cosine Transforms	338
7.2.1	The Continuous Cosine and Sine Transforms	339
7.2.2	From the DFT to the DCT and DST	340
7.2.3	Matrix Forms of DCT and DST	344
7.2.4	Fast Algorithms for DCT and DST	348
7.2.5	The DCT and DST Filtering	354
7.2.6	The Two-Dimensional DCT and DST	356

7.3	Homework Problems	361
8	The Walsh-Hadamard, Slant and Haar Transforms	363
8.1	The Walsh-Hadamard Transform	363
8.1.1	Hadamard Matrix	363
8.1.2	Hadamard Ordered Walsh-Hadamard Transform (WHT _h)	365
8.1.3	Fast Walsh-Hadamard Transform Algorithm	366
8.1.4	Sequency Ordered Walsh-Hadamard Matrix (WHT _w)	368
8.1.5	Fast Walsh-Hadamard Transform (Sequency Ordered)	370
8.2	The Slant Transform	375
8.2.1	Slant Matrix	375
8.2.2	Slant Transform and Its Fast Algorithm	378
8.3	The Haar Transform	382
8.3.1	Continuous Haar Transform	382
8.3.2	Discrete Haar Transform	384
8.3.3	Computation of Discrete Haar Transform	387
8.3.4	Filter Bank Implementation	390
8.4	Two-dimensional Transforms	392
8.5	Homework Problems	395
9	Karhunen-Loeve Transform and Principal Component Analysis	397
9.1	Stochastic Process and Signal Correlation	397
9.1.1	Signals as Stochastic Processes	397
9.1.2	Signal Correlation	400
9.2	Karhunen-Loeve Transform (KLT)	402
9.2.1	Continuous Karhunen-Loeve Theorem	402
9.2.2	Discrete Karhunen-Loeve Transform	403
9.2.3	Optimalities of the KLT	404
9.2.4	Geometric Interpretation of the KLT	407
9.2.5	Principal Component Analysis (PCA)	410
9.2.6	Comparison with Other Orthogonal Transforms	411
9.2.7	Approximation of KLT by DCT	416
9.3	Applications of the KLT	422
9.3.1	Image processing and analysis	422
9.3.2	Feature Extraction for Pattern Classification	428
9.4	Singular Value Decomposition Transform	433
9.4.1	Singular Value Decomposition (SVD)	433
9.4.2	Application in Image Compression	436
9.5	Homework Problems	437
10	Continuous and Discrete-time Wavelet Transforms	444
10.1	Why Wavelet?	444
10.1.1	Short-time Fourier Transform and Gabor Transform	444

10.1.2	The Heisenberg Uncertainty	445
10.2	Continuous-Time Wavelet Transform (CTWT)	447
10.2.1	Mother and Daughter Wavelets	447
10.2.2	The Forward and Inverse Wavelet Transforms	449
10.3	Properties of the CTWT	451
10.4	Typical Mother Wavelet Functions	455
10.5	Discrete-Time Wavelet Transform (DTWT)	457
10.5.1	Discretization of Wavelet Functions	457
10.5.2	The Forward and Inverse Transform	459
10.5.3	A Fast Inverse Transform Algorithm	461
10.6	Wavelet Transform Computation	464
10.7	Filtering Based on Wavelet Transform	467
10.8	Homework Problems	474
11	Multiresolution Analysis and Discrete Wavelet Transform	476
11.1	Multiresolution Analysis (MRA)	476
11.1.1	Scale Spaces	476
11.1.2	Wavelet Spaces	481
11.1.3	Properties of the Scaling and Wavelet Filters	485
11.1.4	Relationship Between Scaling and Wavelet Filters	489
11.1.5	Wavelet Series Expansion	491
11.1.6	Construction of Scaling and Wavelet Functions	493
11.2	Discrete Wavelet Transform (DWT)	503
11.2.1	Discrete Wavelet Transform (DWT)	503
11.2.2	Fast Wavelet Transform (FWT)	506
11.3	Filter Bank Implementation of DWT and Inverse DWT	509
11.3.1	Two-Channel Filter Bank and Inverse DWT	509
11.3.2	Two-Dimensional DWT	515
11.4	Applications in Filtering and Compression	520
11.5	Homework Problems	526
12	Appendix 1: Review of Linear Algebra	531
12.1	Basic Definitions	531
12.2	Eigenvalues and Eigenvectors	536
12.3	Hermitian Matrix and Unitary Matrix	537
12.4	Toeplitz and Circulant Matrices	539
12.5	Vector and Matrix Differentiation	540
13	Appendix 2: Review of Random Variables	542
13.1	Random Variables	542
13.2	Multivariate Random Variables	544
13.3	Stochastic Models	549

14	Bibliography	552
	<i>Index</i>	553

Preface

What Is the Book All about?

“When a straight line standing on a straight line makes the adjacent angles equal to one another, each of the equal angles is right, and the straight line standing on the other is called a *perpendicular* to that on which it stands.”

— Euclid, *Elements, Book 1, definition 10*

This is Euclid’s definition for “perpendicular”, which is synonymous to the word “orthogonal” used in the title of this book. Although the meaning of this word has been generalized since Euclid’s time to describe the relationship between two functions as well as two vectors, as what we will be mostly concerned with in this book, they are essentially no different from two perpendicular straight lines, as discussed by Euclid some 23 centuries ago.

Orthogonality is of important significance not only in geometry and mathematics, but also in science and engineering in general, and in data processing and analysis in particular. This book is about a set of mathematical and computational methods, known collectively as the orthogonal transforms, that enables us to take advantage of the orthogonal axes of the space in which the data reside. As we will see through out the book, such orthogonality is a much desired property that can keep things untangled and nicely separated for ease of manipulation, and an orthogonal transform can rotate a signal, represented as a vector in a Euclidean, or more generally, Hilbert space, in such a way that the signal components tend to become, approximately or accurately, orthogonal to each other. Such orthogonal transforms, typified by the most well known Fourier transform, lend themselves well to various data processing and analysis needs, and are therefore used in a wide variety of disciplines and areas including both social and natural sciences as well as engineering. The book also covers the Laplace and Z transforms, which can be considered as the extended versions of the Fourier transform for continuous and discrete functions, respectively, and the wavelet transforms that may not be strictly orthogonal but are still closely related to those that are.

In the last few decades the scales of data collection across almost all fields have been increasing drastically due mostly to the rapid advances in technologies. Consequently how to best make sense of the fast accumulating data has become more challenging than ever. Wherever a large amount of data is collected, from stock market indices in economy to microarray data in bioinformatics, from seismic

data in geophysics to audio and video data in communication and broadcasting engineering, there is always the need to process, analyze and compress the data in some meaningful way for the purpose of effective and efficient data transmission, interpretation and storage, by various computational methods and algorithms. The transform methods discussed in this book can be used as a set of basic tools for the data processing and the subsequent analysis such as data mining, knowledge discovery, and machine learning.

The specific purpose of each data processing and analysis task at hand may vary from case to case. From a set of given data, one may desire to remove certain type of noise, extract a particular kind of features of interest, and/or reduce the quantity of the data without losing useful information for storage and transmission. On the other hand, many operations needed for achieving these very different goals may all be carried out using the same mathematical tool of orthogonal transform, by which the data is manipulated and represented in such a way that the desired results can be achieved effectively in the subsequent stage. To address all such needs, this book presents a thorough introduction to the mathematical background common to these transform methods, and provides a repertoire of computational algorithms for these methods.

The basic approach of the book is the combination of the theoretical derivation and practical implementation of each of transform method considered. Certainly many existing books touch upon the topics of both orthogonal and wavelet transforms, from either mathematical or engineering point of view. Some of them may concentrate on the theories of these methods, while others may emphasize their applications, but relatively few would guide the reader directly from the mathematical theories to the computational algorithms, and then to their applications to real data analysis, as what this book intends to do. Here deliberate efforts are made to bridge the gap between the theoretical background and the practical implementation, based on the belief that to truly understand a certain method, one needs ultimately to be able to convert the mathematical theory into computer code for the algorithms to be actually implemented and tested. This idea has been the guiding principle through out the writing of the book. For each of the methods covered, we will first derive the theory mathematically, then present the corresponding computational algorithm, and finally provide the necessary code segments in Matlab or C for the key parts of the algorithm. Moreover, we will also include some relatively simple application examples to illustrate the actual data processing effects of the algorithm. In fact every one of the transform methods considered in the book has been implemented by either Matlab or C programming language and tested on real data. The complete programs are also made readily available in the CD attached to the book, as well as a website dedicated to the book at: www.cambridge.org/orthogonaltransforms. The reader is encouraged and expected to try these algorithms out by running the code on his/her own data.

Why Orthogonal Transforms?

The transform methods covered in the book are a collection of both old and new ideas ranging from the classical Fourier series expansion that goes back almost 200 years, to some relatively recent thoughts such as the various origins of the wavelet transform. While all of these ideas were originally developed with different goals and applications in mind, from solving the heat equation to the analysis of seismic data, they can all be considered to belong to the same family, based on the common mathematical frame work they all share, and their similar applications in data processing and analysis. The discussions of specific methods and algorithms in the chapters will all be approached from such a unified point of view.

Before the specific discussion of each of the methods, let us first address a fundamental issue: why do we need to carry out an orthogonal transform to start with? As the measurement of a certain variable, e.g., the temperature, of a physical process, a signal tends to vary continuously and smoothly, as the energy associated with the physical process is most probably distributed relatively evenly in both space and time. Most such spatial or temporal signals are likely to be correlated, in the sense that given the value of a signal at a certain point in space or time, one can predict with reasonable confidence that the signal at a neighboring point will take a similar value. Such everyday experience is due to the fundamental nature of the physical world governed by the principles of minimum energy and maximum entropy, in which any abruptness and discontinuities, typically caused by energy surge of some kind, are relatively rare and unlikely events (except in the microscopic world governed by quantum mechanics). On the other hand, from the signal processing view point, the high signal correlation and even energy distribution are not desirable in general, as it is difficult to decompose such a signal, as needed in various applications such as information extraction, noise reduction and data compression. The issue therefore becomes, how can the signal be converted in such a way that it is less correlated and its energy less evenly distributed, and to what extent such a conversion can be carried out to achieve the goal.

Specifically, in order to represent, process and analyze a signal, it needs to be decomposed into a set of components along a certain dimension. While typically a signal is represented by default as a continuous or discrete function of time or space, it may be desirable to represent it along some alternative dimension, most commonly frequency, so that it can be processed and analyzed more effectively and conveniently. Viewed mathematically, a signal is a vector in some vector space which can be represented by any of a set of different orthogonal bases all spanning the same space. Each representation corresponds to a different decomposition of the signal. Moreover, all such representations are equivalent in the sense that they are related to each other by certain rotation in the space by which the total energy or information contained in the signal is conserved. From this point of view, all different orthogonal transform methods developed in the last two hundred years by mathematicians, scientists and engineers for

various purposes can be unified to form a family of methods for the same general purpose.

While all transform methods are equivalent as they all conserve the total energy or information of the signal, they can be very different in terms of how the total energy or information in the signal is redistributed among its components after the transform, and how much these components are correlated. If, after a properly chosen orthogonal transform, the signal is represented in such a way that its components are decorrelated and most of the signal information of interest is concentrated in a small subset of its components, then the remaining components could be neglected as they carry little information. This simple idea is essentially the answer to the question asked above: why an orthogonal transform is needed, and it is actually the foundation of the general orthogonal transform method for feature selection, data compression, and noise reduction. In a certain sense, once a proper basis of the space is chosen so that the signal is represented in such a favorable manner, the signal-processing goal is already achieved to a significant extent.

What Is in the Chapters?

The purpose of the first two chapters is to establish a solid mathematical foundation for the thorough understanding of the topics of the subsequent chapters each discussing a specific type of transform method. Chapter 1 is a brief summary of the basic concepts of signals and linear time-invariant (LTI) systems. For readers with an engineering background, much of this chapter may be a quick review that could be scanned through or even skipped. For others this chapter serves as an introduction to the mathematical language by which the signals and systems will be described in the following chapters.

Chapter 2 sets up the stage for all transform methods by introducing the key concepts of the vector space, or more strictly speaking, Hilbert space, and the linear transformations in such a space. Here a usual N-dimensional space can be generalized in several aspects: (1) the dimension N of the space may be extended to infinity, (2) a vector space may also include a function space composed of all continuous functions satisfying certain conditions, and (3) the basis vectors of a space may become uncountable. The mathematics needed for a rigorous treatment of these much-generalized spaces is likely to be beyond the comfort zone of most readers with typical engineering or science background, and it is therefore also beyond the scope of this book. The emphasis of the discussion here is not mathematical rigor, but the basic understanding and realization that many of the properties of these generalized spaces are just the natural extensions of those of the familiar N-D vector space. The purpose of such discussions is to establish a common foundation for all transform methods so that they can all be studied from a unified point of view, namely, any given signal, either continuous or discrete, with either finite or infinite duration, can be treated as a vector in a certain space and represented differently by any of a variety of orthogonal transform methods, each corresponding to one of the orthogonal bases that span the space. Moreover, all of these different representations are related to each

other by rotations in the space. Such basic ideas may also be extended to non-orthogonal (e.g., biorthogonal) bases that are used in wavelet transforms. All transform methods considered in later chapters will be studied in light of such a frame work. Although it is highly recommended for the reader to at least read through the materials in the first two chapters, those who feel difficult to thoroughly follow the discussions could skip them and move on to the following chapters, as many of the topics could be studied relatively independently, and one can always come back to learn some of the concepts in the first two chapters when needed.

In Chapters 3 and 4, we study the classical Fourier methods for continuous and discrete signals respectively. Fourier's theory is mathematically beautiful and is referred to as "mathematical poem", and it has great significance through out a wide variety of disciplines in practice as well as in theory. While the general topic of the Fourier transform is covered in a large number of textbooks in various fields such as engineering, physics, and mathematics, here a not-so-conventional approach is adopted to treat all Fourier related methods from a unified point of view. Specifically, the Fourier series (FS) expansion, the continuous and discrete-time Fourier transforms (CTFT and DTFT), and the discrete Fourier transform (DFT), will be considered as four different variations of the same general Fourier transform, corresponding to the four combinations of the two basic categories of signals: continuous versus discrete, periodic versus non-periodic. By doing so, many of the dual and symmetrical relationships among these four different forms and between time and frequency domains of the Fourier transform can be much more clearly and conveniently presented and understood.

Chapter 5 discusses the Laplace and Z transforms. Strictly speaking, these transforms do not belong to the family of orthogonal transforms, which convert a 1-dimensional signal of time t into another 1-dimensional function along a different variable, typically, frequency f or angular frequency $\omega = 2\pi f$. Instead, the Laplace converts a 1-D continuous signal from time domain into a function in a 2-D complex plane $s = \sigma + j\omega$, and the Z-transforms converts a 1-D discrete signal from time domain into a function in a 2-D complex plane $z = e^s$. However, as these transforms are respectively the natural extensions of the continuous and discrete-time Fourier transforms, and are widely used in signal processing and system analysis, they are included in the book as two extra tools in our toolbox.

Chapter 6 discusses the Hartley and sine/cosine transforms, both of which are closely related to the Fourier transform. As real transforms, both Hartley and sine/cosine transforms have the advantage of reduced computational cost when compared with the complex Fourier transform. If the signal in question is real with zero imaginary part, then half of the computation in its Fourier transform is redundant and therefore wasted. However, this redundancy is avoided by a real transform such as the cosine transform, which is widely used for data compression, such as in the image compression standard JPEG.

Chapter 7 combines three transform methods, the Walsh-Hadamard, slant, and Haar transforms, all sharing some similar characteristics, i.e., the basis functions

associated with these transforms all have square-wave like waveforms. Moreover, as the Haar transform also possesses the basic characteristics of the wavelet transform method, it can also serve as a bridge between the two camps of the orthogonal transforms and the wavelet transforms, leading a natural transition from the former to the latter.

In Chapter 8 we discuss the Karhunen-Loeve transform (KLT), which can be considered as a capstone of all previously discussed transform methods, and the associated principal component analysis (PCA), which is popularly used in many data processing applications. The KLT is the optimal transform method among all orthogonal transforms in terms of the two main characteristics of the general orthogonal transform method, namely, the compaction of signal energy and the decorrelation among all signal components. In this regard, all orthogonal transform methods can be compared against the optimal KLT for an assessment of their performances.

We next consider in Chapter 9 both the continuous and discrete-time wavelet transforms (CTWT and DTWT), which differ from all orthogonal transforms discussed previously in two main aspects. First, the wavelet transforms are not strictly orthogonal as the bases used to span the vector space and to represent a given signal may not be necessarily orthogonal. Second, the wavelet transform converts a 1-D time signal into a 2-D function of two variables, one for different levels of details or scales, corresponding to different frequencies in the Fourier transform, while the other for different temporal positions, which is totally absent in the Fourier or any other orthogonal transform. While redundancy is inevitably introduced into the 2-D transform domain by such a wavelet transform, the additional second dimension also enables the transform to achieve both temporal and frequency localities in signal representation at the same time (while all other transform methods can only achieve either one of the two localities). Such a capability of the wavelet transform is its main advantage over orthogonal transforms in some applications such as signal filtering.

Finally in Chapter 10, we introduce the basic concept of multiresolution analysis (MRA), and Mallat's fast algorithm for the discrete wavelet transform (DWT) together with its filter bank implementation. Similar to the orthogonal transforms, this algorithm converts a discrete signal of size N into a set of DWT coefficients also of size N , from which the original signal can be perfectly reconstructed, i.e., there is no redundancy introduced by the DWT. However, different from orthogonal transforms, the DWT coefficients represent the signal with temporal as well as frequency (levels of details) localities, and can therefore be more advantageous in some applications such as data compressions.

Moreover, some fundamental results in linear algebra and statistics are also summarized in the two appendices in the back of the book.

Who Are the Intended Readers?

The book can be used as a textbook for either an undergraduate or graduate course in digital signal processing, communication, or other related areas. In such a classroom setting, all orthogonal transform methods can be systemati-

cally studied following a thorough introduction of the mathematical background common to these methods. The mathematics prerequisite is no more than basic calculus and linear algebra. Moreover, the book can also be used as a reference by practicing professionals in both natural and social sciences, as well as engineering. A financial analyst or a biologist may need to learn how to effectively analyze and interpret his/her data, a database designer may need to know how to compress his data before storing them in the database, and a software engineer may need to learn the basic data processing algorithms while developing a software tool in the field. In general, anyone who deals with a large quantity of data may desire to gain some basic knowledge in data processing, regardless of his/her backgrounds and specialties. In fact the book has been developed with such potential readers in mind. Due possibly to the personal experience, the author always feels that self-learning (or to borrow a machine learning terminology, “unsupervised learning”) is no less important than formal classroom learning. One may have been out of school for some years but still feel the need to update and expand his/her knowledge. Such readers could certainly study whichever chapters of interest, instead of systematically reading through each chapter from beginning to end. They can also skip certain mathematical derivations, which are included in the book for completeness (and for those who feel comfortable only if the complete proof and derivations of all conclusions are provided). For some readers, neglecting much of the mathematical discussion for a specific transform method should be just fine if the basic ideas regarding the method and its implementation are understood. It is hoped that the book can serve as a toolbox, as well as a textbook, from which certain transform methods of interest can be learned and applied, in combination with the reader’s expertise in his/her own field, to solving the specific data processing/analysis problems at hand.

About the Homework Problems and Projects

Understanding the transform methods and the corresponding computational algorithms is not all. Eventually they all need to be implemented and realized by either software or hardware, specifically by computer code of some sort. This is why the book emphasizes the algorithm and coding as well as theoretical derivation, and many homework problems and projects require certain basic coding skills, such as some knowledge in Matlab. However, being able to code is not expected of all readers. Those who may not need or wish to learn coding can by all means skip the sections in the text as well as those homework problems involving software programming. However, all readers are encouraged to at least run some of the Matlab functions provided to see the effects of the transform methods. (There are a lot of such Matlab m-files on the website of the book. In fact, all functions used to generate many of the figures in the book are provided on the site.) If a little more interested, the reader can read through the code to see how things are done. Of course a step further is to modify the code, use different parameters and different datasets to better appreciate the various effects of the algorithms.

Back to Euclid

Finally let us end by again quoting Euclid, this time, a story about him. A youth who had begun to study geometry with Euclid, when he had learned the first proposition, asked, "What do I get by learning these things?" So Euclid called a slave and said "Give him three pence, since he must make a gain out of what he learns." Surely explicit efforts are made in this book to discuss the practical uses of the orthogonal transforms as well as the mathematics behind them, but one should realize that after all the book is about a set of mathematical tools, just like those propositions in Euclid's geometry, out of learning which the reader may not be able to make a direct and immediate gain. However, in the end, it is the application of these tools toward solving specific problems in practice that will enable the reader to make a gain out of the book, much more than three pence, hopefully.

Acknowledgment

The author is in debt to two of his colleagues Professors John Molinder and Ellis Cumberbatch for their support and help with the book project. In addition to our discussions regarding some of the topics in the book, John provided the application example of orthogonal frequency division modulation (OFDM) discussed in section 5.8, together with the Matlab code that is used in a homework problem. Also Ellis read through the first two chapters of the manuscript and made numerous suggestions for the improvement of the coverage of the topics in these two chapters. All such valuable help and support are greatly appreciated.

Notation

General notation

iff		if and only if
$j = \sqrt{-1} = e^{j\pi/2}$		imaginary unit
$\overline{u + jv} = u - jv$		complex conjugate of $u + jv$
$Re(u + jv) = u$		real part of $u + jv$
$Im(u + jv) = v$		imaginary part of $u + jv$
$ u + jv = \sqrt{u^2 + v^2}$		magnitude (absolute value) of a complex value $u + jv$
$\angle(u + jv) = \tan^{-1}(v/u)$		phase of $u + jv$
$\boldsymbol{x}_{n \times 1}$		an n by 1 column vector
$\overline{\boldsymbol{x}}$		complex conjugate of \boldsymbol{x}
\boldsymbol{x}^T		transpose of \boldsymbol{x} , a 1 by n row vector
$\boldsymbol{x}^* = \overline{\boldsymbol{x}}^T$		conjugate transpose of matrix \boldsymbol{A}
$\ \boldsymbol{x}\ $		norm of vector \boldsymbol{x}
$\boldsymbol{A}_{m \times n}$		an m by n matrix of m rows and n columns
$\overline{\boldsymbol{A}}$		complex conjugate of matrix \boldsymbol{A}
\boldsymbol{A}^{-1}		inverse of matrix \boldsymbol{A}
\boldsymbol{A}^T		transpose of matrix \boldsymbol{A}
$\boldsymbol{A}^* = \overline{\boldsymbol{A}}^T = \overline{\boldsymbol{A}}^T$		conjugate transpose of matrix \boldsymbol{A}
\mathbb{N}		set of all positive integers including 0
\mathbb{Z}		set of all real integers
\mathbb{R}		set of all real numbers
\mathbb{C}		set of all complex numbers
\mathbb{R}^N		N -dimensional Euclidean space
\mathbb{C}^N		N -dimensional unitary space
L^2		space of all square-integrable functions
l^2		space of all square-summable infinite vectors (sequences)
$x(t)$		a function representing a continuous signal
$\boldsymbol{x} = [\dots, x[n], \dots]^T$		a vector representing a discrete signal
$\dot{x}(t) = dx(t)/dt$		first order time derivative of $x(t)$
$\ddot{x}(t) = dx^2/dt^2$		second order time derivative of $x(t)$
f		frequency (cycle per unit time)
$\omega = 2\pi f$		angular frequency (radian per unit time)

Through the book, angular frequency ω will be used interchangeably with $2\pi f$, whichever more convenient in the context of discussion.

As a convention, a bold-faced lower case letter \mathbf{x} is typically used to represent a vector, while a bold-faced upper case letter \mathbf{A} represents a matrix, unless otherwise noted.

1 Signals and Systems

In the first two chapters, we will consider some basic concepts and ideas as the mathematical background for the specific discussions of the various orthogonal transform methods in the subsequent chapters. Here we will set up a framework common to all such methods, so that they can be studied from a unified point of view. While some discussions here may seem mathematical, the emphasis is the intuitive understanding, instead of the theoretical rigor.

1.1 Continuous and Discrete Signals

A physical signal can always be represented as a real or complex-valued continuous function of time $x(t)$ (unless otherwise specified, such as a function of space). The continuous signal can be sampled to become a discrete signal $x[n]$. If the time interval between two consecutive samples is assumed to be Δ , then the nth sample is:

$$x[n] = x(t)|_{t=n\Delta} = x(n\Delta) \quad (1.1)$$

In either continuous or discrete case, a signal can be assumed in theory to have infinite duration, i.e., $-\infty < t < \infty$ for $x(t)$ and $-\infty < n < \infty$ for $x[n]$. However, any signal in practice is finite and can be considered as the truncated version of a signal of infinite duration. We typically assume $0 \leq t \leq T$ for a finite continuous signal $x(t)$, and $1 \leq n \leq N$ (or sometimes $0 \leq n \leq N - 1$ for certain convenience) for a discrete signal $x[n]$. The value of such a finite signal $x(t)$ is not defined if $t < 0$ or $t > T$, similarly $x[n]$ is not defined if $n < 0$ or $n > N$. However, for mathematical convenience sometimes we could assume a finite signal to be periodic, i.e., $x(t + T) = x(t)$ and $x[n + N] = x[n]$.

A discrete signal can also be represented as a vector $\mathbf{x} = [\dots, x[n-1], x[n], x[n+1], \dots]^T$ of finite or infinite dimensions composed of all of its samples or components as the vector elements. We will always represent a discrete signal as a column vector (transpose of a row vector).

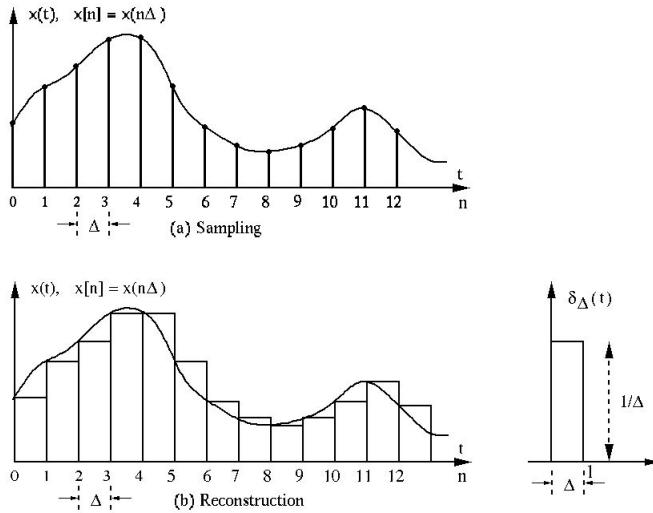


Figure 1.1 Sampling and reconstruction of a continuous signal

Definition 1.1. The discrete unit impulse or Kronecker delta function is defined as:

$$\delta[n] = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases} \quad (1.2)$$

Based on this definition, a discrete signal can be represented as:

$$x[n] = \sum_{m=-\infty}^{\infty} x[m] \delta[n - m], \quad (n = 0, \pm 1, \pm 2, \dots) \quad (1.3)$$

This equation can be interpreted in two conceptually different ways.

- First, a discrete signal $x[n]$ can be decomposed into a set of unit impulses each at a different moment $n = m$ and weighted by the signal amplitude $x[m]$ at the moment, as shown in Fig.1.1(a).
- Second, the Kronecker delta $\delta[n - m]$ acts as a filter that sifts out a particular value of the signal $x[n]$ at the moment $m = n$ from a sequence of signal samples $x[m]$ for all m . This is the *sifting property* of the Kronecker delta.

In a similar manner, a continuous signal $x(t)$ can also be represented by its samples. We first define a unit square impulse function as:

$$\delta_{\Delta}(t) = \begin{cases} 1/\Delta & 0 \leq t < \Delta \\ 0 & \text{else} \end{cases} \quad (1.4)$$

Note that the width and height of this square impulse are respectively Δ and $1/\Delta$, i.e, it covers a unit area $\Delta \times 1/\Delta = 1$, independent of the value of Δ :

$$\int_{-\infty}^{\infty} \delta_{\Delta}(t) dt = \Delta \cdot 1/\Delta = 1 \quad (1.5)$$

Now a continuous signal $x(t)$ can be approximated as a sequence of square impulses $\delta_{\Delta}(t - n\Delta)$ weighted by the sample value $x[n]$ for the amplitude of the signal at the moment $t = n\Delta$:

$$x(t) \approx \hat{x}(t) = \sum_{n=-\infty}^{\infty} x[n] \delta_{\Delta}(t - n\Delta)\Delta \quad (1.6)$$

This is shown in Fig.1.1(b).

The approximation $\hat{x}(t)$ above will become a perfect reconstruction of the signal if we take the limit $\Delta \rightarrow 0$, so that the square impulse becomes a *continuous unit impulse* or *Dirac delta*:

$$\lim_{\Delta \rightarrow 0} \delta_{\Delta}(t) = \delta(t) \quad (1.7)$$

which is also formally defined as below:

Definition 1.2. *The continuous unit impulse or Dirac delta function $\delta(t)$ is a function that has an infinite height but zero width at $t = 0$, and it covers a unit area, i.e., it satisfies the following two conditions:*

$$\delta(t) = \begin{cases} \infty & t = 0 \\ 0 & t \neq 0 \end{cases}, \quad \text{and} \quad \int_{-\infty}^{\infty} \delta(t) dt = 1 \quad (1.8)$$

Now at the limit $\Delta \rightarrow 0$, the summation in the approximation of Eq.1.6 above becomes an integral, the square impulse becomes a Dirac delta, and the approximation becomes a perfect reconstruction of the continuous signal:

$$x(t) = \lim_{\Delta \rightarrow 0} \sum_{n=-\infty}^{\infty} x[n] \delta_{\Delta}(t - n\Delta)\Delta = \int_{-\infty}^{\infty} x(\tau) \delta(t - \tau) d\tau \quad (1.9)$$

In particular, when $t = 0$, Eq.1.9 becomes:

$$x(0) = \int_{-\infty}^{\infty} x(\tau) \delta(\tau) d\tau \quad (1.10)$$

Eq.1.9 can be interpreted in two conceptually different ways.

- First, a continuous signal $x(t)$ can be decomposed into a set of infinitely many uncountable unit impulses each at a different moment $t = \tau$, weighted by the signal intensity $x(\tau)$ at the moment $t = \tau$.
- Second, the Dirac delta $\delta(\tau - t)$ acts as a filter that sifts out the value of $x(t)$ at the moment $\tau = t$ from a sequence of infinite uncountable signal samples. This is the *sifting property* of the Dirac delta.

Note that the discrete impulse function $\delta[n]$ has a unit height, while the continuous impulse function $\delta(t)$ has a unit area (product of height and width for time), i.e., the two types of impulses have different dimensions. The dimension of the discrete impulse is the same as that of the signal (e.g., voltage), while the dimension of the continuous impulse is the signal's dimension divided by time (e.g., voltage/time). In other words, $x(\tau)\delta(t - \tau)$ represents the density of the signal at $t = \tau$, only when integrated over time will the continuous impulse functions have the same dimension as the signal $x(t)$.

The results above indicate that a time signal, either continuous or discrete, can be decomposed in time domain to become a linear combination, either an integral or a summation, of a sequence of time impulses (or components). However, as we will see in the future chapters, the decomposition of the time signal is not unique. The signal can also be decomposed in different domains other than time, such as frequency, and the representations of the signal in different domains are related by certain orthogonal transformations.

1.2 Unit Step and Nascent Delta Functions

The *discrete unit step function* defined below is an important function to be used frequently in the future:

Definition 1.3.

$$u[n] = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases} \quad (1.11)$$

The Kronecker delta can be obtained as the first order difference of the unit step function:

$$\delta[n] = u[n] - u[n - 1] = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases} \quad (1.12)$$

Similarly, in continuous case, the impulse function $\delta(t)$ is also closely related to the *continuous unit step function* (also called *Heaviside step function*) $u(t)$. To see this, we first consider a piece-wise linear function defined as:

$$u_{\Delta}(t) = \begin{cases} 0 & t < 0 \\ t/\Delta & 0 \leq t < \Delta \\ 1 & t \geq \Delta \end{cases} \quad (1.13)$$

Taking the time derivative of this function, we get the square impulse considered before in Eq.1.4:

$$\delta_{\Delta}(t) = \frac{d}{dt}u_{\Delta}(t) = \begin{cases} 0 & t < 0 \\ 1/\Delta & 0 \leq t < \Delta \\ 0 & t \geq \Delta \end{cases} \quad (1.14)$$

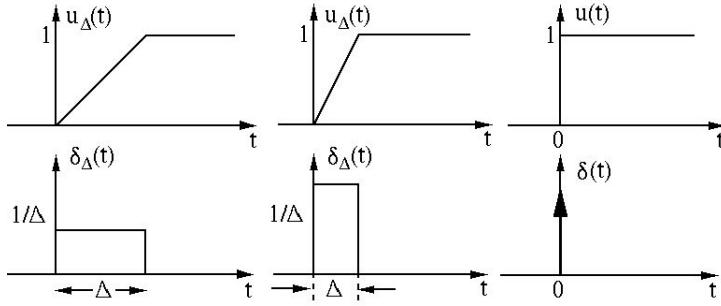


Figure 1.2 Generation of unit step and unit impulse

If we let Δ approach zero $\Delta \rightarrow 0$, then $u_\Delta(t)$ becomes the *unit step function* $u(t)$ at the limit:

Definition 1.4.

$$u(t) = \lim_{\Delta \rightarrow 0} u_\Delta(t) = \begin{cases} 0 & t < 0 \\ 1/2 & t = 0 \\ 1 & t > 0 \end{cases} \quad (1.15)$$

Here we define $u(0) = 1/2$ at $t = 0$ for reasons to be discussed in the future.¹ Also, at the limit $\Delta \rightarrow 0$, $\delta_\Delta(t)$ becomes Dirac delta discussed above:

$$\delta(t) = \lim_{\Delta \rightarrow 0} \delta_\Delta(t) = \begin{cases} \infty & t = 0 \\ 0 & t \neq 0 \end{cases} \quad (1.16)$$

Therefore by taking the limit $\Delta \rightarrow 0$ on both sides of Eq.1.14 we obtain a useful relationship between $u(t)$ and $\delta(t)$:

$$\frac{d}{dt}u(t) = \delta(t), \quad u(t) = \int_{-\infty}^t \delta(\tau)d\tau \quad (1.17)$$

In addition to the square impulse $\delta_\Delta(t)$, the Dirac delta can also be generated from a variety of different *nascent delta functions* at the limit when a certain parameter of the function approaches the limit of either zero or infinity. Consider, for example, the Gaussian function:

$$g(t) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-t^2/2\sigma^2} \quad (1.18)$$

which is the probability density function of a normally distributed random variable t with zero mean and variance σ^2 . Obviously the area underneath this density function is always one, independent of σ :

$$\int_{-\infty}^{\infty} g(t)dt = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} e^{-t^2/2\sigma^2} dt = 1 \quad (1.19)$$

¹ Although in some literatures it could be alternatively defined as either $u(0) = 0$ or $u(0) = 1$.

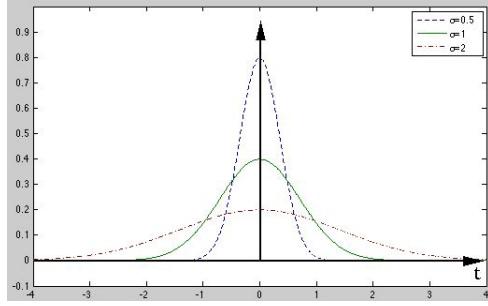


Figure 1.3 Gaussian functions with different σ values

At the limit $\sigma \rightarrow 0$, this Gaussian function $g(t)$ becomes infinity when $t = 0$ but it is zero for all $t \neq 0$, i.e., it becomes the unit impulse function:

$$\lim_{\sigma \rightarrow 0} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-t^2/2\sigma^2} dt = \delta(t) \quad (1.20)$$

The argument t of a Dirac delta $\delta(t)$ may be scaled so that it becomes $\delta(at)$. In this case Eq.1.10 becomes:

$$\int_{-\infty}^{\infty} x(\tau) \delta(a\tau) d\tau = \int_{-\infty}^{\infty} x\left(\frac{u}{a}\right) \delta(u) \frac{1}{|a|} du = \frac{1}{|a|} x(0) \quad (1.21)$$

where we have defined $u = a\tau$. Comparing this result with Eq.1.10, we see that

$$\delta(at) = \frac{1}{|a|} \delta(t), \quad \text{i.e.} \quad |a| \delta(at) = \delta(t) \quad (1.22)$$

For example, a delta function $\delta(f)$ in frequency can also be expressed as a function of angular frequency $\omega = 2\pi f$ as:

$$\delta(f) = 2\pi \delta(\omega) \quad (1.23)$$

More generally, the Dirac delta may also be defined over a function $f(t)$, instead of a variable t , so that it become $\delta(f(t))$, which is zero except when $f(t) = 0$, i.e., when $t = t_k$ is one of the roots of $f(t)$ ($f(t_k) = 0$). To see how such an impulse is scaled, consider the following integral:

$$\int_{-\infty}^{\infty} x(\tau) \delta(f(\tau)) d\tau = \int_{-\infty}^{\infty} x(\tau) \delta(u) \frac{1}{|f'(\tau)|} du \quad (1.24)$$

where we have changed the integral variable from τ to $u = f(\tau)$. If $\tau = \tau_0$ is the only root of $f(\tau)$, i.e., $u = f(\tau_0) = 0$, then the integral above becomes:

$$\int_{-\infty}^{\infty} x(\tau) \delta(f(\tau)) d\tau = \frac{x(\tau_0)}{|f'(\tau_0)|} \quad (1.25)$$

If $f(\tau)$ has multiple roots τ_k , then we have:

$$\int_{-\infty}^{\infty} x(\tau) \delta(f(\tau)) d\tau = \sum_k \frac{x(\tau_k)}{|f'(\tau_k)|} \quad (1.26)$$

This is the generalized sifting property of the impulse function. We can now express the delta function as:

$$\delta(f(t)) = \sum_k \frac{\delta(t - t_k)}{|f'(\tau_k)|} \quad (1.27)$$

which is composed of a set of impulses each corresponding to one of the roots of $f(t)$, weighted by the reciprocal of the absolute value of the derivative of the function evaluated at the root.

1.3 Relationship Between Complex Exponentials and Delta Functions

Here we list a set of six important formulas that will be used in the discussions of various forms of the Fourier transform in Chapters 3 and 4. These formulas show that the Kronecker and Dirac delta functions can be generated as the sum or integral of some forms of the general complex exponential function $e^{j2\pi ft} = e^{j\omega t}$. The proofs of these formulas are left as homework problems.

- I. Dirac delta as an integral of a complex exponential:

$$\begin{aligned} \int_{-\infty}^{\infty} e^{\pm j2\pi ft} dt &= \int_{-\infty}^{\infty} \cos(2\pi ft) dt \pm j \int_{-\infty}^{\infty} \sin(2\pi ft) dt \\ &= 2 \int_0^{\infty} \cos(2\pi ft) dt = \delta(f) = 2\pi\delta(\omega) \end{aligned} \quad (1.28)$$

Note that the integral of the odd function $\sin(2\pi ft)$ over all time $-\infty < t < \infty$ is zero, while the integral of the even function $\cos(2\pi ft)$ over all time is twice the integral over $0 < t < \infty$. Eq.1.28 can also be interpreted intuitively. The integral of any sinusoid over all time is always zero, except if $f = 0$ and $e^{\pm j2\pi ft} = 1$, then the integral becomes infinity. Alternatively, if we integrate the complex exponential with respect to frequency f , we get:

$$\int_{-\infty}^{\infty} e^{\pm j2\pi ft} df = 2 \int_0^{\infty} \cos(2\pi ft) df = \delta(t) \quad (1.29)$$

which can also be interpreted intuitively as a superposition of uncountably infinite sinusoids with progressively higher frequency f . These sinusoids cancel each other at any time $t \neq 0$ except if $t = 0$ and $\cos(2\pi ft) = 1$ for all f then their superposition becomes infinity.

- Ia. This formula is associated with Eq.1.28:

$$\int_0^{\infty} e^{\pm j2\pi ft} dt = \int_0^{\infty} e^{\pm j\omega t} dt = \frac{1}{2} \delta(f) \mp \frac{1}{j2\pi f} = \pi\delta(\omega) \mp \frac{1}{j\omega} \quad (1.30)$$

Given the above, we can also get:

$$\begin{aligned} \int_{-\infty}^0 e^{\pm j\omega t} dt &= \int_0^{-\infty} e^{\pm j\omega t} d(-t) = \int_0^\infty e^{\mp j\omega t} dt \\ &= \frac{1}{2}\delta(f) \pm \frac{1}{j2\pi f} = \pi\delta(\omega) \pm \frac{1}{j\omega} \end{aligned} \quad (1.31)$$

Adding the two equations above we get the same result as given in Eq.1.28:

$$\int_{-\infty}^\infty e^{\pm j\omega t} dt = \int_{-\infty}^0 e^{\pm j\omega t} dt + \int_0^\infty e^{\pm j\omega t} dt = \delta(f) = 2\pi\delta(\omega) \quad (1.32)$$

- II. Kronecker delta as an integral of a complex exponential:

$$\begin{aligned} \frac{1}{T} \int_T e^{\pm j2\pi kt/T} dt &= \frac{1}{T} \int_T \cos(2\pi kt/T) dt \pm j \frac{1}{T} \int_T \sin(2\pi kt/T) dt \\ &= \frac{1}{T} \int_T \cos(2\pi kt/T) dt = \delta[k] \end{aligned} \quad (1.33)$$

In particular if $T = 1$ we have:

$$\int_0^1 e^{\pm j2\pi kt} dt = \delta[k] \quad (1.34)$$

- III. A train of Dirac deltas with period F as a summation of a complex exponential:

$$\begin{aligned} \frac{1}{F} \sum_{n=-\infty}^{\infty} e^{\pm j2\pi fn/F} &= \frac{1}{F} \sum_{n=-\infty}^{\infty} \cos(2\pi fn/F) \pm j \frac{1}{F} \sum_{n=-\infty}^{\infty} \sin(2\pi fn/F) \\ &= \frac{1}{F} \sum_{n=-\infty}^{\infty} \cos(2\pi fn/F) = \sum_{k=-\infty}^{\infty} \delta(f - kF) = \sum_{k=-\infty}^{\infty} 2\pi\delta(\omega - 2\pi kF) \end{aligned} \quad (1.35)$$

In particular if $F = 1$ we have:

$$\sum_{n=-\infty}^{\infty} e^{\pm j2\pi fn} = \sum_{k=-\infty}^{\infty} \delta(f - k) = \sum_{k=-\infty}^{\infty} 2\pi\delta(\omega - 2\pi k) \quad (1.36)$$

- IIIa. This formula is associated with Eq.1.36:

$$\sum_{n=0}^{\infty} e^{\pm j2\pi fn} = \frac{1}{2} \sum_{k=-\infty}^{\infty} \delta(f - k) + \frac{1}{1 - e^{\pm j2\pi f}} = \sum_{k=-\infty}^{\infty} \pi\delta(\omega - 2\pi k) + \frac{1}{1 - e^{\pm j\omega}} \quad (1.37)$$

Given the above, we can also get:

$$\begin{aligned} \sum_{n=-\infty}^{-1} e^{\pm j2\pi fn} &= \sum_{n=0}^{\infty} e^{\mp j2\pi fn} - 1 = \frac{1}{2} \sum_{k=-\infty}^{\infty} \delta(f - k) + \frac{1}{1 - e^{\mp j2\pi f}} - 1 \\ &= \frac{1}{2} \sum_{k=-\infty}^{\infty} \delta(f - k) - \frac{1}{1 - e^{\pm j2\pi f}} \end{aligned} \quad (1.38)$$

Adding the two equations above we get the same result as given in Eq.1.36:

$$\begin{aligned} \sum_{n=-\infty}^{\infty} e^{\pm j2\pi f n} &= \sum_{n=-\infty}^{-1} e^{\pm j2\pi f n} + \sum_{n=0}^{\infty} e^{\pm j2\pi f n} \\ &= \sum_{k=-\infty}^{\infty} \delta(f - k) = 2\pi \sum_{k=-\infty}^{\infty} \delta(\omega - 2\pi k) \end{aligned} \quad (1.39)$$

- IV. A train of Kronecker deltas with period N as a summation of complex exponential:

$$\begin{aligned} \frac{1}{N} \sum_{n=0}^{N-1} e^{\pm j2\pi nm/N} &= \frac{1}{N} \sum_{n=0}^{N-1} \cos(2\pi nm/N) \pm \frac{j}{N} \sum_{n=0}^{N-1} \sin(2\pi nm/N) \\ &= \frac{1}{N} \sum_{n=0}^{N-1} \cos(2\pi nm/N) = \sum_{k=-\infty}^{\infty} \delta[m - kN] \end{aligned} \quad (1.40)$$

1.4 Attributes of Signals

A time signal can be characterized by the following parameters:

- The *Energy* contained in a continuous signal $x(t)$ is:

$$\mathcal{E} = \int_{-\infty}^{\infty} |x(t)|^2 dt \quad (1.41)$$

or in a discrete signal $x[n]$:

$$\mathcal{E} = \sum_{n=-\infty}^{\infty} |x[n]|^2 \quad (1.42)$$

Note that $|x(t)|^2$ and $|x[n]|^2$ have different dimensions and they represent respectively the power and energy of the signal at the corresponding moment. If the energy contained in a signal is finite $\mathcal{E} < \infty$, then the signal is called an *energy signal*. A continuous energy signal is said to be *square-integrable*, and a discrete energy signal is said to be *square-summable*. All signals to be considered in the future, either continuous or discrete, will be assumed to be energy signals.

- The *average power* of the signal:

$$\mathcal{P} = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T |x(t)|^2 dt \quad (1.43)$$

or for a discrete signal:

$$\mathcal{P} = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=1}^N |x[n]|^2 \quad (1.44)$$

If \mathcal{E} of $x(t)$ is not finite but \mathcal{P} is, $x(t)$ is a *power signal*. Obviously the average power of an energy signal is zero.

- The *cross-correlation* defined below measures the similarity between two signals as a function of the relative time shift:

$$\begin{aligned} r_{xy}(\tau) &= x(t) \star y(t) = \int_{-\infty}^{\infty} x(t) \bar{y}(t - \tau) dt = \int_{-\infty}^{\infty} x(t + \tau) \bar{y}(t) dt \\ &\neq \int_{-\infty}^{\infty} \bar{x}(t - \tau) y(t) dt = y(t) \star x(t) = r_{yx}(\tau) \end{aligned} \quad (1.45)$$

Note that the cross-correlation is not commutative. For discrete signal, we have

$$r_{xy}[m] = x[n] \star y[n] = \sum_{n=-\infty}^{\infty} x[n] \bar{y}[n-m] = \sum_{n=-\infty}^{\infty} x[n+m] \bar{y}[n] \quad (1.46)$$

In particular, when $x(t) = y(t)$ and $x[n] = y[n]$, the cross-correlation becomes the *autocorrelation* which measures the self-similarity of the signal:

$$r_x(\tau) = \int_{-\infty}^{\infty} x(t) \bar{x}(t - \tau) dt = \int_{-\infty}^{\infty} x(t + \tau) \bar{x}(t) dt \quad (1.47)$$

and

$$r_x[m] = \sum_{n=-\infty}^{\infty} x[n] \bar{x}[n-m] = \sum_{n=-\infty}^{\infty} x[n+m] \bar{x}[n] \quad (1.48)$$

More particularly when $\tau = 0$ and $m = 0$ we have:

$$r_x(0) = \int_{-\infty}^{\infty} |x(t)|^2 dt, \quad r_x[0] = \sum_{n=-\infty}^{\infty} |x[n]|^2 \quad (1.49)$$

which represent the total energy contained in the signal.

- A random time signal $x(t)$ is called a *stochastic process*. Its mean or expectation (Appendix B) is:

$$\mu_x(t) = E[x(t)] \quad (1.50)$$

The cross-covariance of two stochastic processes $x(t)$ and $y(t)$ is:

$$\begin{aligned} Cov_{xy}(t, \tau) &= \sigma_{xy}^2(t, \tau) = E[(x(t) - \mu_x(t))(y(t) - \mu_y(t))] \\ &= E[x(t)\bar{y}(\tau)] - \mu_x(t)\bar{\mu}_y(\tau) \end{aligned} \quad (1.51)$$

- A stochastic process $x(t)$ can be truncated and sampled to become a random vector $\mathbf{x} = [x[1], \dots, x[N]]^T$. The mean or expectation of \mathbf{x} is a vector:

$$\boldsymbol{\mu}_x = E[\mathbf{x}] \quad (1.52)$$

with the n th element of $\boldsymbol{\mu}$ being $\mu[n] = E[x[n]]$. The cross-covariance of \mathbf{x} and \mathbf{y} is an N by N matrix:

$$\boldsymbol{\Sigma}_{xy} = E[(\mathbf{x} - \boldsymbol{\mu}_x)(\mathbf{y} - \boldsymbol{\mu}_y)^*] = E[\mathbf{x}\mathbf{y}^*] - \boldsymbol{\mu}_x\boldsymbol{\mu}_y^* \quad (1.53)$$

with the mn-th element being:

$$\sigma_{xy}^2[m, n] = E[(x[m] - \mu_x[m])(\bar{y}[n] - \bar{\mu}_y[n])] = E[x[m]\bar{y}[n]] - \mu_x[m]\bar{\mu}_y[n] \quad (1.54)$$

In particular, when $x(t) = y(t)$ and $x[n] = y[n]$, the cross-covariance becomes autocovariance:

$$\begin{aligned} Cov_x(t, \tau) &= \sigma_x^2(t, \tau) = E[(x(t) - \mu_x(t))(\bar{x}(\tau) - \bar{\mu}_x(\tau))] \\ &= E[x(t)\bar{x}(\tau)] - \mu_x(t)\bar{\mu}_x(\tau) \end{aligned} \quad (1.55)$$

and

$$\Sigma_x = E[(\mathbf{x} - \boldsymbol{\mu}_x)(\mathbf{x} - \boldsymbol{\mu}_x)^*] = E[\mathbf{x}\mathbf{x}^*] - \boldsymbol{\mu}_x\boldsymbol{\mu}_x^* \quad (1.56)$$

More particularly, when $t = \tau$ and $m = n$ we have:

$$\sigma_x^2(t) = E[|x(t)|^2] - |\mu_x(t)|^2, \quad \sigma_x^2[n] = E[|x[n]|^2] - |\mu_x[n]|^2 \quad (1.57)$$

We see that $\sigma_x^2(t)$ represents the average dynamic power of the signal $x(t)$, and $\sigma_x^2[n]$ represents the average dynamic energy contained in the nth signal component $x[n]$.

1.5 Signal Arithmetics and Transformations

Any of the arithmetic operations (addition/subtraction and multiplication/division) can be applied to two continuous signal $x(t)$ and $y(t)$, or two discrete signals $x[n]$ and $y[n]$ to produce a new signal $z(t)$ or $z[n]$:

- Scaling: $z(t) = ax(t)$ or $z[n] = ax[n]$
- Addition/subtraction: $z(t) = x(t) \pm y(t)$ or $z[n] = x[n] \pm y[n]$;
- Multiplication: $z(t) = x(t)y(t)$ or $z[n] = x[n]y[n]$
- Division: $z(t) = x(t)/y(t)$ or $z[n] = x[n]/y[n]$

Note that these operations are actually applied to the amplitude values of the two signals $x(t)$ and $y(t)$ at each moment t , and the result becomes the value of $z(t)$ at the same moment, and the same is true for the operations on the discrete signals.

Moreover, a linear transformation in the general form of $y = ax + b = a(x + b/a)$ can be applied to the amplitude of a function $x(t)$ (vertical in time plot) in two steps:

- Translation:
 $y(t) = x(t) + x_0$: the time function $x(t)$ is moved either upward if $x_0 > 0$ or downward if $x_0 < 0$.
- Scaling:
 $y(t) = ax(t)$: the time function $x(t)$ is either up-scaled if $|a| > 1$ or down-scaled if $|a| < 1$. $x(t)$ is also flipped vertically (upside-down) if $a < 0$.

The same linear transformation $y = ax + b$ can also be applied to the time argument t of the function $x(t)$ (horizontal in time plot) as well as to its amplitude:

$$\tau = at + t_0 = a(t + t_0/a), \quad y(\tau) = x(at + t_0) = x(a(t + t_0/a)) \quad (1.58)$$

- Translation or shift:

$y(t) = x(t + t_0)$ is translated by $|t_0|$ either to the right if $t_0 < 0$, or to the left if $t_0 > 0$.

- Scaling:

$y(t) = x(at)$ is either compressed if $|a| > 1$, or expanded if $|a| < 1$. The signal is also reversed (flipped horizontally) in time if $a < 0$.

In general, the transformation in time $y(t) = x(at + t_0) = x(a(t + t_0/a))$ containing both translation and scaling can be carried out in either of the two methods.

1. A two-step process:

- Step 1: define an intermediate signal $z(t) = x(t + t_0)$ due to translation;
- Step 2: find the transformed signal $y(t) = z(at)$ due to time-scaling (containing time reversal if $a < 0$);

The two steps can be carried out equivalently in reverse order:

- Step 1: define an intermediate signal $z(t) = x(at)$ due to time-scaling (containing time reversal if $a < 0$);
 - Step 2: find the transformed signal $y(t) = z(t + t_0/a)$ due to translation;
- However, note that the translation parameters (direction and amount) are different depending on whether it is carried before or after scaling.

2. A two-point process:

Evaluate $x(t)$ at two arbitrarily chosen time points $t = t_1$ and $t = t_2$ to get $v_1 = x(t_1)$ and $v_2 = x(t_2)$. Then $y(t) = x(at + t_0) = v_1$ when its argument is $at + t_0 = t_1$, i.e., when $t = (t_1 - t_0)/a$, and $y(t) = x(at + t_0) = v_2$ when $at + t_0 = t_2$, i.e., $t = (t_2 - t_0)/a$. As the transformation $at + t_0$ is linear, the value of $y(t)$ at any other time moment t can be found by linear interpolation based on these two points.

Example 1.1: Consider the transformation of a time signal:

$$x(t) = \begin{cases} t & 0 < t < 2 \\ 0 & \text{else} \end{cases} \quad (1.59)$$

- Translation: $y(t) = x(t + 3)$ and $z(t) = x(t - 1)$ are shown in Fig.1.4(a).
- Expansion/compression: $y(t) = x(2t/3)$ and $z(t) = x(2t)$ are shown in Fig.1.4(b).
- Time reversal: $y(t) = x(-t)$ and $z(t) = x(-2t)$ are shown in Fig.1.4c.
- Combination of translation, scaling and reversal:

$$y(t) = x(-2t + 3) = x\left(-2\left(t - \frac{3}{2}\right)\right) \quad (1.60)$$

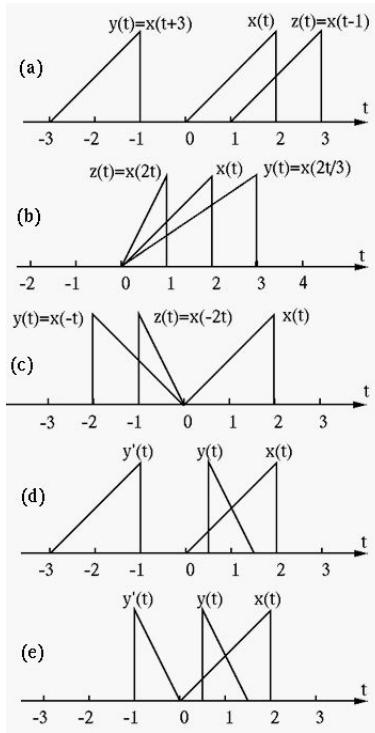


Figure 1.4 Transformation of continuous signal

- Method 1: based on the first expression $y(t) = x(-2t + 3)$ we get (Fig.1.4 (d)):

$$z(t) = x(t + 3), \quad y(t) = z(-2t) \quad (1.61)$$

Alternatively, based on the second expression $y(t) = x(-2(t - 3/2))$ we get (Fig.1.4 (e)):

$$z(t) = x(-2t), \quad y(t) = z(t - \frac{3}{2}) \quad (1.62)$$

- Method 2: the signal has two break points at $t_1 = 0$ and $t_2 = 2$, correspondingly, the two break points for $y(t)$ can be found to be:

$$\begin{aligned} -2t + 3 &= t_1 = 0 \implies t = \frac{3}{2} \\ -2t + 3 &= t_2 = 2 \implies t = \frac{1}{2} \end{aligned}$$

By linear interpolation based on these two points, the entire signal $y(t)$ can be easily obtained, same as that obtained by the previous method shown in Fig.1.4(d) and (e).

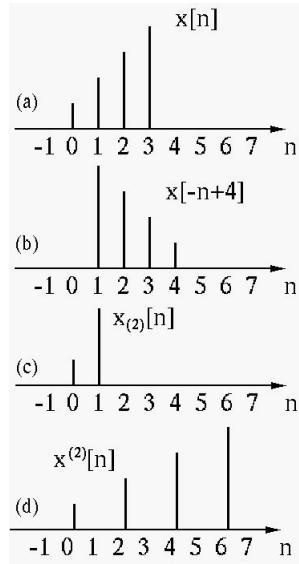


Figure 1.5 Transformation of discrete signal

In the transformation of discrete signals, the expansion and compression for continuous signals are replaced respectively by *up-sampling* and *down-sampling*.

- Down-sampling (decimation):

Keep every N th sample and discard the rest. Signal size becomes $1/N$ of the original one.

$$x_{(N)}[n] = x[nN] \quad (1.63)$$

For example, if $N = 3$, $x_{(3)}[0] = x[0]$, $x_{(3)}[1] = x[3]$, $x_{(3)}[2] = x[6]$, ...

- Up-sampling (interpolation by zero stuffing):

Insert $N - 1$ zeros between every two consecutive samples $x[n]$ and $x[n + 1]$. Signal size becomes N times the original one.

$$x^{(N)}[n] = \begin{cases} x[n/N] & n = 0, \pm N, \pm 2N, \dots \\ 0 & \text{else} \end{cases} \quad (1.64)$$

For example, if $N = 2$, $x^{(2)}[0] = x[0]$, $x^{(2)}[2] = x[1]$, $x^{(2)}[4] = x[2]$, ..., and $x[n] = 0$ for all other n .

Example 1.2: Given $x[n]$ as shown in Fig.1.5(a), a transformation $y[n] = x[-n + 4]$, shown in Fig.1.5(b), can be obtained based on two time points:

$$\begin{aligned} -n + 4 &= 0 \implies n = 4 \\ -n + 4 &= 3 \implies n = 1 \end{aligned} \quad (1.65)$$

The up and down sampling of the signal in Fig.1.5(a) can be obtained in the following table and shown in Fig.1.5(c) and (d), respectively.

n	...	-1	0	1	2	3	4	5	6	7	...
$x[n]$...	0	1	2	3	4	0	0	0	0	...
$x_{(2)}[n]$...	0	1	3	0	0	0	0	0	0	...
$x^{(2)}[n]$...	0	1	0	2	0	3	0	4	0	...

1.6 Linear and Time Invariant Systems

A generic system (electrical, mechanical, biological, economical, etc.) can be symbolically represented in terms of the relationship between its input $x(t)$ (stimulus, excitation) and output $y(t)$ (response, reaction):

$$\mathcal{O}[x(t)] = y(t) \quad (1.67)$$

where the symbol $\mathcal{O}[\]$ represents the operation applied by the system to its input. A system is *linear* if its input-output relationship satisfies both *homogeneity* and *superposition*.

- Homogeneity:

$$\mathcal{O}[ax(t)] = a\mathcal{O}[x(t)] = ay(t) \quad (1.68)$$

- Superposition: If $\mathcal{O}[x_n(t)] = y_n(t)$ ($n = 1, 2, \dots, N$), then:

$$\mathcal{O}\left[\sum_{n=1}^N x_n(t)\right] = \sum_{n=1}^N \mathcal{O}[x_n(t)] = \sum_{n=1}^N y_n(t) \quad (1.69)$$

or

$$\mathcal{O}\left[\int_{-\infty}^{\infty} x(\tau)d\tau\right] = \int_{-\infty}^{\infty} \mathcal{O}[x(\tau)]d\tau = \int_{-\infty}^{\infty} y(\tau)d\tau \quad (1.70)$$

Combining these two properties, we have

$$\mathcal{O}\left[\sum_{n=1}^N a_n x_n(t)\right] = \sum_{n=1}^N a_n \mathcal{O}[x_n(t)] = \sum_{n=1}^N a_n y_n(t) \quad (1.71)$$

or

$$\mathcal{O}\left[\int_{-\infty}^{\infty} a(\tau)x(\tau)d\tau\right] = \int_{-\infty}^{\infty} a(\tau)\mathcal{O}[x(\tau)]d\tau = \int_{-\infty}^{\infty} a(\tau)y(\tau)d\tau \quad (1.72)$$

A system is *time-invariant* if how it responds to the input does not change over time. In other words:

$$\text{if } \mathcal{O}[x(t)] = y(t), \text{ then } \mathcal{O}[x(t - \tau)] = y(t - \tau) \quad (1.73)$$

A *linear and time-invariant (LTI)* system is both linear and time-invariant.

As an example, we see that the response of an LTI system $y(t) = \mathcal{O}[x(t)]$ to $dx(t)/dt$ is $dy(t)/dt$:

$$\mathcal{O}\left[\frac{1}{\Delta}[x(t + \Delta) - x(t)]\right] = \frac{1}{\Delta}[y(t + \Delta) - y(t)] \quad (1.74)$$

Taking the limit $\Delta \rightarrow 0$, we get:

$$\mathcal{O}\left[\frac{d}{dt}x(t)\right] = \mathcal{O}[\dot{x}(t)] = \frac{d}{dt}y(t) = \dot{y}(t) \quad (1.75)$$

Example 1.3: Determine if each of the following systems is linear.

- The input $x(t)$ is the voltage across a resistor R and the output $y(t)$ is the current through R :

$$y(t) = \mathcal{O}[x(t)] = \frac{x(t)}{R} \quad (1.76)$$

This is obviously a linear system.

- The input $x(t)$ is the voltage across a resistor R and the output $y(t)$ is the power consumed by R :

$$y(t) = \mathcal{O}[x(t)] = \frac{x^2(t)}{R} \quad (1.77)$$

This is not a linear system.

- The input $x(t)$ is the voltage across a resistor R and a capacitor C in series and the output is the voltage across C :

$$RC\frac{d}{dt}y(t) + y(t) = \tau\frac{d}{dt}y(t) + y(t) = x(t) \quad (1.78)$$

where $\tau = RC$ is the *time constant* of the system. As the system is characterized by a linear, first order ordinary differential equation (ODE), it is linear.

- A system produces its output $y(t)$ by adding a constant a to its input $x(t)$:

$$y(t) = \mathcal{O}[x(t)] = x(t) + a \quad (1.79)$$

Consider:

$$\begin{aligned} \mathcal{O}[x_1(t) + x_2(t)] &= x_1(t) + x_2(t) + a \\ &\neq \mathcal{O}[x_1(t)] + \mathcal{O}[x_2(t)] = x_1(t) + x_2(t) + 2a \end{aligned} \quad (1.80)$$

This is not a linear system.

- The input $x(t)$ is the force f applied to a spring of length l_0 and spring constant k , the output is the length of the spring. According to Hooke's law, $\Delta l = -kf = -kx(t)$, we have

$$y(t) = l = l_0 + \Delta l = l_0 - kx(t) \quad (1.81)$$

This is not a linear system.

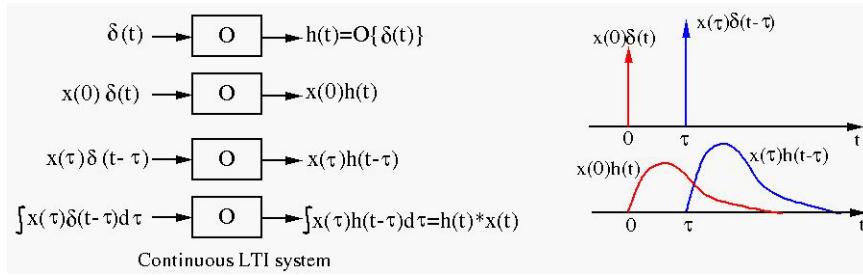


Figure 1.6 Response of a continuous LTI system

- Same as above except the output $y(t) = l - l_0 = \Delta l$ is the displacement of the moving end of the spring:

$$y(t) = \Delta l = -kf = -kx(t) \quad (1.82)$$

This system is linear.

1.7 Signals Through LTI Systems (Continuous)

If the input to an LTI system is an impulse $x(t) = \delta(t)$ at $t = 0$, then the response of the system, called the *impulse response function*, is

$$h(t) = \mathcal{O}[\delta(t)] \quad (1.83)$$

We now show that given the impulse response $h(t)$ of an LTI system, we can find its response to *any* input $x(t)$. First, according to Eq. 1.9, we can express the input as

$$x(t) = \int_{-\infty}^{\infty} x(\tau)\delta(t-\tau)d\tau \quad (1.84)$$

Next, according to Eqs. 1.70 and 1.73, we have

$$\begin{aligned} y(t) &= \mathcal{O}[x(t)] = \mathcal{O}\left[\int_{-\infty}^{\infty} x(\tau)\delta(t-\tau)d\tau\right] \\ &= \int_{-\infty}^{\infty} x(\tau)\mathcal{O}[\delta(t-\tau)]d\tau = \int_{-\infty}^{\infty} x(\tau)h(t-\tau)d\tau \end{aligned} \quad (1.85)$$

This process is illustrated in Fig. 1.6. The integration on the right hand side above is called the *continuous convolution*, which is generally defined as an operation of two continuous functions $x(t)$ and $y(t)$:

$$z(t) = x(t) * y(t) = \int_{-\infty}^{\infty} x(\tau)y(t-\tau)d\tau = \int_{-\infty}^{\infty} y(\tau)x(t-\tau)d\tau = y(t) * x(t) \quad (1.86)$$

Note that convolution is commutative, i.e., $x(t) * y(t) = y(t) * x(t)$.

In particular, if the input to an LTI system is a complex exponential function:

$$x(t) = e^{st} = e^{(\sigma+j\omega)t} = [\cos(\omega t) + j \sin(\omega t)]e^{\sigma t} \quad (1.87)$$

where $s = \sigma + j\omega$ is a complex parameter, the corresponding output is

$$y(t) = \mathcal{O}[e^{st}] = \int_{-\infty}^{\infty} h(\tau)e^{s(t-\tau)}d\tau = e^{st} \int_{-\infty}^{\infty} h(\tau)e^{-s\tau}d\tau = H(s)e^{st} \quad (1.88)$$

where $H(s)$ is a constant (independent of the time variable t) defined as

$$H(s) = \int_{-\infty}^{\infty} h(\tau)e^{-s\tau}d\tau \quad (1.89)$$

This is called the *transfer function (TF)* of the continuous LTI system, which is the *Laplace transform* of the impulse response function $h(t)$ of the system, to be discussed in Chapter 6. We can rewrite Eq.1.88 as an *eigenequation*:

$$\mathcal{O}[e^{st}] = H(s)e^{st} \quad (1.90)$$

where the constant $H(s)$ and the complex exponential e^{st} are, respectively, the *eigenvalue* and the corresponding *eigenfunction* of the LTI system, i.e., the response of the system to the complex exponential input e^{st} is equal to the input multiplied by a constant $H(s)$. Also note that the complex exponential e^{st} is the eigenfunction of *any* continuous LTI system, independent of its specific impulse response $h(t)$.

In particular, when $s = j\omega = j2\pi f$ ($\sigma = 0$), $H(s)$ becomes:

$$H(j\omega) = \int_{-\infty}^{\infty} h(\tau)e^{-j2\pi f\tau}d\tau = \int_{-\infty}^{\infty} h(\tau)e^{-j\omega\tau}d\tau \quad (1.91)$$

This is the *frequency response function (FRF)* of the system, which is the *Fourier transform* of the impulse response function $h(t)$, to be discussed in Chapter 3. Different notations such as $H(f)$ and (ω) can also be used for the FRF as a function of frequency f or $\omega = 2\pi f$ in various literatures, depending on the convention adopted by the authors. We may use any of these depending on the context and convention adopted by the authors.

Given the FRF $H(j\omega)$ of a system, its response to an input $x(t) = e^{j\omega_0 t}$ with a specific frequency $\omega_0 = 2\pi f_0$ can be found by evaluating Eq.1.88 at $s = j\omega_0$:

$$y(t) = \mathcal{O}[e^{j\omega_0 t}] = H(j\omega_0)e^{j\omega_0 t} \quad (1.92)$$

Moreover, if the input $x(t)$ can be written as a linear combination of a set of complex exponentials:

$$x(t) = \sum_{k=-\infty}^{\infty} X[k]e^{jk\omega_0 t} \quad (1.93)$$

where $X[k]$ is the weighting coefficient for the k th complex exponential of frequency $k\omega_0$, then, due to the linearity of the system, its output is:

$$\begin{aligned} y(t) &= \mathcal{O}[x(t)] = \mathcal{O}\left[\sum_{k=-\infty}^{\infty} X[k]e^{jk\omega_0 t}\right] = \sum_{k=-\infty}^{\infty} X[k]\mathcal{O}[e^{jk\omega_0 t}] \\ &= \sum_{k=-\infty}^{\infty} X[k]H(jk\omega_0)e^{jk\omega_0 t} = \sum_{k=-\infty}^{\infty} Y[k]e^{jk\omega_0 t} \end{aligned} \quad (1.94)$$

where $Y[k] = X[k]H(jk\omega_0)$ is the k th coefficient for the output. The result can be generalized to cover signals composed of infinite uncountable complex exponentials:

$$x(t) = \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df \quad (1.95)$$

where $X(f)$ is the weighting function for all exponentials with frequency in the range of $-\infty < f < \infty$, then its output is:

$$\begin{aligned} y(t) &= \mathcal{O}[x(t)] = \mathcal{O}\left[\int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df\right] = \int_{-\infty}^{\infty} X(f)\mathcal{O}[e^{j2\pi ft}]df \\ &= \int_{-\infty}^{\infty} X(f)H(f)e^{j2\pi ft}df = \int_{-\infty}^{\infty} Y(f)e^{j2\pi ft}df \end{aligned} \quad (1.96)$$

where $Y(f) = X(f)H(f)$ is the weighting function for the output.

The results above are of great significance, as they indicate that we can obtain the response of an LTI system described by its transfer function $H(s)$ or equivalently its impulse response function $h(t)$ to *any* input $x(t)$ in the form of a linear combination of a set of complex exponentials. This is also an important conclusion of the Fourier transform theory to be considered in Chapter 3.

An LTI system is *stable* if its response to any bounded input is also bounded:

$$\text{if } |x(t)| < B_x \text{ then } |y(t)| < B_y \quad (1.97)$$

As the input and output of an LTI are related by convolution

$$y(t) = h(t) * x(t) = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau \quad (1.98)$$

we have:

$$\begin{aligned} |y(t)| &= \left| \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau \right| \leq \int_{-\infty}^{\infty} |h(\tau)||x(t-\tau)|d\tau \\ &< B_x \int_{-\infty}^{\infty} |h(\tau)|d\tau < B_y \end{aligned} \quad (1.99)$$

which obviously requires:

$$\int_{-\infty}^{\infty} |h(\tau)|d\tau < \infty \quad (1.100)$$

In other words, if the impulse response function $h(t)$ of an LTI system is absolutely integrable, then the system is stable, i.e., Eq.1.100 is the sufficient con-

dition for an LTI system to be stable. We can show that this condition is also necessary, i.e., all stable LTI systems' impulse response functions are absolutely integrable.

An LTI system is *causal* if its output $y(t)$ only depends on the current and past input $x(t)$ (but not the future). If the system is initially at rest with zero output $y(t) = 0$ for $t < 0$, then its response $y(t) = h(t)$ to an impulse $x(t) = \delta(t)$ at moment $t = 0$ will be at rest before the moment $t = 0$, i.e., $h(t) = h(t)u(t)$. Its response to a general input $x(t)$ is:

$$y(t) = h(t) * x(t) = \int_{-\infty}^{\infty} h(\tau)x(t - \tau)d\tau = \int_0^{\infty} h(\tau)x(t - \tau)d\tau \quad (1.101)$$

Moreover, if the input begins at a specific moment, e.g., $t = 0$, i.e., $x(t) = x(t)u(t)$ and $x(t - \tau) = 0$ for $\tau > t$, then we have

$$y(t) = h(t) * x(t) = \int_{-\infty}^{\infty} h(\tau)x(t - \tau)d\tau = \int_0^t h(\tau)x(t - \tau)d\tau \quad (1.102)$$

1.8 Signals Through LTI Systems (Discrete)

Similar to the above discussion for continuous signals and systems, the following results can be obtained for discrete signals and systems. First, as shown in Eq.1.3, any discrete signal can be written as:

$$x[n] = \sum_{m=-\infty}^{\infty} x[m]\delta[n - m] \quad (1.103)$$

Let the impulse response of a discrete LTI system be

$$h[n] = \mathcal{O}[\delta[n]] \quad (1.104)$$

then its response to the signal $x[n]$ is:

$$\begin{aligned} y[n] &= \mathcal{O}[x[n]] = \mathcal{O}\left[\sum_{m=-\infty}^{\infty} x[m]\delta[n - m]\right] = \sum_{m=-\infty}^{\infty} x[m]\mathcal{O}[\delta[n - m]] \\ &= \sum_{m=-\infty}^{\infty} x[m]h[n - m] = \sum_{m=-\infty}^{\infty} x[n - m]h[m] \end{aligned} \quad (1.105)$$

This process is illustrated in Fig.1.7.

The last summation in Eq.1.105 is defined called the *discrete convolution*, which is generally defined as an operation of two discrete functions $x[n]$ and $h[n]$:

$$z[n] = x[n] * y[n] = \sum_{m=-\infty}^{\infty} x[m]y[n - m] = \sum_{m=-\infty}^{\infty} y[m]x[n - m] = y[n] * x[n] \quad (1.106)$$

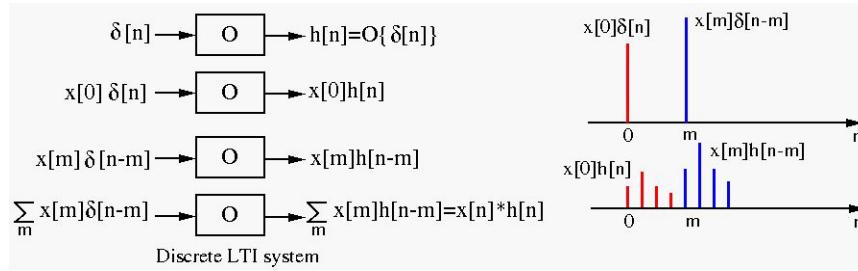


Figure 1.7 Response of a discrete LTI system

Note that convolution is commutative, i.e., $x[n] * y[n] = y[n] * x[n]$. Similar to the continuous case, if the system is causal and the input $x[n]$ is zero until $n = 0$, we have:

$$y[n] = \sum_{m=0}^n x[m]h[n-m] = \sum_{m=0}^n x[n-m]h[m] \quad (1.107)$$

In particular, if the input to an LTI system is a complex exponential function:

$$x[n] = e^{sn} = (e^s)^n = z^n \quad (1.108)$$

where $s = \sigma + j\omega$ as defined above, and z is defined as $z = e^s$, then according to Eq.1.105, the corresponding output is

$$y[n] = \mathcal{O}[z^n] = \sum_{k=-\infty}^{\infty} h[k]z^{n-k} = z^n \sum_{k=-\infty}^{\infty} h[k]z^{-k} = H(z)z^n \quad (1.109)$$

where $H(z)$ is a constant (independent of the time variable n) defined as

$$H(z) = \sum_{k=-\infty}^{\infty} h[k]z^{-k} \quad (1.110)$$

This is called the *transfer function (TF)* of the discrete LTI system, which is the *Z-transform* of the impulse response $h[n]$ of the system, to be discussed in Chapter 6. We note that Eq.1.109 is an eigenequation, where the constant $H(z)$ and the complex exponential z^n are, respectively, the eigenvalue and the corresponding eigenfunction of the LTI system. Also note that the complex exponential z^n is the eigenfunction of *any* discrete LTI system, independent of its specific impulse response $h[n]$. In particular, when $s = j\omega$ ($\sigma = 0$) and $z = e^s = e^{j\omega}$, $H(z)$ becomes:

$$H(e^{j\omega}) = \sum_{k=-\infty}^{\infty} h[k]e^{-j2k\pi f} = \sum_{k=-\infty}^{\infty} h[k]e^{-jkw} \quad (1.111)$$

This is the *frequency response function (FRF)* of the system, which is the Fourier transform of the discrete impulse response function $h[n]$, to be discussed in Chapter 4. As in the continuous case, different notations such as $H(f)$ and (ω) can

also be used for the FRF as a function of frequency f or $\omega = 2\pi f$ in various literatures, depending on the convention adopted by the authors.

Given $H(e^{j\omega})$ of a discrete system, its response to a discrete input $x[n] = z^n = e^{j\omega_0 n}$ with a specific frequency $\omega_0 = 2\pi f_0$ can be found to by evaluating Eq.1.109 at $z = e^{j\omega_0}$:

$$y[n] = \mathcal{O}[e^{j\omega_0 n}] = H(e^{j\omega_0})e^{j\omega_0 n} \quad (1.112)$$

Moreover, if the input $x[n]$ can be written as a linear combination of a set of complex exponentials:

$$x[n] = \sum_{k=0}^{N-1} X[k]e^{jk\omega_0 n/N} \quad (1.113)$$

where $X[k]$ ($0 \leq k < N$) are a set of constant coefficients, then, due to the linearity of the system, its output is:

$$\begin{aligned} y[n] &= \mathcal{O}[x[n]] = \mathcal{O}\left[\sum_{k=0}^{N-1} X[k]e^{jk\omega_0 n/N}\right] = \sum_{k=0}^{N-1} X[k]\mathcal{O}[e^{jk\omega_0 n}] \\ &= \sum_{k=0}^{N-1} X[k]H(e^{jk\omega_0})e^{jk\omega_0 n} = \sum_{k=0}^{N-1} Y[k]e^{jk\omega_0 n} \end{aligned} \quad (1.114)$$

where $Y[k] = X[k]H(e^{jk\omega_0})$ is the kth coefficient of the output. The result can be generalized to cover signals composed of uncountably infinite complex exponentials:

$$x[n] = \int_0^F X(f)e^{j2\pi fn/F} df \quad (1.115)$$

where $X(f)$ is the weighting function for all exponentials with frequencies in the range of $0 < f < F$, then its output is:

$$\begin{aligned} y[n] &= \mathcal{O}[x[n]] = \mathcal{O}\left[\int_0^F X(f)e^{j2\pi fn/F} df\right] = \int_0^F X(f)\mathcal{O}[e^{j2\pi fn/F}] df \\ &= \int_0^F X(f)H(e^{j2\pi fn/F})e^{j2\pi fn/F} df = \int_0^F Y(f)e^{j2\pi fn/F} df \end{aligned} \quad (1.116)$$

where $Y(f) = X(f)H(e^{j2\pi fn/F})$ is the weighting function for the output.

The significance of this result is that we can obtain the response of a discrete LTI system described by $H(z)$ or equivalently $h[k]$ to *any* input $x[n]$ in the form of a linear combination of a set of complex exponentials. This is an important conclusion of the discrete-time Fourier transform theory to be considered in Chapter 4.

Similar to a stable continuous LTI system, a stable discrete LTI system's response to any bounded input is also bounded for all n :

$$\text{if } |x[n]| < B_x \text{ then } |y[n]| < B_y \quad (1.117)$$

As the output and input of an LTI is related by convolution

$$y[n] = h[n] * x[n] = \sum_{m=-\infty}^{\infty} h[m]x[n-m] \quad (1.118)$$

we have:

$$\begin{aligned} |y[n]| &= \left| \sum_{m=-\infty}^{\infty} h[m]x[n-m] \right| \leq \sum_{m=-\infty}^{\infty} |h[m]| |x[n-m]| \\ &< B_x \sum_{m=-\infty}^{\infty} |h[m]| d\tau < B_y \end{aligned} \quad (1.119)$$

which obviously requires:

$$\sum_{m=-\infty}^{\infty} |h[m]| < \infty \quad (1.120)$$

In other words, if the impulse response function $h[n]$ of an LTI system is absolutely summable, then the system is stable, i.e., Eq.1.120 is the sufficient condition for an LTI system to be stable. We can show that this condition is also necessary, i.e., all stable LTI systems' impulse response functions are absolutely summable.

Also, a discrete LTI system is causal if its output $y[n]$ only depends on the current and past input $x[n]$ (but not the future). Assuming the system is initially at rest with zero output $y[n] = 0$ for $n < 0$, then its response $y[n] = h[n]$ to an impulse $x[n] = \delta[n]$ at moment $n = 0$ will be at rest before the moment $n = 0$, i.e., $h[n] = h[n]u[n]$. Its response to a general input $x[n]$ is:

$$y[n] = h[n] * x[n] = \sum_{m=-\infty}^{\infty} h[m]x[n-m] = \sum_{m=0}^{\infty} h[m]x[n-m] \quad (1.121)$$

Moreover, if the input begins at a specific moment, e.g., $n = 0$, i.e., $x[n] = x[n]u[n]$ and $x[n-m] = 0$ for $m > n$, then we have

$$y[n] = h[n] * x[n] = \sum_{m=-\infty}^{\infty} h[m]x[n-m] = \sum_{m=0}^n h[m]x[n-m] \quad (1.122)$$

1.9 Continuous and Discrete Convolutions

The continuous and discrete convolutions defined respectively in Eqs.1.86 and 1.106 are of great importance in the future discussions. Here we further consider how these convolutions can be specifically carried out. First we reconsider the continuous convolution

$$z(t) = x(t) * y(t) = \int_{-\infty}^{\infty} x(\tau)y(t-\tau)d\tau \quad (1.123)$$

which can be carried out conceptually in the following three steps:

1. Find the time reversal of one of the two functions, say, $y(\tau)$, by flipping it in time to get $y(-\tau)$;
2. Slide this flipped function along the τ axis to get $y(t - \tau)$ as the shift amount t goes from $-\infty$ to ∞ ;
3. For each shift amount t , find the integral of $x(\tau)y(t - \tau)$ over all τ , the area of overlap between $x(\tau)$ and $y(t - \tau)$, which is the convolution $z(t)$ at t .

This process is illustrated in the following example and in Fig.1.8.

Example 1.4: Let $x(t) = u(t)$ be the input to an LTI system with impulse response function $h(t) = e^{-at}u(t)$ (a first order system to be considered in Example 5.1), the output $y(t)$ of the system is:

$$\begin{aligned} y(t) &= h(t) * x(t) = \int_0^t h(t - \tau)d\tau = \int_0^t e^{-a(t-\tau)}d\tau \\ &= \frac{1}{a}e^{-at}e^{a\tau}\Big|_0^t = \frac{1}{a}e^{-at}(e^{at} - 1) = \frac{1}{a}(1 - e^{-at}), \quad (t > 0) \end{aligned} \quad (1.124)$$

The result can be written as $h(t) = \frac{1}{a}(1 - e^{-at})u(t)$ as it is zero when $t < 0$. Alternatively, the convolution can also be written as:

$$\begin{aligned} y(t) &= x(t) * h(t) = \int_{-\infty}^{\infty} h(\tau)x(t - \tau)d\tau = \int_0^t h(\tau)d\tau = \int_0^t e^{-a\tau}d\tau \\ &= -\frac{1}{a}e^{-a\tau}\Big|_0^t = \frac{1}{a}(1 - e^{-at})u(t) \end{aligned} \quad (1.125)$$

Moreover, if the input is

$$x(t) = u(t) - u(t - \tau) = \begin{cases} 1 & 0 \leq t < \tau \\ 0 & \text{else} \end{cases} \quad (1.126)$$

Due to the previous result and the linearity of the system, its output is:

$$\begin{aligned} y(t) &= h(t) * [u(t) - u(t - \tau)] = h(t) * u(t) - h(t) * u(t - \tau) \\ &= \frac{1}{a}[(1 - e^{-at})u(t) - (1 - e^{-a(t-\tau)})u(t - \tau)] \end{aligned} \quad (1.127)$$

This result is shown in Fig.1.9

Although convolution and cross-correlation (Eq.1.45) are two different operations, they look similar and are closely related. If we flip one of the two functions in a convolution, it becomes the same as the cross correlation.

$$x(t) * y(-t) = \int_{-\infty}^{\infty} x(\tau)y(-\tau - t)d\tau = r_{xy}(t) = x(t) \star y(t) \quad (1.128)$$

In other words, if one of the signals $y(t) = y(-t)$ is even, then we have $x(t) * y(t) = x(t) \star y(t)$.

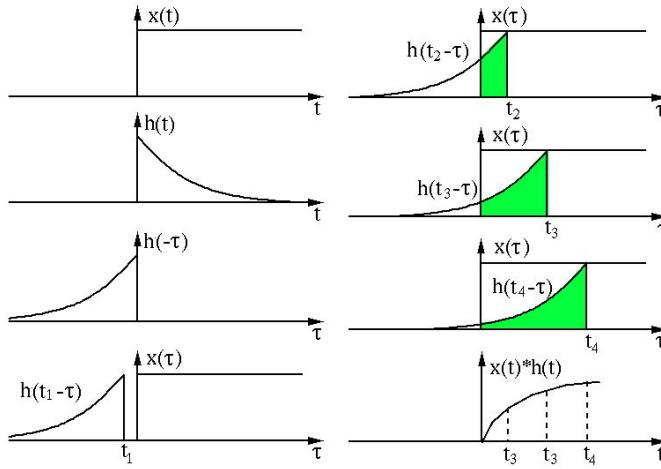


Figure 1.8 The convolution of two functions

The three steps are shown top-down, then left to right. The shaded area represents the convolution evaluated at a specific time moment such as $t = t_2$, $t = t_3$, and $t = t_4$.

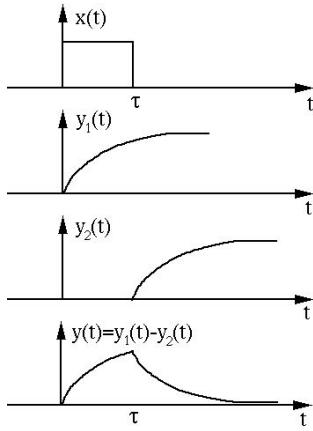


Figure 1.9 The linearity of convolution

Given $y_1(t) = h(t) * u(t)$ and $y_2(t) = h(t) * u(t - \tau)$, then $h(t) * [u(t) - u(t - \tau)] = y_1(t) - y_2(t)$.

Example 1.5: Let $x(t) = e^{-at}u(t)$ and $y(t) = e^{-bt}u(t)$, and both a and b are positive. We first find their convolution:

$$x(t) * y(t) = \int_{-\infty}^{\infty} x(\tau)y(t - \tau)d\tau \quad (1.129)$$

As $y(t - \tau)$ can be written as:

$$y(t - \tau) = e^{-b(t-\tau)} u(t - \tau) = \begin{cases} e^{-b(t-\tau)} & \tau < t \\ 0 & \tau > t \end{cases} \quad (1.130)$$

we have

$$\begin{aligned} x(t) * y(t) &= \int_0^t e^{-at} e^{-b(t-\tau)} d\tau = e^{-bt} \int_0^t e^{-(a-b)\tau} d\tau = \frac{1}{a-b} (e^{-bt} - e^{-at}) \\ &= \frac{1}{b-a} (e^{-at} - e^{-bt}) = y(t) * x(t) \end{aligned}$$

Next we find the cross-correlation $x(t) \star y(t)$:

$$x(t) \star y(t) = \int_{-\infty}^{\infty} x(\tau) y(\tau - t) d\tau \quad (1.131)$$

Consider two cases:

- When $t > 0$, the above becomes:

$$\int_t^{\infty} e^{-a\tau} e^{-b(\tau-t)} d\tau = e^{bt} \int_t^{\infty} e^{-(a+b)\tau} d\tau = \frac{e^{-at}}{a+b} u(t) \quad (1.132)$$

- When $t < 0$, the above becomes:

$$\int_0^{\infty} e^{-a\tau} e^{-b(\tau-t)} d\tau = e^{bt} \int_0^{\infty} e^{-(a+b)\tau} d\tau = \frac{e^{bt}}{a+b} u(-t) \quad (1.133)$$

Combining these two cases, we have:

$$x(t) \star y(t) = \frac{1}{a+b} \begin{cases} e^{-at} u(t) & t > 0 \\ e^{bt} u(-t) & t < 0 \end{cases} \quad (1.134)$$

Example 1.6: Let $x[n] = u[n]$ be the input to a discrete LTI system with impulse response $h[n] = a^n u[n]$ ($|a| < 1$), the output $y[n]$ is the following convolution (illustrated in Fig.1.10):

$$\begin{aligned} y[n] = h[n] * x[n] &= \sum_{m=-\infty}^{\infty} y[m] x[n-m] = \sum_{m=0}^n y[m] \\ &= \sum_{m=0}^n a^m = \frac{1 - a^{n+1}}{1 - a} \end{aligned} \quad (1.135)$$

Here we have used the geometric series formula:

$$\sum_{n=0}^N x^n = \frac{1 - x^{N+1}}{1 - x} \quad (1.136)$$

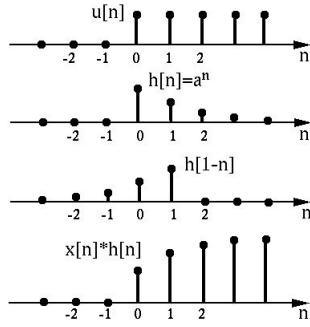


Figure 1.10 Discrete convolution

Alternatively, the convolution can also be written as:

$$\begin{aligned} x[n] * y[n] &= \sum_{m=-\infty}^{\infty} x[m]y[n-m] = \sum_{m=0}^n y[n-m] \\ &= a^n \sum_{m=0}^n a^{-m} = a^n \frac{1-a^{-(n+1)}}{1-a^{-1}} = \frac{1-a^{n+1}}{1-a} \end{aligned}$$

If $a = 1/2$, then the output $y[n]$ is $\dots, 0, 1, 3/2, 7/4, 15/8, \dots$, and when $n \rightarrow \infty$, $y[n] \rightarrow 1/(1-a) = 2$, as shown in the bottom panel of Fig.1.10.

1.10 Homework Problems

1. Given two square impulses as shown below:

$$r_a(t) = \begin{cases} 1 & |t| < a/2 \\ 0 & \text{else} \end{cases}, \quad r_b(t) = \begin{cases} 1 & |t| < b/2 \\ 0 & \text{else} \end{cases} \quad (1.137)$$

where we assume $b > a$, find their convolution $x(t) = r_a(t) * r_b(t)$ in analytical form (piecewise functions, i.e., one expression for one particular time interval) as well as graphic form.

2. Given a triangle wave, an isosceles triangle, as shown below:

$$s_a(t) = \begin{cases} 1 + t/a & -a < t < 0 \\ 1 - t/a & 0 < t < a \\ 0 & \text{else} \end{cases} \quad (1.138)$$

Find the convolution $s_a(t) * s_a(t)$ in analytical form (piecewise function) as well as graphic form.

3. Prove the identity in Eq.1.28:

$$\int_{-\infty}^{\infty} e^{\pm j2\pi f t} dt = \delta(f) \quad (1.139)$$

Hint: Follow these steps:

- a. Change the lower and upper integral limits to $-a/2$ and $a/2$, respectively, and show that this definite integral results in a sinc function $a \text{sinc}(af)$ of frequency f with a parameter a . A sinc function is defined as $\text{sinc}(x) = \sin(\pi x)/\pi x$, and $\lim_{x \rightarrow 0} \text{sinc}(x) = 1$.
- b. Show that the following integral of this sinc function $a \text{sinc}(af)$ is 1 (independent of a):

$$a \int_{-\infty}^{\infty} \text{sinc}(af) df = 1 \quad (1.140)$$

based on the integral formula:

$$\int_0^{\infty} \frac{\sin(x)}{x} dx = \frac{\pi}{2} \quad (1.141)$$

- c. Let $a \rightarrow \infty$ and show that $a \text{sinc}(af)$ approaches a unit impulse:

$$\lim_{a \rightarrow \infty} s(f, a) = \delta(f) \quad (1.142)$$

4. Prove the identity in Eq.1.30:

$$\int_0^{\infty} e^{\pm j2\pi f t} dt = \frac{1}{2} \delta(f) \mp \frac{1}{j2\pi f} = \pi \delta(\omega) \mp \frac{1}{j\omega} \quad (1.143)$$

Hint: Following these steps:

- a. Introduce an extra term e^{-at} with a real parameter $a > 0$ so that the integrand becomes $e^{-(a+j\omega)t}$ and the integral can be carried out. Note that we cannot take the limit $a \rightarrow 0$ for the integral result due to the singularity at $f = 0$.
- b. Take the limit $a \rightarrow 0$ on the imaginary part, which is odd without singularity at $f = 0$.
- c. Take the limit on the real part, which is even with a singularity at $f = 0$. However, show this impulse is one half of Dirac delta as its integral over $-\infty < f < \infty$ is $1/2$. You may need to use this integral:

$$\int \frac{1}{a^2 + x^2} dx = \frac{1}{a} \tan^{-1} \left(\frac{x}{a} \right) \quad (1.144)$$

5. Prove the identity in Eq.1.33:

$$\frac{1}{T} \int_T e^{\pm j2\pi kt/T} dt = \delta[k] \quad (1.145)$$

Hint: Use Euler's formula to represent the integrand as:

$$e^{\pm j2\pi kt/T} = \cos \left(\frac{2\pi t}{T/k} \right) \pm j \sin \left(\frac{2\pi t}{T/k} \right) \quad (1.146)$$

6. Prove the identity in Eq.1.35:

$$\frac{1}{F} \sum_{k=-\infty}^{\infty} e^{\pm j2k\pi f/F} = \sum_{n=-\infty}^{\infty} \delta(f - nF) \quad (1.147)$$

Hint: Follow these steps:

- a. Find the summation of the following series:

$$\sum_{k=-\infty}^{\infty} (ae^x)^k = \sum_{k=0}^{\infty} (ae^x)^k + \sum_{k=-\infty}^0 (ae^x)^k - 1 = \sum_{k=0}^{\infty} (ae^x)^k + \sum_{k=0}^{\infty} (ae^{-x})^k - 1 \quad (1.148)$$

based on the power series formula for $|a| < 1$:

$$\sum_{k=0}^{\infty} (ae^x)^k = \frac{1}{1 - ae^x} \quad (1.149)$$

- b. Show that when $a = 1$ the sum above is zero if $f \neq nF$ but infinity when $f = nF$, for any integer n , i.e., the sum is a train of impulses.
c. Show that each impulse is a Dirac delta, a unit impulse, as its integral over the period of F with respect to f is 1. Here the result of the previous problem may be needed.
7. Prove the identity in Eq.1.37:

$$\sum_{m=0}^{\infty} e^{-j2\pi fm} = \frac{1}{2} \sum_{k=-\infty}^{\infty} \delta(f - k) + \frac{1}{1 - e^{-j2\pi f}} = \sum_{k=-\infty}^{\infty} \pi \delta(\omega - 2k\pi) + \frac{1}{1 - e^{-j\omega}} \quad (1.150)$$

Hint: Following these steps:

- a. Introduce an extra term a^n with a real parameter $0 < a < 1$ so that the summation term becomes $(ae^{-j\omega})^n$ and the summation can be carried out. Note that we cannot take the limit $a \rightarrow 1$ directly on the result due to the singularity at $f = k$ ($\omega = 2k\pi$) for any integer value of k .
b. Take the limit $a \rightarrow 0$ on the imaginary part, which is odd without singularity at $f = k$.
c. Take the limit on the real part, which is even with a singularity at $f = k$. However, show each impulse is one half of Dirac delta as its integral over $-1/2 < f - k < 1/2$ is 1/2. You may need to use this integral:

$$\int \frac{dx}{a^2 + b^2 - 2ab \cos x} = \frac{2}{a^2 - b^2} \tan^{-1} \left[\frac{a+b}{a-b} \tan \left(\frac{x}{2} \right) \right] \quad (1.151)$$

8. Prove the identity in Eq.1.40:

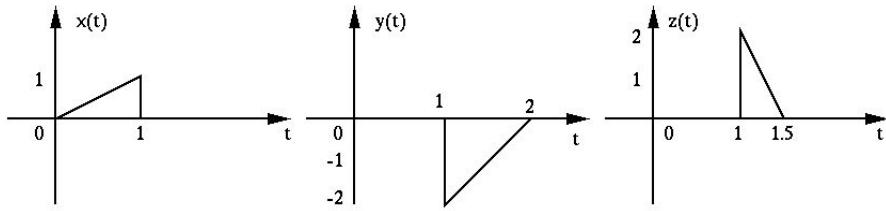
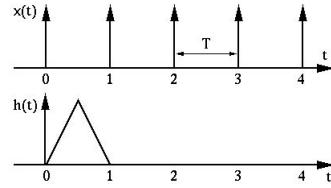
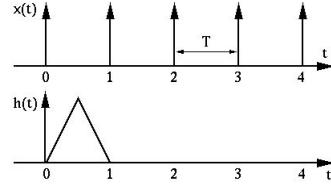
$$\frac{1}{N} \sum_{n=0}^{N-1} e^{\pm j2\pi nm/N} = \sum_{k=-\infty}^{\infty} \delta[m - kN] \quad (1.152)$$

Hint: Consider the summation on the left-hand side in the following two cases to show that:

- a. If $m = kN$ for any integer value of k , the summation is 1;
b. If $m \neq kN$, the summation is 0, based on the formula of geometric series:

$$\sum_{n=0}^{N-1} x^n = \frac{1 - x^N}{1 - x} \quad (1.153)$$

9. Consider the three signals $x(t)$, $y(t)$ and $z(t)$ in Fig.1.11.

**Figure 1.11** Orthogonal Projection**Figure 1.12** Impulse and input of an LTI system**Figure 1.13** Impulse and input of an LTI system

- Give the expressions for \$y(t)\$ in terms of \$x(t)\$.
 - Give the expressions for \$z(t)\$ in terms of \$x(t)\$.
 - Give the expressions for \$y(t)\$ in terms of \$z(t)\$.
 - Give the expressions for \$z(t)\$ in terms of \$y(t)\$.
10. Let \$\boldsymbol{x} = [1, 1, -1, -1, 1, 1, -1, -1]^T\$ be the input to an LTI system with impulse response \$\boldsymbol{h} = [1, 2, 3]^T\$. Find the output \$y[n] = h[n] * x[n]\$. Write a Matlab program to Confirm your result.

Note that given the input \$x[n]\$ and the corresponding output \$y[n]\$, it is difficult to find \$h[n]\$, similarly, given the output \$y[n]\$ and the impulse response \$h[n]\$, it is also difficult to find the input \$x[n]\$. As we will see later, such difficulties can be resolved by the Fourier transform method in frequency domain.

11. The impulse response \$h(t)\$ of an LTI system is shown in Fig.1.13, and the input signal is \$x(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT)\$. Draw the system's response \$y(t) = h(t) * x(t)\$ when \$T\$ takes each of the these values: \$T = 2\$, \$T = 1\$, \$T = 2/3\$, \$T = 1/2\$, and \$T = 1/3\$.
12. The impulse response of an LTI system is

$$h(t) = \begin{cases} 1 & 0 < t < T \\ 0 & \text{else} \end{cases} \quad (1.154)$$

Find the response of the system to an input $x(t) = \cos(2\pi ft)$, and then write a Matlab program to Confirm your result.

13. The impulse response of a discrete LTI system is $h[n] = a^n u[n]$ with $|a| < 1$ and the input is $x[n] = \cos(2\pi n f_0)$. Find the corresponding output $y[n] = h[n] * x[n]$.

Hint: when needed, any complex expression (such as $1/(1 - ae^{j2\pi f_0})$) can be represented in polar form $re^{j\theta}$. But the magnitude r and angle θ need to be expressed in terms of the given parameters (such as a and f_0).

2 Vector Spaces and Signal Representation

In this chapter we discuss some basic concepts of Hilber space and the related operations and properties as the mathematical foundation for the topics of the subsequent chapters. Specifically, based on the concept of unitary transformation in a Hilbert space, all of the unitary transform methods to be specifically considered in the following chapters can be treated from a unified point of view: they are just a set of different rotations of the standard basis of the Hilber space in which a given signal, as a vector, resides. By such a rotation the signal can be better represented in the sense that the variouis signal processing needs, such as noise filtering, information extraction and data compression, can all be carried out more effectively and efficiently.

2.1 Inner Product Space

2.1.1 Vector Space

In our future discussion, any signal, either a continuous one represented as a time function $x(t)$, or a discrete one represented as a vector $\mathbf{x} = [\dots, x[n], \dots]^T$, will be considered as a *vector* in a *vector space*, which is just a generalization of the familiar concept of N-dimensional (N-D) space, formally defined as below.

Definition 2.1. *A vector space is a set V with two operations of addition and scalar multiplication defined for its members, referred to as vectors.*

1. *Vector addition maps any two vectors $\mathbf{x}, \mathbf{y} \in V$ to another vector $\mathbf{x} + \mathbf{y} \in V$ satisfying the following properties:*
 - *Commutativity: $\mathbf{x} + \mathbf{y} = \mathbf{y} + \mathbf{x}$*
 - *Associativity: $\mathbf{x} + (\mathbf{y} + \mathbf{z}) = (\mathbf{x} + \mathbf{y}) + \mathbf{z}$*
 - *Existence of zero: there is a vector $\mathbf{0} \in V$ such that: $\mathbf{0} + \mathbf{x} = \mathbf{x} + \mathbf{0} = \mathbf{x}$*
 - *Existence of inverse: for any vector $\mathbf{x} \in V$, there is another vector $-\mathbf{x} \in V$ such that $\mathbf{x} + (-\mathbf{x}) = \mathbf{0}$*
2. *Scalar multiplication maps a vector $\mathbf{x} \in V$ and a real or complex scalar $a \in \mathbb{C}$ to another vector $a\mathbf{x} \in V$ with the following properties:*
 - $a(\mathbf{x} + \mathbf{y}) = a\mathbf{x} + a\mathbf{y}$
 - $(a + b)\mathbf{x} = a\mathbf{x} + b\mathbf{x}$

- $a\mathbf{b}\mathbf{x} = a(\mathbf{b}\mathbf{x})$
- $1\mathbf{x} = \mathbf{x}$

Listed below is a set of typical vector spaces for various types of signals of interest.

- N-dimensional vector space \mathbb{R}^N or \mathbb{C}^N

This space contains all N-dimensional (N-D) vectors expressed as an N-tuple, an ordered list of N elements (or components):

$$\mathbf{x} = \begin{bmatrix} x[1] \\ x[2] \\ \vdots \\ x[N] \end{bmatrix} = [x[1], x[2], \dots, x[N]]^T \quad (2.1)$$

which can be used to represent a discrete signal containing N samples. We will always represent a vector as a column vector, or the transpose of a row vector. The space is denoted by either \mathbb{C}^N if the elements are complex $x[n] \in \mathbb{C}$, or \mathbb{R}^N if they are all real $x[n] \in \mathbb{R}$ ($n = 1, \dots, N$). Sometimes the N elements of a vector can be alternatively indexed by $n = 0, \dots, N - 1$ to gain some convenience, as can be seen in future chapters.

- A vector space can be defined to contain all $M \times N$ matrices composed of N M-D column vectors:

$$\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_N] = \begin{bmatrix} x[1, 1] & x[1, 2] & \cdots & x[1, N] \\ x[2, 1] & x[2, 2] & \cdots & x[2, N] \\ \vdots & \vdots & \ddots & \vdots \\ x[M, 1] & x[M, 2] & \cdots & x[M, N] \end{bmatrix} \quad (2.2)$$

where the n th column is an M-D vector $\mathbf{x}_n = [x[1, n], \dots, x[M, n]]^T$. Such a matrix can be converted to an MN-D vector by cascading all of the column (or row) vectors. A matrix \mathbf{X} can be used to represent a 2-D signal, such as an image.

- l^2 space:

The dimension N of \mathbb{R}^N or \mathbb{C}^N can be extended to infinity so that a vector in the space becomes a sequence $\mathbf{x} = [\dots, x[n], \dots]^T$ for $0 \leq n < \infty$ or $-\infty < n < \infty$. If all vectors are square-summable, the space is denoted by l^2 . All discrete energy signals are vectors in l^2 .

- \mathcal{L}^2 space:

A vector space can also be a set of real or complex valued continuous functions $x(t)$ defined over either a finite range such as $0 \leq t < T$, or an infinite range $-\infty < t < \infty$. If all functions are square-integrable, the space is denoted by \mathcal{L}^2 . All continuous energy signals are vectors in \mathcal{L}^2 .

Note that the term “vector”, generally denoted by \mathbf{x} , may be interpreted in two different ways. First, in the most general sense, it represents a member of a

vector space, such as any of the vector spaces considered above, e.g., a function $\mathbf{x} = x(t) \in \mathcal{L}^2$. Second, in a more narrow sense, it can also represent a tuple of N elements, an N-D vector $\mathbf{x} = [x[1], \dots, x[N]]^T \in \mathbb{C}^N$, where N may be infinity. It should be clear what a vector \mathbf{x} represents from the context in our future discussion.

Definition 2.2. *The sum of two subspaces $S_1 \subset V$ and $S_2 \subset V$ of a vector space V is defined as*

$$S_1 + S_2 = \{\mathbf{s}_1 + \mathbf{s}_2 | \mathbf{s}_1 \in S_1, \mathbf{s}_2 \in S_2\} \quad (2.3)$$

In particular, if S_1 and S_2 are mutually exclusive:

$$S_1 \cap S_2 = \emptyset \quad (2.4)$$

then their sum $S_1 + S_2$ is called direct sum, denoted by $S_1 \oplus S_2$. Moreover, if $S_1 \oplus S_2 = V$, then S_1 and S_2 form a direct sum decomposition of the vector space V , and S_1 and S_2 are said to be complementary. The direct sum decomposition of V can be generalized to include multiple subspaces:

$$V = \bigoplus_{n=1}^N S_n = S_1 \oplus \dots \oplus S_N \quad (2.5)$$

where all subspaces $S_n \subset V$ are mutually exclusive:

$$S_m \cap S_n = \emptyset, \quad (m \neq n) \quad (2.6)$$

Definition 2.3. *Let $S_1 \subset V$ and $S_2 \subset V$ be subsets of V and $S_1 \oplus S_2 = V$. Then*

$$\mathbf{p}_{S_1, S_2}(\mathbf{s}_1 + \mathbf{s}_2) = \mathbf{s}_1, \quad (\mathbf{s}_1 \in S_1, \mathbf{s}_2 \in S_2) \quad (2.7)$$

is called the projection of $\mathbf{s}_1 + \mathbf{s}_2$ onto S_1 along S_2 .

2.1.2 Inner Product Space

Definition 2.4. *An inner product on a vector space V is a function that maps two vectors $\mathbf{x}, \mathbf{y} \in V$ to a scalar $\langle \mathbf{x}, \mathbf{y} \rangle \in \mathbb{C}$ and satisfies the following conditions:*

- *Positive definiteness:*

$$\langle \mathbf{x}, \mathbf{x} \rangle \geq 0, \quad \langle \mathbf{x}, \mathbf{x} \rangle = 0 \quad \text{iff } \mathbf{x} = \mathbf{0} \quad (2.8)$$

- *Conjugate symmetry:*

$$\langle \mathbf{x}, \mathbf{y} \rangle = \overline{\langle \mathbf{y}, \mathbf{x} \rangle} \quad (2.9)$$

If the vector space is real, the inner product becomes symmetric:

$$\langle \mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{y}, \mathbf{x} \rangle \quad (2.10)$$

- *Linearity in the first variable:*

$$\langle a\mathbf{x} + b\mathbf{y}, \mathbf{z} \rangle = a\langle \mathbf{x}, \mathbf{z} \rangle + b\langle \mathbf{y}, \mathbf{z} \rangle \quad (2.11)$$

where $a, b \in \mathbb{C}$. The linearity does not apply to the second variable:

$$\begin{aligned} <\mathbf{x}, a\mathbf{y} + b\mathbf{z}> &= \overline{<a\mathbf{y} + b\mathbf{z}, \mathbf{x}>} = \overline{a <\mathbf{y}, \mathbf{x}>} + b <\mathbf{z}, \mathbf{x}> \\ &= \bar{a} <\mathbf{x}, \mathbf{y}> + \bar{b} <\mathbf{x}, \mathbf{z}> \neq a <\mathbf{x}, \mathbf{y}> + b <\mathbf{x}, \mathbf{z}> \end{aligned} \quad (2.12)$$

unless the coefficients are real $a, b \in \mathbb{R}$. As a special case, when $b = 0$, we have:

$$<\mathbf{x}, a\mathbf{y}> = a <\mathbf{x}, \mathbf{y}>, \quad <\mathbf{x}, a\mathbf{y}> = \bar{a} <\mathbf{x}, \mathbf{y}> \quad (2.13)$$

More generally we have:

$$<\sum_n c_n \mathbf{x}_n, \mathbf{y}> = \sum_n c_n <\mathbf{x}_n, \mathbf{y}>, \quad <\mathbf{x}, \sum_n c_n \mathbf{y}_n> = \sum_n \bar{c}_n <\mathbf{x}, \mathbf{y}_n> \quad (2.14)$$

Definition 2.5. A vector space with inner product defined is called an inner product space.

In particular, when the inner product is defined, \mathbb{C}^N is called a *unitary space* and \mathbb{R}^N is called a *Euclidean space*. All vector spaces in the future discussion will be assumed to be inner product spaces. Some examples of the inner product are listed below:

- In an N-D vector space, the inner product, also called the *dot product*, of two vectors $\mathbf{x} = [x[1], \dots, x[N]]^T$ and $\mathbf{y} = [y[1], \dots, y[N]]^T$ is defined as:

$$<\mathbf{x}, \mathbf{y}> = \mathbf{x}^T \bar{\mathbf{y}} = \mathbf{y}^* \mathbf{x} = [x[1], x[2], \dots, x[N]] \begin{bmatrix} \bar{y}[1] \\ \bar{y}[2] \\ \vdots \\ \bar{y}[N] \end{bmatrix} = \sum_{n=1}^N x[n] \bar{y}[n] \quad (2.15)$$

where $\mathbf{y}^* = \bar{\mathbf{y}}^T$ is the conjugate transpose of \mathbf{y} .

- In a space of 2-D matrices $\mathbf{X}_{M \times N}$ containing $M \times N$ elements $x[m, n]$ ($m = 1, \dots, M$, $n = 1, \dots, N$), the inner product of two such matrices \mathbf{X} and \mathbf{Y} is defined as:

$$<\mathbf{X}, \mathbf{Y}> = \sum_{m=1}^M \sum_{n=1}^N x[m, n] \bar{y}[m, n] \quad (2.16)$$

This inner product is equivalent to Eq.2.15 if we cascade the column (or row) vectors of \mathbf{X} and \mathbf{Y} to form two MN-D vectors.

- In a function space, the inner product of two function vectors $\mathbf{x} = x(t)$ and $\mathbf{y} = y(t)$ is defined as:

$$<x(t), y(t)> = \int_a^b x(t) \bar{y}(t) dt = \overline{\int_a^b x(t) y(t) dt} = \overline{<y(t), x(t)>} \quad (2.17)$$

In particular, Eq.1.10 for the sifting property of the delta function $\delta(t)$ is an inner product:

$$\langle x(t), \delta(t) \rangle = \int_{-\infty}^{\infty} x(\tau) \delta(\tau) d\tau = x(0) \quad (2.18)$$

- The inner product of two random variables x and y can be defined as:

$$\langle x, y \rangle = E[x\bar{y}] \quad (2.19)$$

If the two random variables have zero means, i.e., $\mu_x = E(x) = 0$ and $\mu_y = E(y) = 0$, the inner product above is also their covariance:

$$\sigma_{xy}^2 = E[(x - \mu_x)(\bar{y} - \mu_y)] = E(x\bar{y}) - \mu_x \mu_y = E(x\bar{y}) = \langle x, y \rangle \quad (2.20)$$

The concept of inner product is of essential importance based on which a whole set of other important concepts can be defined.

Definition 2.6. If the inner product of two vectors \mathbf{x} and \mathbf{y} is zero, $\langle \mathbf{x}, \mathbf{y} \rangle = 0$, they are orthogonal (perpendicular) to each other, denoted by $\mathbf{x} \perp \mathbf{y}$.

Definition 2.7. The norm (or length) of a vector $\mathbf{x} \in V$ is defined as:

$$\|\mathbf{x}\| = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle} = \langle \mathbf{x}, \mathbf{x} \rangle^{1/2}, \quad \text{or} \quad \|\mathbf{x}\|^2 = \langle \mathbf{x}, \mathbf{x} \rangle \quad (2.21)$$

The norm $\|\mathbf{x}\|$ is nonnegative and it is zero if and only if $\mathbf{x} = \mathbf{0}$. In particular, if $\|\mathbf{x}\| = 1$, then it is said to be *normalized* and becomes a *unit vector*. Any vector can be normalized when divided by its own norm: $\mathbf{x}/\|\mathbf{x}\|$. The vector norm squared $\|\mathbf{x}\|^2 = \langle \mathbf{x}, \mathbf{x} \rangle$ can be considered as the energy of the vector.

Specifically in an N-D unitary space, the norm of a vector $\mathbf{x} = [x[1], \dots, x[N]]^T \in \mathbb{C}^N$ is:

$$\|\mathbf{x}\| = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle} = \sqrt{\mathbf{x}^T \mathbf{x}} = \left[\sum_{n=1}^N x[n] \bar{x}[n] \right]^{1/2} = \left[\sum_{n=1}^N |x[n]|^2 \right]^{1/2} \quad (2.22)$$

The total energy contained in this vector is its norm squared:

$$\mathcal{E} = \|\mathbf{x}\|^2 = \langle \mathbf{x}, \mathbf{x} \rangle = \sum_{n=1}^N |x[n]|^2 \quad (2.23)$$

This norm can be generalized to *p-norm* defined as:

$$\|\mathbf{x}\|_p = \left[\sum_{n=1}^N |x[n]|^p \right]^{1/p} \quad (2.24)$$

Particularly

$$\|\mathbf{x}\|_1 = \sum_{n=1}^N |x[n]|, \quad \|\mathbf{x}\|_\infty = \max(|x[1]|, \dots, |x[N]|) \quad (2.25)$$

The norm of a matrix \mathbf{X} can be defined differently but here we will only consider the element-wise norm defined as:

$$\|\mathbf{X}\|_p = \left[\sum_{n=1}^N |x[m][n]|^p \right]^{1/p} \quad (2.26)$$

When $p = 2$, $\|\mathbf{X}\|_2^2$ can be considered as the total energy contained in the 2-D signal \mathbf{X} . We will always use this matrix norm in the future.

The concept of N-D unitary (or Euclidean) space can be generalized to an infinite-dimensional space, in which case the range of the summation will cover all real integers \mathbb{Z} in the entire real axis $-\infty < n < \infty$. This norm exists only if the summation converges to a finite value, i.e., the vector \mathbf{x} is an energy signal with finite energy:

$$\sum_{n=-\infty}^{\infty} |x[n]|^2 < \infty \quad (2.27)$$

All such vectors \mathbf{x} satisfying the above are square-summable and form the vector space denoted by $l^2(\mathbb{Z})$.

Similarly, in a function space, the norm of a function vector $\mathbf{x} = x(t)$ is defined as:

$$\|\mathbf{x}\| = \left(\int_a^b x(t) \overline{x(t)} dt \right)^{1/2} = \left(\int_a^b |x(t)|^2 dt \right)^{1/2} \quad (2.28)$$

where the lower and upper integral limits $a < b$ are two real numbers, which may be extended to all real values \mathbb{R} in the entire real axis $-\infty < t < \infty$. This norm exists only if the integral converges to a finite value, i.e., $x(t)$ is an energy signal containing finite energy:

$$\int_{-\infty}^{\infty} |x(t)|^2 dt < \infty \quad (2.29)$$

All such functions $x(t)$ satisfying the above are square-integrable, and they form a function space denoted by $L^2(\mathbb{R})$.

All vectors and functions in the future discussion are assumed to be square-summable/integrable, i.e., they represent energy signals containing finite amount of energy, so that these conditions do not need to be mentioned every time a signal vector is considered.

Theorem 2.1. (*The Cauchy-Schwarz inequality*) *The following inequality holds for any two vectors $\mathbf{x}, \mathbf{y} \in V$ in an inner product space V :*

$$|\langle \mathbf{x}, \mathbf{y} \rangle|^2 \leq \langle \mathbf{x}, \mathbf{x} \rangle \langle \mathbf{y}, \mathbf{y} \rangle, \quad \text{i.e.,} \quad 0 \leq |\langle \mathbf{x}, \mathbf{y} \rangle| \leq \|\mathbf{x}\| \|\mathbf{y}\| \quad (2.30)$$

Proof: If either \mathbf{x} or \mathbf{y} is zero, we have $\langle \mathbf{x}, \mathbf{y} \rangle = 0$, i.e., Eq. 2.30 holds (an equality). Otherwise, we consider the following inner product:

$$\langle \mathbf{x} - \lambda \mathbf{y}, \mathbf{x} - \lambda \mathbf{y} \rangle = \|\mathbf{x}\|^2 - \bar{\lambda} \langle \mathbf{x}, \mathbf{y} \rangle - \lambda \langle \mathbf{y}, \mathbf{x} \rangle + |\lambda|^2 \|\mathbf{y}\|^2 \geq 0 \quad (2.31)$$

where $\lambda \in \mathbb{C}$ is an arbitrary complex number, which can be assumed to be:

$$\lambda = \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{y}\|^2}, \quad \text{then} \quad \bar{\lambda} = \frac{\langle \mathbf{y}, \mathbf{x} \rangle}{\|\mathbf{y}\|^2}, \quad |\lambda|^2 = \frac{|\langle \mathbf{x}, \mathbf{y} \rangle|^2}{\|\mathbf{y}\|^4} \quad (2.32)$$

Substitute these into Eq.2.31, we get

$$\|\mathbf{x}\|^2 - \frac{|\langle \mathbf{x}, \mathbf{y} \rangle|^2}{\|\mathbf{y}\|^2} \geq 0, \quad \text{i.e.,} \quad |\langle \mathbf{x}, \mathbf{y} \rangle| \leq \|\mathbf{x}\| \|\mathbf{y}\| \quad (2.33)$$

Definition 2.8. *The angle between two vectors \mathbf{x} and \mathbf{y} is defined as:*

$$\theta = \cos^{-1} \left(\frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{x}\| \|\mathbf{y}\|} \right) \quad (2.34)$$

Now the inner product of \mathbf{x} and \mathbf{y} can also be written as

$$\langle \mathbf{x}, \mathbf{y} \rangle = \|\mathbf{x}\| \|\mathbf{y}\| \cos \theta \quad (2.35)$$

In particular, if $\theta = 0$, then $\cos \theta = 1$, and \mathbf{x} and \mathbf{y} are collinear, the inner product $\langle \mathbf{x}, \mathbf{y} \rangle = \|\mathbf{x}\| \|\mathbf{y}\|$ in Eq.2.30 is maximized. Else if $\theta = \pi/2$, then $\cos \theta = 0$, and \mathbf{x} and \mathbf{y} are orthogonal to each other, the inner product $\langle \mathbf{x}, \mathbf{y} \rangle = 0$ is minimized.

Definition 2.9. *The orthogonal projection of a vector $\mathbf{x} \in V$ onto another vector $\mathbf{y} \in V$ is defined as*

$$\mathbf{p}_y(\mathbf{x}) = \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{y}\|} \frac{\mathbf{y}}{\|\mathbf{y}\|} = \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\langle \mathbf{y}, \mathbf{y} \rangle} \mathbf{y} = \|\mathbf{x}\| \cos \theta \frac{\mathbf{y}}{\|\mathbf{y}\|} \quad (2.36)$$

where $\theta = \cos^{-1}[\langle \mathbf{x}, \mathbf{y} \rangle / (\|\mathbf{x}\| \|\mathbf{y}\|)]$ is the angle between the two vectors.

The projection $\mathbf{p}_y(\mathbf{x})$ is a vector and its norm is a scalar denoted by:

$$p_y(\mathbf{x}) = \|\mathbf{p}_y(\mathbf{x})\| = \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{y}\|} = \|\mathbf{x}\| \cos \theta \quad (2.37)$$

which is sometimes also referred to as the scalar projection or simply projection. The projection $\mathbf{p}_y(\mathbf{x})$ is illustrated in Fig.2.1. In particular, if \mathbf{y} is a unit (normalized) vector with $\|\mathbf{y}\| = 1$, we have

$$\mathbf{p}_y(\mathbf{x}) = \langle \mathbf{x}, \mathbf{y} \rangle \mathbf{y}, \quad \|\mathbf{p}_y(\mathbf{x})\| = \langle \mathbf{x}, \mathbf{y} \rangle \quad (2.38)$$

In other words, the magnitude of the projection of \mathbf{x} onto a unit vector is simply their inner product.

Example 2.1: Find the projection of $\mathbf{x} = [1, 2]^T$ onto $\mathbf{y} = [3, 1]^T$.

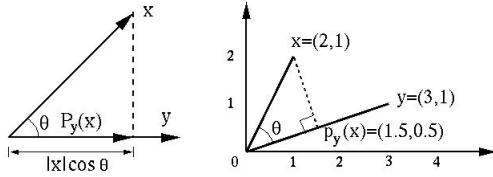


Figure 2.1 Orthogonal Projection

The angle between the two vectors is

$$\theta = \cos^{-1} \left(\frac{\langle x, y \rangle}{\sqrt{\langle x, x \rangle \langle y, y \rangle}} \right) = \cos^{-1} \left(\frac{5}{\sqrt{5 \times 10}} \right) = \cos^{-1} 0.707 = 45^\circ \quad (2.39)$$

The projection of x on y is:

$$p_y(x) = \frac{\langle x, y \rangle}{\langle y, y \rangle} y = \frac{5}{10} \begin{bmatrix} 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 1.5 \\ 0.5 \end{bmatrix} \quad (2.40)$$

The norm of the projection is $\sqrt{1.5^2 + 0.5^2} \approx 1.58$, which is of course the same as $\|x\| \cos \theta = \sqrt{5} \cos 45^\circ \approx 1.58$. If y is normalized to become $z = y/\|y\| = [3, 1]/\sqrt{10}$, then the projection of x onto z can be simply obtained as their inner product:

$$p_z(x) = \|p_z(x)\| = \langle x, z \rangle = [1, 2] \begin{bmatrix} 3 \\ 1 \end{bmatrix} / \sqrt{10} = 5 / \sqrt{10} \approx 1.58 \quad (2.41)$$

Definition 2.10. Two subspaces $S_1 \subset V$ and $S_2 \subset V$ of an inner product space V are orthogonal, denoted by $S_1 \perp S_2$, if $s_1 \perp s_2$ for any $s_1 \in S_1$ and $s_2 \in S_2$. In particular, if one of the subsets contains only one vector $S_1 = \{s_1\}$, then the vector is orthogonal to the other subset $s_1 \perp S_2$.

Definition 2.11. The orthogonal complement of a subspace $S \subset V$ is the set of all vectors in V that are orthogonal to S :

$$S^\perp = \{v \in V \mid v \perp S\} = \{v \in V \mid \langle v, u \rangle = 0, \forall u \in S\} \quad (2.42)$$

Definition 2.12. An inner product space V as the direct sum of n mutually orthogonal subspaces $S_i \subset V$ ($i = 1, \dots, n$) is called the orthogonal direct sum of these subspaces:

$$V = S_1 \oplus \dots \oplus S_n, \quad \text{with } S_i \perp S_j \quad \text{for all } i \neq j \quad (2.43)$$

It can be shown that if $V = S_1 \oplus S_2$ and $S_1 \perp S_2$, then

$$S \cap S^\perp = \emptyset, \quad \text{and} \quad S \oplus S^\perp = V \quad (2.44)$$

Definition 2.13. Let $S \subset V$ and $S \oplus S^\perp = V$ and $s \in S$, $r \in S^\perp$. Then $p_S(s + r) = s$ is called the orthogonal projection of $s + r$ onto S .

All of these definitions can be intuitively and trivially visualized in a 3-D space spanned by three perpendicular coordinates (x, y, z) representing three mutually orthogonal subspaces. The orthogonal direct sum of these subspaces is the 3-D space, and the orthogonal complement of the subspace in x direction is the 2-D y - z plane formed by coordinates y and z . The orthogonal projection of a vector $\mathbf{v} = [1, 2, 3]^T$ onto the subspace in x direction is $[1, 0, 0]^T$, and its orthogonal projection onto the y - z subspace is a 2-D vector $[0, 2, 3]^T$.

Definition 2.14. *The distance between two vectors \mathbf{x}, \mathbf{y} is*

$$d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\| \quad (2.45)$$

Theorem 2.2. *The distance satisfies the following three conditions:*

- *Nonnegative:*

$$d(\mathbf{x}, \mathbf{y}) \geq 0, \quad d(\mathbf{x}, \mathbf{y}) = 0 \quad \text{iff} \quad \mathbf{x} = \mathbf{y} \quad (2.46)$$

- *Symmetric:*

$$d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x}) \quad (2.47)$$

- *Triangle inequality:*

$$d(\mathbf{x}, \mathbf{y}) \leq d(\mathbf{x}, \mathbf{z}) + d(\mathbf{z}, \mathbf{y}) \quad (2.48)$$

Proof: The first two conditions are self-evident based on the definition. We now show the third condition also holds by considering the following:

$$\begin{aligned} \|\mathbf{u} + \mathbf{v}\|^2 &= \langle \mathbf{u} + \mathbf{v}, \mathbf{u} + \mathbf{v} \rangle = \|\mathbf{u}\|^2 + \langle \mathbf{u}, \mathbf{v} \rangle + \langle \mathbf{v}, \mathbf{u} \rangle + \|\mathbf{v}\|^2 \\ &= \|\mathbf{u}\|^2 + 2 \operatorname{Re} \langle \mathbf{u}, \mathbf{v} \rangle + \|\mathbf{v}\|^2 \leq \|\mathbf{u}\|^2 + 2 |\langle \mathbf{u}, \mathbf{v} \rangle| + \|\mathbf{v}\|^2 \\ &\leq \|\mathbf{u}\|^2 + 2 \|\mathbf{u}\| \|\mathbf{v}\| + \|\mathbf{v}\|^2 = (\|\mathbf{u}\| + \|\mathbf{v}\|)^2 \end{aligned} \quad (2.49)$$

The first \leq sign above is due to the fact that the magnitude of a complex number is no less than its real part, and the second \leq sign is simply the Cauchy-Schwarz inequality. Taking the square root on both sides, we get:

$$\|\mathbf{u} + \mathbf{v}\| \leq \|\mathbf{u}\| + \|\mathbf{v}\| \quad (2.50)$$

We further let $\mathbf{u} = \mathbf{x} - \mathbf{z}$ and $\mathbf{v} = \mathbf{z} - \mathbf{y}$, the above becomes the triangle inequality:

$$\|\mathbf{x} - \mathbf{y}\| \leq \|\mathbf{x} - \mathbf{z}\| + \|\mathbf{z} - \mathbf{y}\| \quad (2.51)$$

This is Eq.2.48. Q.E.D.

Definition 2.15. *When distance is defined between any two vectors in a vector space, it is called a metric space.*

In a unitary space \mathbb{C}^N , the *Euclidean distance* between any two vectors \mathbf{x} and \mathbf{y} can be defined as the norm of the difference vector $\mathbf{x} - \mathbf{y}$:

$$d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\| = \left(\sum_{n=1}^N |x[n] - y[n]|^2 \right)^{1/2} \quad (2.52)$$

This distance can be considered as a special case ($p = 2$) of the more general *p-norm distance*:

$$d_p(\mathbf{x}, \mathbf{y}) = \left(\sum_{n=1}^N |x[n] - y[n]|^p \right)^{1/p} \quad (2.53)$$

Other commonly used p-norm distances include:

$$d_1(\mathbf{x}, \mathbf{y}) = \sum_{n=1}^N |x[n] - y[n]| \quad (2.54)$$

$$d_\infty(\mathbf{x}, \mathbf{y}) = \max(|x[1] - y[1]|, \dots, |x[N] - y[N]|) \quad (2.55)$$

In a function space, p-norm distance between two functions $x(t)$ and $y(t)$ is similarly defined as:

$$d_p(x(t), y(t)) = \left(\int_a^b |x(t) - y(t)|^p dt \right)^{1/p} \quad (2.56)$$

In particular when $p = 2$, we have:

$$d_2(x(t), y(t)) = \|x(t) - y(t)\| = \left(\int_a^b |x(t) - y(t)|^2 dt \right)^{1/2} \quad (2.57)$$

2.1.3 Bases of Vector Space

Definition 2.16. In a vector space V , the subspace W of all linear combinations of a set of M vectors $\mathbf{b}_k \in V$, ($k = 1, \dots, M$) is called the *linear span* of the vectors:

$$W = \text{span}(\mathbf{b}_1, \dots, \mathbf{b}_M) = \left\{ \sum_{k=1}^M c[k] \mathbf{b}_k \mid c[k] \in \mathbb{C} \right\} \quad (2.58)$$

Definition 2.17. A set of linearly independent vectors that spans a vector space is called a *basis* of the space.

The basis vectors are linearly independent, i.e., none of them can be represented as a linear combination of the rest. They are also complete, i.e., including any additional vector in the basis it would no longer be linearly independent, and removing any of them would result in inability to represent certain vectors in the space. In other words, a basis is a minimum set of vectors capable of

representing any vector in the space. Also as any rotation of a given basis will result in a different basis, we see that there are infinitely many bases that all span the same space. This idea is of great importance in our future discussion.

For example, any vector $\mathbf{x} \in \mathbb{C}^N$ can be uniquely expressed as a linear combination of some N basis vectors \mathbf{b}_k :

$$\mathbf{x} = \sum_{k=1}^N c[k] \mathbf{b}_k \quad (2.59)$$

Moreover, the concept of a finite N-D space spanned by a basis composed of N discrete (countable) linearly independent vectors can be generalized to a vector space V spanned by a basis composed of a family of uncountably infinite vectors $\mathbf{b}(f)$. Any vector $\mathbf{x} \in V$ in the space can be expressed as a linear combination, an integral, of these basis vectors:

$$\mathbf{x} = \int_a^b c(f) \mathbf{b}(f) df \quad (2.60)$$

We see that the index k for the summation in Eq.2.59 is replaced by a continuous variable f for the integral, and the coefficients $c[k]$ is replace by a continuous weighting function $c(f)$ for the set of uncountable basis vectors $\mathbf{b}(f)$ with $a < f < b$. The significance of this generalization becomes clear during our future discussion of orthogonal transforms of continuous signals $x(t)$. An important issue is how to find the coefficients $c[k]$ or the weighting function $c(f)$, given the vector \mathbf{x} and the basis \mathbf{b}_k or $\mathbf{b}(f)$.

Consider specifically the case of an N-D unitary space \mathbb{C}^N as an example. Let $\{\mathbf{b}_1, \dots, \mathbf{b}_M\}$ be a basis consisting of M linearly independent N-D vectors. Then any vector $\mathbf{x} \in \mathbb{C}^N$ can be represented as a linear combination of these basis vectors:

$$\mathbf{x} = \begin{bmatrix} x[1] \\ \vdots \\ x[N] \end{bmatrix}_{N \times 1} = \sum_{k=1}^M c[k] \mathbf{b}_k = [\mathbf{b}_1, \dots, \mathbf{b}_M]_{N \times M} \begin{bmatrix} c[1] \\ \vdots \\ c[M] \end{bmatrix}_{M \times 1} = \mathbf{B}\mathbf{c} \quad (2.61)$$

where $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_M]$ is an N by M matrix composed of the M N-D basis vectors as its columns, and the n th coefficient $c[n]$ is the n th element of an M-D vector $\mathbf{c} = [c[1], \dots, c[M]]^T$. This coefficient vector \mathbf{c} can be found by solving the equation system in Eq.2.61. For the solution to exist, the number of unknown coefficients must be no fewer than the number of constraining equations, i.e., $M \geq N$. On the other hand, as there can be no more than N independent basis vectors in this N-D space, we must also have $M \leq N$. Therefore there must be exactly $M = N$ vectors in a basis of an N-D space. In this case, \mathbf{B} is an N by N square matrix with full rank (as all column vectors are independent), i.e., its inverse \mathbf{B}^{-1} exists and the coefficients can be obtained by solving the system

with N unknowns and N equations:

$$\mathbf{c} = \begin{bmatrix} c[1] \\ \vdots \\ c[N] \end{bmatrix} = [\mathbf{b}_1, \dots, \mathbf{b}_N]^{-1} \begin{bmatrix} x[1] \\ \vdots \\ x[N] \end{bmatrix} = \mathbf{B}^{-1} \mathbf{x} \quad (2.62)$$

The computational complexity to solve this system of N equations and N unknowns is $O(N^3)$.

Similarly, we may need to find the weighting function $c(f)$ in Eq.2.60 in order to represent a vector \mathbf{x} in terms of the basis $\mathbf{b}(f)$. However, solving this equation for $c(f)$ is not as trivial as solving Eq.2.61 for \mathbf{c} in the previous case of a vector space spanned by a finite and discrete basis. In the next subsection, this problem will be reconsidered when some additional condition is imposed on the basis \mathbf{c} to make the problem easier to solve.

Example 2.2:

A 2-D Euclidean \mathbb{R}^2 space can be spanned by two basis vectors $\mathbf{e}_1 = [1, 0]^T$ and $\mathbf{e}_2 = [0, 1]^T$, by which two vectors $\mathbf{a}_1 = [1, 0]^T$ and $\mathbf{a}_2 = [-1, 2]^T$ can be represented as:

$$\mathbf{a}_1 = 1\mathbf{e}_1 + 0\mathbf{e}_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \mathbf{a}_2 = -1\mathbf{e}_1 + 2\mathbf{e}_2 = \begin{bmatrix} -1 \\ 2 \end{bmatrix} \quad (2.63)$$

As \mathbf{a}_1 and \mathbf{a}_2 are independent (as they are not co-linear), they in turn form a basis of the space. Any given vector such as

$$\mathbf{x} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} = 1\mathbf{e}_1 + 2\mathbf{e}_2 = 1 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + 2 \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (2.64)$$

can be expressed in terms of $\{\mathbf{a}_1, \mathbf{a}_2\}$ as

$$\mathbf{x} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} = c[1]\mathbf{a}_1 + c[2]\mathbf{a}_2 = c[1] \begin{bmatrix} 1 \\ 0 \end{bmatrix} + c[2] \begin{bmatrix} -1 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} c[1] \\ c[2] \end{bmatrix} \quad (2.65)$$

Solving this we get $c[1] = 2$ and $c[2] = 1$, so that \mathbf{x} can be expressed by \mathbf{a}_1 and \mathbf{a}_2 as:

$$\mathbf{x} = c[1]\mathbf{a}_1 + c[2]\mathbf{a}_2 = 2 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + 1 \begin{bmatrix} -1 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad (2.66)$$

This example is illustrated in Fig.2.2.

Example 2.3: The previous example in \mathbb{R}^2 can also be extended to a function space defined over $[0, 2]$ spanned by two basis functions:

$$a_1(t) = \begin{cases} 1 & 0 \leq t < 1 \\ 0 & 1 \leq t < 2 \end{cases}, \quad a_2(t) = \begin{cases} -1 & 0 \leq t < 1 \\ 2 & 1 \leq t < 2 \end{cases} \quad (2.67)$$

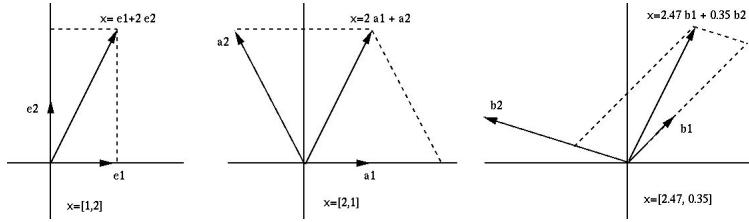


Figure 2.2 Different basis vectors of a 2-D space

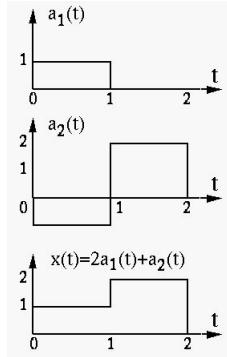


Figure 2.3 Representation of a time function by basis functions

A given time function $x(t)$ in the space

$$x(t) = \begin{cases} 1 & 0 \leq t < 1 \\ 2 & 1 \leq t < 2 \end{cases} \quad (2.68)$$

can be represented by the two basis functions as:

$$x(t) = c[1]a_1(t) + c[2]a_2(t) \quad (2.69)$$

To obtain the coefficients $c[1]$ and $c[2]$, we first find the inner products of this equation with the following two functions:

$$e_1(t) = \begin{cases} 1 & 0 \leq t < 1 \\ 0 & 1 \leq t < 2 \end{cases}, \quad e_2(t) = \begin{cases} 0 & 0 \leq t < 1 \\ 1 & 1 \leq t < 2 \end{cases} \quad (2.70)$$

to get:

$$\begin{aligned} < x(t), e_1(t) > &= 1 = c[1] < a_1(t), e_1(t) > + c[2] < a_2(t), e_1(t) > = c[1] - c[2] \\ < x(t), e_2(t) > &= 2 = c[1] < a_1(t), e_2(t) > + c[2] < a_2(t), e_2(t) > = 2c[2] \end{aligned} \quad (2.71)$$

Solving this equation system, which is identical to that in the previous example, we get the same coefficients $c[1] = 2$ and $c[2] = 1$. Now $x(t)$ can be expressed as $x(t) = 2a_1(t) + a_2(t)$, as illustrated in Fig.2.3.

So far we have only considered inner product spaces of finite dimensions. Additional theory is needed to deal with spaces of infinite dimensions.

Definition 2.18. • In a metric space V , a sequence $\{\mathbf{x}_1, \mathbf{x}_2, \dots\}$ is a Cauchy sequence if for any $\epsilon > 0$, there exists an $N > 0$ such that for any $m, n > N$, $d(\mathbf{x}_m, \mathbf{x}_n) < \epsilon$.

- A metric space V is complete if every Cauchy sequence $\{\mathbf{x}_n\}$ in V converges to a $\mathbf{x} \in V$:

$$\lim_{m \rightarrow \infty} d(\mathbf{x}_m, \mathbf{x}) = \lim_{m \rightarrow \infty} \|\mathbf{x} - \mathbf{x}_m\| = 0 \quad (2.72)$$

In other words, for any $\epsilon > 0$, there exists an $N > 0$ such that

$$d(\mathbf{x}_m, \mathbf{x}) < \epsilon \quad \text{if } m > N \quad (2.73)$$

- A complete inner product space is a Hilbert space, denoted by H .
- Let \mathbf{b}_k be a set of orthogonal vectors ($k = 1, 2, \dots$) in H , and an arbitrary vector \mathbf{x} is approximated in an M -D subspace by

$$\hat{\mathbf{x}}_M = \sum_{k=1}^M c[k] \mathbf{b}_k \quad (2.74)$$

If the least squares error of this approximation $\|\mathbf{x} - \hat{\mathbf{x}}_M\|^2$ converges to zero when $M \rightarrow \infty$, i.e.,

$$\lim_{M \rightarrow \infty} \|\mathbf{x} - \hat{\mathbf{x}}_M\|^2 = \lim_{M \rightarrow \infty} \left\| \mathbf{x} - \sum_{k=1}^M c[k] \mathbf{b}_k \right\|^2 = 0 \quad (2.75)$$

then this set of orthogonal vectors is said to be complete, called a complete orthogonal system, and the approximation converges to the given vector:

$$\lim_{M \rightarrow \infty} \sum_{k=1}^M c[k] \mathbf{b}_k = \sum_{k=1}^{\infty} c[k] \mathbf{b}_k = \mathbf{x} \quad (2.76)$$

In the following, to keep the discussion generic, the lower and upper limits of a summation or an integral may not be always explicitly specified, as the summation or integral may be finite (e.g., from 1 to N) or infinite (e.g., from 0 or $-\infty$ to ∞), depending on each specific case.

2.1.4 Signal Representation by Orthogonal Bases

As shown in Eqs.2.59 and 2.60, a vector $\mathbf{x} \in V$ in a vector space can be represented as a linear combination of a set of linearly independent basis vectors, either countable like \mathbf{b}_k , or uncountable like $\mathbf{b}(f)$, that span the space V . However, it may not be always easy to find the weighting coefficients $c[k]$ or function $c(f)$. As shown in Eq.2.62 for the simple case of the finite dimensional space \mathbb{C}^N , in order to obtain the coefficient vector \mathbf{c} , we need to find the inverse of

the $N \times N$ matrix $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_N]$, which may not be a trivial problem if N is large. Moreover, in the case of uncountable basis $\mathbf{b}(f)$ of Eq.2.60, it is certainly not a trivial problem to find the coefficient function $c(f)$. However, as to be shown below, finding the coefficients $c[k]$ or weighting function $c(f)$ can become most straight forward if the basis is orthogonal.

Theorem 2.3. Let \mathbf{x} and \mathbf{y} be any two vectors in a Hilbert space H spanned by a complete orthonormal system $\{\mathbf{u}_k\}$ satisfying:

$$\langle \mathbf{u}_k, \mathbf{u}_l \rangle = \delta[k - l] \quad (2.77)$$

Then we have:

1. Series expansion:

$$\mathbf{x} = \sum_k \langle \mathbf{x}, \mathbf{u}_k \rangle \mathbf{u}_k \quad (2.78)$$

2. Plancherel theorem:

$$\langle \mathbf{x}, \mathbf{y} \rangle = \sum_k \langle \mathbf{x}, \mathbf{u}_k \rangle \overline{\langle \mathbf{y}, \mathbf{u}_k \rangle} \quad (2.79)$$

3. Parseval's theorem:

$$\langle \mathbf{x}, \mathbf{x} \rangle = \|\mathbf{x}\|^2 = \sum_k |\langle \mathbf{x}, \mathbf{u}_k \rangle|^2 \quad (2.80)$$

Here the dimensionality of the space is not specified to keep the discussion more general.

Proof: As $\{\mathbf{u}_k\}$ is the basis of H , any $\mathbf{x} \in H$ can be written as:

$$\mathbf{x} = \sum_k c[k] \mathbf{u}_k \quad (2.81)$$

Taking an inner product with \mathbf{u}_l on both sides we get

$$\langle \mathbf{x}, \mathbf{u}_l \rangle = \left\langle \sum_k c[k] \mathbf{u}_k, \mathbf{u}_l \right\rangle = \sum_k c[k] \langle \mathbf{u}_k, \mathbf{u}_l \rangle = \sum_k c[k] \delta[k - l] = c[l] \quad (2.82)$$

We therefore have $c[k] = \langle \mathbf{x}, \mathbf{u}_k \rangle$ and

$$\mathbf{x} = \sum_k c[k] \mathbf{u}_k = \sum_k \langle \mathbf{x}, \mathbf{u}_k \rangle \mathbf{u}_k \quad (2.83)$$

Here \mathbf{x} is expressed as the vector sum of its projections $\mathbf{p}_{\mathbf{u}_k}(\mathbf{x}) = \langle \mathbf{x}, \mathbf{u}_k \rangle \mathbf{u}_k$ onto each of the unit basis vectors \mathbf{u}_k (Eq.2.38), and the scalar coefficient $c[k] = \langle \mathbf{x}, \mathbf{u}_k \rangle$ is the norm of the projection. Vector $\mathbf{y} \in H$ can also be written as:

$$\mathbf{y} = \sum_l d[l] \mathbf{u}_l = \sum_l \langle \mathbf{y}, \mathbf{u}_l \rangle \mathbf{u}_l \quad (2.84)$$

and we have:

$$\begin{aligned}
 \langle \mathbf{x}, \mathbf{y} \rangle &= \left\langle \sum_k c[k] \mathbf{u}_k, \sum_l d[l] \mathbf{u}_l \right\rangle = \sum_k c[k] \sum_l \bar{d}[l] \langle \mathbf{u}_k, \mathbf{u}_l \rangle \\
 &= \sum_k c[k] \sum_l \bar{d}[l] \delta[k-l] = \sum_k c[k] \bar{d}[k] \\
 &= \sum_k \langle \mathbf{x}, \mathbf{u}_k \rangle \overline{\langle \mathbf{y}, \mathbf{u}_k \rangle} = \langle \mathbf{c}, \mathbf{d} \rangle
 \end{aligned} \tag{2.85}$$

where $\mathbf{c} = [\dots, c[k], \dots]^T$ and $\mathbf{d} = [\dots, d[k], \dots]^T$ are the coefficient vectors of either finite or infinite dimensions. This is the Plancherel theorem. In particular, when $\mathbf{x} = \mathbf{y}$, we have:

$$\langle \mathbf{x}, \mathbf{x} \rangle = \|\mathbf{x}\|^2 = \sum_k |\langle \mathbf{x}, \mathbf{u}_k \rangle|^2 = \sum_k |c[k]|^2 = \langle \mathbf{c}, \mathbf{c} \rangle = \|\mathbf{c}\|^2 \tag{2.86}$$

This is Parseval's theorem or identity. Q.E.D.

Eqs.2.82 and 2.83 can be combined to form a pair of equations:

$$\mathbf{x} = \sum_k c[k] \mathbf{u}_k = \sum_k \langle \mathbf{x}, \mathbf{u}_k \rangle \mathbf{u}_k \tag{2.87}$$

$$(2.88)$$

$$c[k] = \langle \mathbf{x}, \mathbf{u}_k \rangle, \quad \text{for all } k \tag{2.89}$$

The first equation is the *generalized Fourier expansion*, which represents a given vector \mathbf{x} as a linear combination of the basis $\{\mathbf{u}_k\}$, and the weighting coefficient $c[k]$ given in the second equation is the *generalized Fourier coefficient*.

The results above can be generalized to a vector space spanned by a basis composed of a continuum of uncountable orthogonal basis vectors $\mathbf{u}(f)$ satisfying:

$$\langle \mathbf{u}(f), \mathbf{u}(f') \rangle = \delta(f - f') \tag{2.90}$$

Under this basis, any vector \mathbf{x} in the space can be expressed as:

$$\mathbf{x} = \int c(f) \mathbf{u}(f) df \tag{2.91}$$

Same as Eq.2.60, this equation also represents a given vector \mathbf{x} in the space as a linear combination (an integral) of the basis function $\mathbf{u}(f)$, weighted by $c(f)$. However, different from the case in Eq.2.60, here the weighting function $c(f)$ can be easily obtained due to the orthogonality of the basis $\mathbf{u}(f)$. Taking the inner product with $\mathbf{u}(f')$ on both sides of Eq.2.91, we get:

$$\begin{aligned}
 \langle \mathbf{x}, \mathbf{u}(f') \rangle &= \left\langle \int c(f) \mathbf{u}(f) df, \mathbf{u}(f') \right\rangle = \int c(f) \langle \mathbf{u}(f), \mathbf{u}(f') \rangle df \\
 &= \int c(f) \delta(f - f') df = c(f')
 \end{aligned} \tag{2.92}$$

We therefore have

$$c(f) = \langle \mathbf{x}, \mathbf{u}(f) \rangle \tag{2.93}$$

representing the projection of \mathbf{x} onto the unit basis vector $\mathbf{u}(f)$. Now Eq.2.91 can also be written as:

$$\mathbf{x} = \int c(f)\mathbf{u}(f)df = \int \langle \mathbf{x}, \mathbf{u}(f) \rangle \mathbf{u}(f)df \quad (2.94)$$

Also, based on Eq.2.91, we can easily show that Parseval's identity holds:

$$\|\mathbf{x}\|^2 = \langle \mathbf{x}, \mathbf{x} \rangle = \int c(f)\bar{c}(f)df = \langle c(f), c(f) \rangle = \|c(f)\|^2 \quad (2.95)$$

As a specific example, space \mathbb{C}^N can be spanned by N orthonormal vectors $\{\mathbf{u}_1, \dots, \mathbf{u}_N\}$, where the k th basis vector is $\mathbf{u}_k = [u[1, k], \dots, u[N, k]]^T$, that satisfy:

$$\langle \mathbf{u}_k, \mathbf{u}_l \rangle = \mathbf{u}_k^T \bar{\mathbf{u}}_l = \sum_{n=1}^N u[n, k] \bar{u}[n, l] = \delta[k - l] \quad (2.96)$$

Any vector $\mathbf{x} \in \mathbb{C}^N$ can be expressed as:

$$\mathbf{x} = \sum_{k=1}^N c[k]\mathbf{u}_k = [\mathbf{u}_1, \dots, \mathbf{u}_N] \begin{bmatrix} c[1] \\ \vdots \\ c[N] \end{bmatrix} = \mathbf{U}\mathbf{c} \quad (2.97)$$

where $\mathbf{c} = [c[1], \dots, c[N]]^T$ and

$$\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_N] = \begin{bmatrix} u[1, 1] & \cdots & u[1, N] \\ \vdots & \ddots & \vdots \\ u[N, 1] & \cdots & u[N, N] \end{bmatrix} \quad (2.98)$$

As the column (and row) vectors in \mathbf{U} are orthogonal, it is a unitary matrix that satisfies $\mathbf{U}^{-1} = \mathbf{U}^*$, i.e., $\mathbf{U}\mathbf{U}^* = \mathbf{U}^*\mathbf{U} = \mathbf{I}$ (Eq.12.51). To find the coefficient vector \mathbf{c} , we pre-multiply $\mathbf{U}^{-1} = \mathbf{U}^*$ on both sides Eq.2.97 and get:

$$\mathbf{U}^*\mathbf{x} = \mathbf{U}^{-1}\mathbf{x} = \mathbf{U}^{-1}\mathbf{U}\mathbf{c} = \mathbf{c} \quad (2.99)$$

Eqs. 2.97 and 2.99 can be rewritten as a pair of transforms:

$$\begin{cases} \mathbf{c} = \mathbf{U}^*\mathbf{x} = \mathbf{U}^{-1}\mathbf{x} \\ \mathbf{x} = \mathbf{U}\mathbf{c} \end{cases} \quad (2.100)$$

We see that the norm of \mathbf{x} is conserved (Parseval's identity):

$$\|\mathbf{x}\|^2 = \langle \mathbf{x}, \mathbf{x} \rangle = \langle \mathbf{U}\mathbf{c}, \mathbf{U}\mathbf{c} \rangle = (\mathbf{U}\mathbf{c})^*\mathbf{U}\mathbf{c} = \mathbf{c}^*\mathbf{U}^*\mathbf{U}\mathbf{c} = \mathbf{c}^*\mathbf{c} = \langle \mathbf{c}, \mathbf{c} \rangle = \|\mathbf{c}\|^2 \quad (2.101)$$

Equivalently, the coefficient $c[k]$ can also be found by an inner product with \mathbf{u}_l on both sides of Eq.2.97:

$$\langle \mathbf{x}, \mathbf{u}_l \rangle = \langle \sum_{k=1}^N c[k]\mathbf{u}_k, \mathbf{u}_l \rangle = \sum_{k=1}^N c[k] \langle \mathbf{u}_k, \mathbf{u}_l \rangle = \sum_{k=1}^N c[k]\delta[k - l] = c[l] \quad (2.102)$$

Now the transform pair above can also be written as:

$$c[k] = \langle \mathbf{x}, \mathbf{u}_k \rangle = \sum_{n=1}^N x[n] \bar{u}[n, k], \quad (k = 1, \dots, N) \quad (2.103)$$

$$\mathbf{x} = \sum_{k=1}^N c[k] \mathbf{u}_k = \sum_{k=1}^N \langle \mathbf{x}, \mathbf{u}_k \rangle \mathbf{u}_k \quad (2.104)$$

The second equation can also be written in component form as

$$x[n] = \sum_{k=1}^N c[k] u[k, n], \quad (n = 1, \dots, N) \quad (2.105)$$

Obviously the N coefficients $c[k]$ ($k = 1, \dots, N$) can be obtained with computational complexity $O(N^2)$, in comparison with the complexity $O(N^3)$ needed to find \mathbf{U}^{-1} in Eq.2.62 when non-orthogonal basis \mathbf{b}_k is used.

Consider another example of \mathcal{L}^2 space composed of all square integrable functions defined over $a < t < b$, spanned by a set of orthonormal basis functions $\phi_k(t)$ satisfying:

$$\langle \phi_k(t), \phi_l(t) \rangle = \int_a^b \phi_k(t) \bar{\phi}_l(t) dt = \delta[k - l] \quad (2.106)$$

Any $x(t)$ in the space can be written as

$$x(t) = \sum_k c[k] \phi_k(t) \quad (2.107)$$

Taking an inner product with $\phi_l(t)$ on both sides, we get:

$$\langle x(t), \phi_l(t) \rangle = \sum_k c[k] \langle \phi_k(t), \phi_l(t) \rangle = \sum_k c[k] \delta[k - l] = c[l] \quad (2.108)$$

i.e.,

$$c[l] = \langle x(t), \phi_l(t) \rangle = \int_a^b x(t) \bar{\phi}_l(t) dt \quad (2.109)$$

which is the projection of $x(t)$ onto the unit basis function $\phi_k(t)$. Again we can easily get:

$$\|x(t)\|^2 = \langle x(t), x(t) \rangle = \int_a^b x(t) \bar{x}(t) dt = \sum_k |c[k]|^2 = \|\mathbf{c}\|^2 \quad (2.110)$$

Since orthogonal bases are more advantageous than non-orthogonal ones, it is often desirable to convert a given non-orthogonal basis $\{\mathbf{a}_1, \dots, \mathbf{a}_N\}$ into an orthogonal one $\{\mathbf{u}_1, \dots, \mathbf{u}_N\}$ by the following *Gram-Schmidt orthogonalization process*:

- $\mathbf{u}_1 = \mathbf{a}_1$
- $\mathbf{u}_2 = \mathbf{a}_2 - P_{\mathbf{u}_1} \mathbf{a}_2$
- $\mathbf{u}_3 = \mathbf{a}_3 - P_{\mathbf{u}_1} \mathbf{a}_3 - P_{\mathbf{u}_2} \mathbf{a}_3$

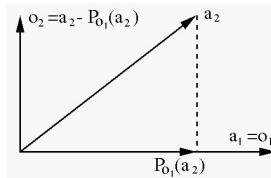


Figure 2.4 Gram-Schmidt orthogonalization

-
- $\mathbf{u}_N = \mathbf{a}_N - \sum_{n=1}^{N-1} P_{\mathbf{u}_n} \mathbf{a}_N$

Example 2.4: In Example 2.2, a vector $\mathbf{x} = [1, 2]^T$ in a 2-D space is represented under a basis composed of $\mathbf{a}_1 = [1, 0]^T$ and $\mathbf{a}_2 = [-1, 2]^T$. Now we show that based on this basis an orthogonal basis can be constructed by the Gram-Schmidt orthogonalization process. In this case of $n = 2$, we have $\mathbf{u}_1 = \mathbf{a}_1 = [1, 0]^T$, $P_{\mathbf{u}_1} \mathbf{a}_2 = [-1, 0]^T$, and

$$\mathbf{u}_2 = \mathbf{a}_2 - P_{\mathbf{u}_1} \mathbf{a}_2 = \begin{bmatrix} -1 \\ 2 \end{bmatrix} - \begin{bmatrix} -1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \end{bmatrix} \quad (2.111)$$

We see that the new basis $\{\mathbf{u}_1, \mathbf{u}_2\}$ is indeed orthogonal as $\langle \mathbf{u}_1, \mathbf{u}_2 \rangle = 0$. Now the same vector $\mathbf{x} = [1, 2]^T$ can be represented by the new orthogonal basis as:

$$\mathbf{x} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} = 1\mathbf{u}_1 + 1\mathbf{u}_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 2 \end{bmatrix} \quad (2.112)$$

In this particular case, both coefficients $c[1] = c[2] = 1$ happen to be 1, as illustrated in Fig.2.4.

2.1.5 Signal Representation by Standard Bases

Here we consider, as a special case of the orthogonal bases, the standard basis in the N-D space \mathbb{R}^N . When $N = 3$, a vector $\mathbf{v} = [x, y, z]^T$ is conventionally represented as:

$$\mathbf{v} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = x\mathbf{i} + y\mathbf{j} + z\mathbf{k} = x \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + y \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + z \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (2.113)$$

where $\mathbf{i} = [1, 0, 0]^T$, $\mathbf{j} = [0, 1, 0]^T$, and $\mathbf{k} = [0, 0, 1]^T$ are the three standard (or canonical) basis vectors along each of the three mutually perpendicular axes. This standard basis $\{\mathbf{i}, \mathbf{j}, \mathbf{k}\}$ in \mathbb{R}^3 can be generalized to \mathbb{R}^N spanned by a set of

N standard basis vectors defined as:

$$\mathbf{e}_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \mathbf{e}_2 = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \quad \dots, \quad \mathbf{e}_N = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad (2.114)$$

All components of the n th standard basis vector \mathbf{e}_n are zero except the n th one which is 1, i.e., the m th component of the n th vector \mathbf{e}_n is $e[m, n] = \delta[m - n]$. These standard basis vectors are indeed orthogonal as $\langle \mathbf{e}_m, \mathbf{e}_n \rangle = \delta[m - n]$ ($m, n = 1, \dots, N$), and they form an identity matrix $\mathbf{I} = [\mathbf{e}_1, \dots, \mathbf{e}_N]$, which is a special unitary matrix satisfying $\mathbf{I}^* = \mathbf{I}^{-1} = \mathbf{I}^T = \mathbf{I}$.

Given this standard basis in \mathbb{R}^N , a vector $\mathbf{x} = [x[1], \dots, x[N]]^T$ representing N samples of a time signal can be expressed as a linear combination of the N standard basis vectors:

$$\mathbf{x} = \sum_{n=1}^N x[n] \mathbf{e}_n = [\mathbf{e}_1, \dots, \mathbf{e}_N] \mathbf{x} = \mathbf{Ix} \quad (2.115)$$

and the m th component $x[m]$ of \mathbf{x} is:

$$x[m] = \sum_{n=1}^N x[n] e[m, n] = \sum_{n=1}^N x[n] \delta[m - n], \quad (m = 1, \dots, N) \quad (2.116)$$

Comparing this equation with Eq.1.3 in the previous chapter we see that they are actually in exactly the same form (except here the signal \mathbf{x} has a finite number of N samples), indicating the fact that whenever a discrete time signal is given in the form of a vector $\mathbf{x} = [x[1], \dots, x[N]]^T$, it is represented implicitly by the standard basis, i.e., the signal is decomposed in time in terms of a set of components $x[m]$ each corresponding to a particular time segment $\delta[m - n]$ at $n = m$. However, while it may seem only natural and reasonable to decompose a signal into a set of time samples, or equivalently, to represent the signal vector by the standard basis, it is also possible, and sometime more beneficial, to decompose the signal into a set of components along some dimension other than time, or equivalently, to represent the signal vector by an orthogonal basis which can be obtained by rotating the standard basis. This is an important point which is to be emphasized through out the book.

The concept of representing a discrete time signal $x[n]$ by the standard basis can be extended to the representation of a continuous time signal $x(t)$ ($0 < t < T$). We first recall the unit square impulse function defined in Eq.1.4:

$$\delta_\Delta(t) = \begin{cases} 1/\Delta & 0 \leq t < \Delta \\ 0 & \text{else} \end{cases} \quad (2.117)$$

based on which a set of basis functions $e_n(t) = \delta_\Delta(t - n\Delta)$ ($n = 0, \dots, N - 1$) can be obtained by a translation of $n\Delta$ in time. These basis functions are obvi-

ously orthonormal:

$$\langle e_m(t), e_n(t) \rangle = \int_0^T \delta_\Delta(t - m\Delta) \delta_\Delta(t - n\Delta) dt = \delta[m - n] \quad (2.118)$$

Next, we sample the continuous time signal $x(t)$ with a sampling interval $\Delta = T/N$ to get a set of discrete samples $\{x[0], \dots, x[N - 1]\}$, and approximate the signal as:

$$x(t) \approx \tilde{x}(t) = \sum_{n=0}^{N-1} x[n]e_n(t) = \sum_{n=0}^{N-1} x[n]\delta_\Delta(t - n\Delta) \Delta \quad (2.119)$$

Here $x[n]e_n(t)$ represents the nth segment of the signal over the time duration $n\Delta < t < (n + 1)\Delta$, as illustrated in Fig.2.5. We see that each of these functions $e_n(t) = \delta_\Delta(t - n\Delta)$ represents a certain time segment, same as the standard basis $e[m, n] = \delta[m - n]$ in \mathbb{C}^N . Note, however, these $\delta_\Delta(t - n\Delta)$ do not form a basis that spans the function space \mathcal{L}^2 , as they are not complete, in the sense that they can only approximate but not precisely represent a continuous function $x(t) \in \mathcal{L}^2$. This shortcoming can be overcome if we continuously reduce the sampling interval Δ to get the Dirac delta at the limit $\Delta \rightarrow 0$:

$$\lim_{\Delta \rightarrow 0} \delta_\Delta(t) = \delta(t) \quad (2.120)$$

Now the summation in Eq.2.119 above becomes an integral, by which the function $x(t)$ can be precisely represented:

$$\lim_{\Delta \rightarrow 0} \tilde{x}(t) = \int x(\tau)\delta(t - \tau)d\tau = x(t) \quad (2.121)$$

This equation is actually the same as Eq. 1.9 in the previous chapter. Now we have obtained a continuum of uncountable basis functions $e_\tau(t) = \delta(t - \tau)$ (for all τ), which are complete as well as orthonormal, i.e., they form a standard basis of the function space \mathcal{L}^2 , by which any continuous signal $x(t)$ can be represented, just as the standard basis e_n in \mathbb{C}^N by which any discrete signal $x[n]$ can be represented.

Again it may seem only natural to represent a continuous time signal $x(t)$ by the corresponding standard basis representing a sequence of time impulses $x(\tau)\delta(t - \tau)$. However, this is not the only way or the best way to represent the signal. The time signal can also be represented by a basis other than the standard basis $\delta(t - \tau)$, so that the signal is decomposed along some different dimension other than time. Such an alternative way of signal decomposition and representation may be desirable, as the signal can be more conveniently processed and analyzed, for whatever purpose of the signal processing task. This is actually the fundamental reason why different orthogonal transforms are developed, as to be discussed in details in future chapters.

Fig.2.6 illustrates the idea that any given vector \mathbf{x} can be equivalently represented under different bases each corresponding to a different set of coefficients,

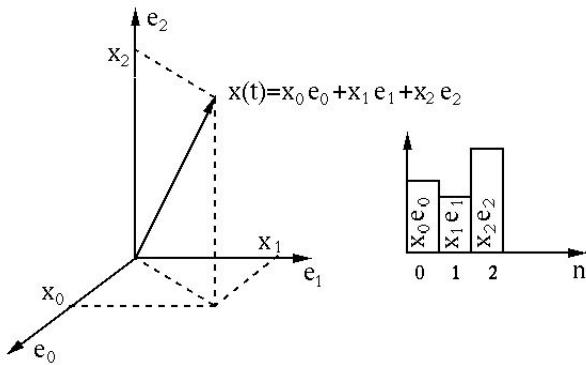


Figure 2.5 Vector representation of an N-D space ($N=3$)

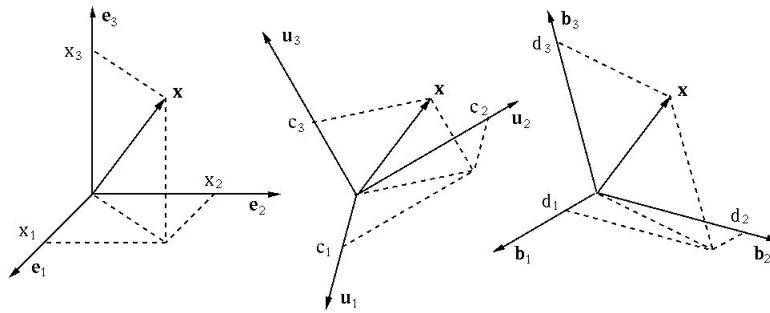


Figure 2.6 Representations of the same vector x under different bases— standard basis e_k (left), an unitary (orthogonal) basis u_k (middle), and a non-orthogonal basis b_k (right)

such as the standard basis, an orthogonal basis (any rotated version of the standard basis), or an arbitrary basis not necessarily orthogonal at all. While non-orthogonal axes are never actually used, one always has many options in terms of what orthogonal basis to use.

2.1.6 An Example: the Fourier Transforms

To illustrate how a vector can be represented by an orthogonal basis that spans the space, we consider the following four Fourier bases that span four different types of vector spaces for signals that are either continuous or discrete, of finite or infinite duration.

- $\mathbf{u}_k = [e^{j2\pi k 0/N}, \dots, e^{j2\pi k(N-1)/N}]^T / \sqrt{N}$ ($k = 0, \dots, N-1$) form a set of N orthonormal basis vectors that span \mathbb{C}^N (Eq.1.40):

$$\langle \mathbf{u}_k, \mathbf{u}_l \rangle = \frac{1}{N} \sum_{n=0}^{N-1} e^{j2\pi(k-l)n/N} = \delta[k-l] \quad (2.122)$$

Any vector $\mathbf{x} = [x[0], \dots, x[N-1]]^T$ in \mathbb{C}^N can be expressed as:

$$\mathbf{x} = \sum_{k=0}^{N-1} X[k] \mathbf{u}_k = \sum_{k=0}^{N-1} \langle \mathbf{x}, \mathbf{u}_k \rangle \mathbf{u}_k \quad (2.123)$$

or in component form:

$$x[n] = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X[k] e^{j2\pi kn/N}, \quad (0 \leq n \leq N-1) \quad (2.124)$$

where the coefficient $X[k]$ is the projection of \mathbf{x} onto \mathbf{u}_k :

$$X[k] = \langle \mathbf{x}, \mathbf{u}_k \rangle = \sum_{n=0}^{N-1} x[n] \bar{u}[n, k] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x[n] e^{-j2\pi nk/N} \quad (2.125)$$

- $\mathbf{u}(f) = [\dots, e^{j2\pi m f / F}, \dots]^T / \sqrt{F}$ ($0 < f < F$) form a set of uncountably infinite orthonormal basis vectors (of infinite dimensions) (Eq.1.35) that spans l^2 space of all square summable vectors of infinite dimensions:

$$\langle \mathbf{u}_f, \mathbf{u}_{f'} \rangle = \frac{1}{F} \sum_{m=-\infty}^{\infty} e^{j2\pi(f-f')m/F} = \delta(f - f') \quad (2.126)$$

Any vector $\mathbf{x} = [\dots, x[n], \dots]^T$ in this space can be expressed as:

$$\mathbf{x} = \int_{-\infty}^{\infty} X(f) \mathbf{u}(f) df = \int_{-\infty}^{\infty} \langle \mathbf{x}, \mathbf{u}(f) \rangle \mathbf{u}(f) df \quad (2.127)$$

or in component form:

$$x[n] = \frac{1}{\sqrt{F}} \int_{-\infty}^{\infty} X(f) e^{j2\pi fn/F} df, \quad (-\infty < n < \infty) \quad (2.128)$$

where the coefficient function $X(f)$ is the projection of \mathbf{x} onto $\mathbf{u}(f)$:

$$X(f) = \langle \mathbf{x}, \mathbf{u}(f) \rangle = \frac{1}{\sqrt{F}} \sum_{n=-\infty}^{\infty} x[n] e^{-j2\pi fn/F} \quad (2.129)$$

- $u_k(t) = e^{j2\pi kt/T} / \sqrt{T}$ ($-\infty < k < \infty$) form a set of infinite orthonormal basis functions (Eq.1.33) that spans the space of all square integrable functions defined over $0 < t < T$:

$$\langle u_k(t), u_l(t) \rangle = \frac{1}{T} \int_0^T e^{j2\pi(k-l)t/T} dt = \delta[k - l] \quad (2.130)$$

Any function $x_T(t)$ in this space can be expressed as:

$$x_T(t) = \sum_{k=-\infty}^{\infty} X[k] u_k(t) = \frac{1}{\sqrt{T}} \sum_{k=-\infty}^{\infty} X[k] e^{j2\pi kt/T} \quad (2.131)$$

where the coefficient $X[k]$ is the projection of $x(t)$ onto the k th basis function $u_k(t)$:

$$X[k] = \langle x(t), u_k(t) \rangle = \int_{-\infty}^{\infty} x(t) \bar{u}_k(t) dt = \frac{1}{\sqrt{T}} \int_{-\infty}^{\infty} x(t) e^{-j2\pi kt/T} dt \quad (2.132)$$

- $u_f(t) = e^{j2\pi ft}$ ($-\infty < f < \infty$) is a set of uncountably infinite orthonormal basis functions (Eq.1.28) that spans \mathcal{L}^2 space of all square integrable functions defined over $-\infty < t < \infty$.

$$\langle u_f(t), u_{f'}(t) \rangle = \int_{-\infty}^{\infty} e^{j2\pi(f-f')t} dt = \delta(f - f') \quad (2.133)$$

Any function $x(t)$ in this space can be expressed as:

$$x(t) = \int_{-\infty}^{\infty} X(f) u_f(t) df = \int_{-\infty}^{\infty} X(f) e^{j2\pi ft} df \quad (2.134)$$

where the coefficient function is the projection of $x(t)$ onto $u_f(t)$:

$$X(f) = \langle x(t), u_f(t) \rangle = \int_{-\infty}^{\infty} x(t) \bar{u}_f(t) dt = \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt \quad (2.135)$$

2.2 Unitary Transformation and Signal Representation

2.2.1 Linear Transformation

Definition 2.19. • Let V and W be two vector spaces. A transformation is a function or mapping $T : V \rightarrow W$ that converts a vector $\mathbf{x} \in V$ to another vector $\mathbf{u} \in W$ denoted by: $T\mathbf{x} = \mathbf{u}$. If $W = V$, the linear transformation T is a linear operator.

- If the transformation is invertible, i.e., then a transformation that converts $\mathbf{u} \in W$ back to $\mathbf{x} \in V$ is an inverse transformation denoted by: $\mathbf{x} = T^{-1}\mathbf{u}$.
- An identity transformation maps a vector to itself: $I\mathbf{x} = \mathbf{x}$. Obviously $TT^{-1} = T^{-1}T = I$ is an identity operator that maps a vector to itself:

$$\begin{aligned} TT^{-1}\mathbf{u} &= T(T^{-1}\mathbf{u}) = T\mathbf{x} = \mathbf{u} = I\mathbf{u} \\ T^{-1}T\mathbf{x} &= T^{-1}(T\mathbf{x}) = T^{-1}\mathbf{u} = \mathbf{x} = I\mathbf{x} \end{aligned} \quad (2.136)$$

- A transformation T is linear if the following is true:

$$T(a\mathbf{x} + b\mathbf{y}) = aT\mathbf{x} + bT\mathbf{y} \quad (2.137)$$

for any scalars $a, b \in \mathbb{C}$ and any vectors $\mathbf{x}, \mathbf{y} \in V$.

For example, the derivative and integral of a continuous function $x(t)$ are linear operators:

$$T_d x(t) = \frac{d}{dt} x(t) = \dot{x}(t), \quad T_i x(t) = \int x(\tau) d\tau \quad (2.138)$$

For another example, an M by N matrix \mathbf{A} with its mn-th element being $a[m, n] \in \mathbb{C}$ is a linear transformation $T_A : \mathbb{C}^N \rightarrow \mathbb{C}^M$ that maps an N-D vector $\mathbf{x} \in \mathbb{C}^N$ to an M-D vector $\mathbf{y} \in \mathbb{C}^M$:

$$T_A \mathbf{x} = \mathbf{A} \mathbf{x} = \mathbf{y} \quad (2.139)$$

or in component form:

$$\begin{bmatrix} y[1] \\ y[2] \\ \vdots \\ y[M] \end{bmatrix}_{M \times 1} = \begin{bmatrix} a[1, 1] & a[1, 2] & \cdots & a[1, N] \\ a[2, 1] & a[2, 2] & \cdots & a[2, N] \\ \vdots & \vdots & \ddots & \vdots \\ a[M, 1] & a[M, 2] & \cdots & a[M, N] \end{bmatrix}_{M \times N} \begin{bmatrix} x[1] \\ x[2] \\ \vdots \\ x[n] \end{bmatrix}_{N \times 1} \quad (2.140)$$

If $M = N$, then $\mathbf{x}, \mathbf{y} \in \mathbb{C}^N$ and \mathbf{A} becomes a linear operator.

However, note that the operation of translation $T_t \mathbf{x} = \mathbf{x} + \mathbf{t}$ is not a linear transformation:

$$T_t(a\mathbf{x} + b\mathbf{y}) = a\mathbf{x} + b\mathbf{y} + \mathbf{t} \neq aT_t\mathbf{x} + bT_t\mathbf{y} = a\mathbf{x} + b\mathbf{y} + (a + b)\mathbf{t} \quad (2.141)$$

Definition 2.20. • For a linear transformation $T : V \rightarrow W$, if there is another transformation $T^* : W \rightarrow V$ so that

$$\langle T\mathbf{x}, \mathbf{u} \rangle = \langle \mathbf{x}, T^*\mathbf{u} \rangle \quad (2.142)$$

for any $\mathbf{x} \in V$ and $\mathbf{u} \in W$, the T^* is called the Hermitian adjoint or simply adjoint of T .

- If a linear operator $T : V \rightarrow V$ is its own adjoint, i.e.,

$$\langle T\mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{x}, T\mathbf{y} \rangle \quad (2.143)$$

for any $\mathbf{x}, \mathbf{y} \in V$, then T is called a self-adjoint or Hermitian transformation.

In the following, the terms “self-adjoint” and “Hermitian” are used interchangeably.

In particular, in the unitary space \mathbb{C}^N , let $\mathbf{B} = \mathbf{A}^*$ be the adjoint of matrix \mathbf{A} , i.e., $\langle \mathbf{A}\mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{B}\mathbf{y} \rangle$, then we have:

$$\langle \mathbf{A}\mathbf{x}, \mathbf{y} \rangle = (\mathbf{A}\mathbf{x})^T \overline{\mathbf{y}} = \mathbf{x}^T \mathbf{A}^T \overline{\mathbf{y}} = \langle \mathbf{x}, \mathbf{B}\mathbf{y} \rangle = \mathbf{x}^T \overline{\mathbf{B}} \overline{\mathbf{y}} \quad (2.144)$$

Comparing the two sides we get $\mathbf{A}^T = \overline{\mathbf{B}}$, i.e., the adjoint matrix $\mathbf{B} = \mathbf{A}^* = \overline{\mathbf{A}}^T$ is the conjugate transpose of \mathbf{A} :

$$\mathbf{A}^* = \overline{\mathbf{A}}^T \quad (2.145)$$

A matrix \mathbf{A} is self-adjoint, or *Hermitian* if $\mathbf{A} = \mathbf{A}^* = \overline{\mathbf{A}}^T$, i.e.,

$$\langle \mathbf{A}\mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{A}\mathbf{y} \rangle \quad (2.146)$$

In particular, when $\overline{\mathbf{A}} = \mathbf{A}$ is real, a self-adjoint matrix $\mathbf{A} = \mathbf{A}^* = \mathbf{A}^T$ is symmetric. Note that we have always used \mathbf{A}^* to denote the conjugate transpose of

a matrix \mathbf{A} , which we now see is also the self-adjoint of \mathbf{A} , and the notation T^* is more generally used to denote the self-adjoint of any operator T .

In a function space, if T^* is the adjoint of a linear operator T , then the following holds:

$$\langle Tx(t), y(t) \rangle = \int Tx(t) \overline{y(t)} dt = \langle x(t), T^*y(t) \rangle = \int x(t) \overline{T^*y(t)} dt \quad (2.147)$$

If $T = T^*$, it is a self-adjoint or Hermitian operator.

2.2.2 Eigenvalue problems

Definition 2.21. If the application of an operator T to a vector $\mathbf{x} \in V$ results in another vector $\lambda\mathbf{x} \in V$, where $\lambda \in \mathbb{C}$ is a constant scalar:

$$T\mathbf{x} = \lambda\mathbf{x} \quad (2.148)$$

then the scalar λ is an eigenvalue of T and vector \mathbf{x} is the corresponding eigenvector or eigenfunctions of T , and the equation above is called the eigenequation of the operator T . The set of all eigenvalues of an operator is called the spectrum of the operator.

In a unitary space \mathbb{C}^N , an N by N matrix \mathbf{A} is a linear operator and the associated eigenequation is:

$$\mathbf{A}\phi_n = \lambda_n\phi_n, \quad n = 1, \dots, N \quad (2.149)$$

where λ_n and ϕ_n are the nth eigenvalue and the corresponding eigenvector of \mathbf{A} , respectively.

In a function space, the nth-order differential operator $D^n = d^n/dt^n$ is a linear operator with the following eigenequation:

$$D^n\phi(t) = D^n e^{st} = \frac{d^n}{dt^n} e^{st} = s^n e^{st} = \lambda\phi(t) \quad (2.150)$$

where s is a complex scalar. Here the $\lambda = s^n$ is the eigenvalue and the complex exponential $\phi(t) = e^{st}$ is the corresponding eigenfunction. More generally, we can write an Nth order *linear constant coefficient differential equation (LCCDE)* as:

$$\sum_{n=0}^N a_n \frac{d^n}{dt^n} y(t) = \left[\sum_{n=0}^N a_n D^n \right] y(t) = x(t) \quad (2.151)$$

where $\sum_{n=0}^N a_n D^n$ is a linear operator that is applied to function $y(t)$, representing the response of a linear system to an input $x(t)$. Obviously the same complex exponential $\phi(t) = e^{st}$ is also the eigenfunction corresponding to the eigenvalue $\lambda = \sum_{k=0}^n a_k s^k$ of this operator.

Perhaps the most well known eigenvalue problem in physics is the Schrodinger equation, which describes a particle in terms of its energy and the De Broglie

wave function. Specifically for a 1-D stationary single particle system, we have:

$$\hat{\mathcal{H}}\psi(x) = \left[-\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + V(x) \right] \psi(x) = \mathcal{E}\psi(x) \quad (2.152)$$

where

$$\hat{\mathcal{H}} = -\frac{\hbar^2}{2m} \frac{\partial^2}{\partial x^2} + V(x) \quad (2.153)$$

is the Hamiltonian operator, \hbar is the Planck constant, m and $V(x)$ are the mass and potential energy of the particle, respectively. \mathcal{E} is the eigenvalue of $\hat{\mathcal{H}}$, representing the total energy of the particle, and the wave function $\psi(x)$ is the corresponding eigenfunction, also called eigenstate, representing probability amplitude of the particle, i.e., $|\psi(x)|^2$ is the probability for the particle to be found at position x .

Theorem 2.4. *A self-adjoint operator has the following properties:*

1. All eigenvalues are real;
2. The eigenvectors corresponding to different eigenvalues are orthogonal;
3. The family of all eigenvectors forms a complete orthogonal system.

Proof: Let λ and μ be two different eigenvalues of a self-adjoint operator T , and \mathbf{x} and \mathbf{y} be the corresponding eigenvectors:

$$T\mathbf{x} = \lambda\mathbf{x}, \quad T\mathbf{y} = \mu\mathbf{y} \quad (2.154)$$

As $T = T^*$ is self-adjoint, we have:

$$\langle T\mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{x}, T\mathbf{y} \rangle \quad (2.155)$$

Substituting $T\mathbf{x} = \lambda\mathbf{x}$ into Eq.2.155 and letting $\mathbf{y} = \mathbf{x}$, we get

$$\langle \lambda\mathbf{x}, \mathbf{x} \rangle = \langle \mathbf{x}, \lambda\mathbf{x} \rangle, \quad \text{i.e. } \lambda \langle \mathbf{x}, \mathbf{x} \rangle = \overline{\lambda} \langle \mathbf{x}, \mathbf{x} \rangle \quad (2.156)$$

As in general $\langle \mathbf{x}, \mathbf{x} \rangle \neq 0$, we see that $\lambda = \overline{\lambda}$ is real. Next, we substitute $T\mathbf{x} = \lambda\mathbf{x}$ and $T\mathbf{y} = \mu\mathbf{y}$ into Eq.2.155 and get:

$$\lambda \langle \mathbf{x}, \mathbf{y} \rangle = \overline{\mu} \langle \mathbf{x}, \mathbf{y} \rangle = \mu \langle \mathbf{x}, \mathbf{y} \rangle \quad (2.157)$$

As in general $\lambda \neq \mu$, we get $\langle \mathbf{x}, \mathbf{y} \rangle = 0$, i.e., \mathbf{x} and \mathbf{y} are orthogonal. The proof of the third property is beyond the scope of the book and is therefore omitted. Q.E.D.

For example, the Hamiltonian operator $\hat{\mathcal{H}}$ in the Schrodinger equation is a self-adjoint operator with real eigenvalues \mathcal{E} representing different energy levels corresponding to different eigenstates of the particle.

The third property in Theorem 2.4 indicates that the eigenvectors of a self-adjoint operator can be used as an orthogonal basis of a vector space, so that any vector in the space can be represented as a linear combination of these eigenvectors.

In space \mathbb{C}^N , let λ_k and ϕ_k ($k = 1, \dots, N$) be the eigenvalues and the corresponding eigenvectors of a Hermitian matrix $\mathbf{A} = \mathbf{A}^*$, then its eigenequation can be written as:

$$\mathbf{A}\phi_k = \lambda_k\phi_k, \quad k = 1, \dots, N, \quad (2.158)$$

We can further combine all N eigenequations to have:

$$\mathbf{A}[\phi_1, \dots, \phi_N] = [\phi_1, \dots, \phi_N]\Lambda, \quad \text{or} \quad \mathbf{A}\Phi = \Phi\Lambda \quad (2.159)$$

where matrices Φ and Λ are defined as:

$$\Phi = [\phi_1, \dots, \phi_N], \quad \Lambda = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_N \end{bmatrix} \quad (2.160)$$

As \mathbf{A} is a self-adjoint operator, its eigenvalues λ_k are real, and their corresponding eigenvectors ϕ_k are orthogonal:

$$\langle \phi_k, \phi_l \rangle = \phi_k^T \overline{\phi_l} = \delta[k - l] \quad (2.161)$$

and they form a complete orthogonal system to span the N-D unitary space. Also Φ is a unitary matrix satisfying:

$$\Phi^* \Phi = \mathbf{I}, \quad \text{or} \quad \Phi^* = \Phi^{-1} \quad (2.162)$$

The eigenequation in Eq.2.159 can also be written in some other useful forms. First, pre-multiplying both sides of the equation by $\Phi^{-1} = \Phi^*$, we get:

$$\Phi^{-1} \mathbf{A} \Phi = \Phi^* \mathbf{A} \Phi = \Lambda \quad (2.163)$$

i.e., the matrix \mathbf{A} can be diagonalized by Φ . Alternatively, if we post-multiply both sides of Eq.2.159 by Φ^* , we get:

$$\mathbf{A} = \Phi \Lambda \Phi^* = [\phi_1, \phi_2, \dots, \phi_N] \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_N \end{bmatrix} \begin{bmatrix} \phi_1^* \\ \phi_2^* \\ \vdots \\ \phi_N^* \end{bmatrix} = \sum_{k=1}^N \lambda_k \phi_k \phi_k^* \quad (2.164)$$

i.e., the matrix \mathbf{A} can be series expanded to become a linear combination of N eigen-matrices $\phi_k \phi_k^*$ ($k = 1, \dots, N$).

2.2.3 Eigenvectors of D^2 as Fourier Basis

Here we consider a particular example of the self-adjoint operators, the second-order differential operator $D^2 = d^2/dt^2$ in \mathcal{L}^2 -space, which is of important significance as its orthogonal eigenfunctions form the basis used in the Fourier transform.

First we show that D^2 is indeed a self-adjoint operator:

$$\langle D^2x(t), y(t) \rangle = \langle x(t), D^2y(t) \rangle \quad (2.165)$$

where $x(t)$ and $y(t)$ are two functions defined over a certain time interval such as $[0, T]$, and $D^2x(t) = \ddot{x}(t)$ is the second time derivative of function $x(t)$. Using integration by parts, we can show that this equation does hold:

$$\begin{aligned} \langle D^2x(t), y(t) \rangle &= \int_0^T \ddot{x}(t)\bar{y}(t)dt = \dot{x}(t)\bar{y}(t) \Big|_0^T - \int_0^T \dot{x}(t)\bar{\dot{y}}(t)dt \\ &= \dot{x}(t)\bar{y}(t) \Big|_0^T - x(t)\bar{\dot{y}}(t) \Big|_0^T + \int_0^T x(t)\bar{\dot{y}}(t)dt = \langle x(t), D^2y(t) \rangle \end{aligned} \quad (2.166)$$

Here we have assumed all functions satisfy $x(0) = x(T)$, $\dot{x}(0) = \dot{x}(T)$, so that

$$[\dot{x}(t)\bar{y}(t) - x(t)\bar{\dot{y}}(t)] \Big|_0^T = 0 \quad (2.167)$$

Next, we find the eigenvalues and eigenfunctions of D^2 by solving this equation:

$$\begin{cases} D^2\phi(t) = \lambda\phi(t), & \text{i.e. } \ddot{\phi}(t) - \lambda\phi(t) = 0 \\ \text{subject to: } \phi(0) = \phi(T), & \dot{\phi}(0) = \dot{\phi}(T) \end{cases} \quad (2.168)$$

Consider the following three cases:

1. $\lambda = 0$:

The equation becomes $\ddot{\phi}(t) = 0$ with solution $\phi(t) = c_1t + c_2$. Substituting this $\phi(t)$ into the boundary conditions, we have:

$$\phi(0) = c_2 = \phi(T) = c_1T + c_2 \quad (2.169)$$

We get $c_1 = 0$ and the eigenfunction $\phi(t) = c_2$ is any constant.

2. $\lambda > 0$:

We assume $\phi(t) = e^{st}$ and substitute it into the equation to get

$$(s^2 - \lambda)e^{st} = 0, \quad \text{i.e. } s = \pm\sqrt{\lambda} \quad (2.170)$$

The solution is $\phi(t) = c e^{\pm\sqrt{\lambda}t}$. Substituting this into the boundary conditions, we have:

$$\phi(0) = c = \phi(T) = c e^{\pm\sqrt{\lambda}T} \quad (2.171)$$

Obviously this equation holds only if $\lambda = 0$, as in the previous case.

3. $\lambda < 0$:

We assume $\lambda = -\omega^2$, i.e., $\sqrt{\lambda} = \pm j\omega$, and the solution is

$$\phi(t) = c e^{\pm\sqrt{\lambda}t} = c e^{\pm j\omega t} \quad (2.172)$$

Substituting this into the boundary conditions we have:

$$\phi(0) = c = \phi(T) = c e^{\pm j\omega T}, \quad \text{i.e. } e^{\pm j\omega T} = 1 \quad (2.173)$$

which can be solved to get:

$$\omega T = 2k\pi, \quad \text{i.e. } \omega = \frac{2k\pi}{T} = 2k\pi f_0 = n\omega_0, \quad (k = 0, \pm 1, \pm 2, \dots) \quad (2.174)$$

where we have defined

$$f_0 = \frac{1}{T}, \quad \omega_0 = 2\pi f_0 = \frac{2\pi}{T}, \quad (2.175)$$

Now the eigenvalues and the corresponding eigenfunctions can be written as:

$$\lambda_k = -\omega^2 = -(k\omega_0)^2 = -(2k\pi f_0)^2 = -(2k\pi/T)^2 \quad (2.176)$$

$$\phi_k(t) = c e^{\pm j k \omega_0 t} = c e^{\pm j 2k\pi f_0 t} = c e^{\pm j 2k\pi/T t} \quad (k = 0, \pm 1, \pm 2, \dots) \quad (2.177)$$

In particular, when $k = 0$, we have $\lambda_k = 0$ and $\phi_0(t) = c$, same as the first case above.

These eigenvalues and their corresponding eigenfunctions have the following properties:

- The eigenvalues are discrete, the gap between two consecutive eigenvalues is:

$$\Delta \lambda_k = \lambda_{k+1} - \lambda_k \quad (2.178)$$

- All eigenfunctions are also discrete with a frequency gap between two consecutive eigenfunctions:

$$\omega_0 = 2\pi f_0 = 2\pi/T \quad (2.179)$$

- All eigenfunctions $\phi_k(t)$ are periodic with period T :

$$\phi_k(t + T) = e^{j2k\pi(t+T)/T} = e^{j2k\pi t/T} e^{j2k\pi} = e^{j2k\pi t/T} = \phi_k(t) \quad (2.180)$$

According to the properties of self-adjoint operators discussed above, the eigenfunctions $\phi_k(t)$ of D^2 form a complete orthogonal system. The orthogonality can be easily verified:

$$\begin{aligned} < \phi_k(t), \phi_l(t) > &= c^2 \int_0^T e^{jk\omega_0 t} e^{-jl\omega_0 t} dt = c^2 \int_0^T e^{j2\pi(k-l)t/T} dt \\ &= c^2 \int_0^T \cos\left(\frac{2\pi(k-l)t}{T}\right) dt + j c^2 \int_0^T \sin\left(\frac{2\pi(k-l)t}{T}\right) dt = \begin{cases} T & k = l \\ 0 & k \neq l \end{cases} \end{aligned} \quad (2.181)$$

If we let $c = 1/\sqrt{T}$, then the eigenfunctions become

$$\phi_k(t) = \frac{1}{\sqrt{T}} e^{j2k\pi t/T} = \frac{1}{\sqrt{T}} e^{j2k\pi f_0 t} \quad (2.182)$$

which are orthonormal:

$$< \phi_k(t), \phi_l(t) > = \frac{1}{T} \int_0^T e^{j2\pi(k-l)t/T} dt = \delta[k - l] \quad (2.183)$$

This is actually Eq.2.130. As a complete orthogonal system, these orthogonal eigenfunctions form a basis to span the function space over $[0, T]$, i.e., all periodic functions $x_T(t) = x_T(t + T)$ can be represented as a linear combination of these basis functions:

$$x_T(t) = \sum_{k=-\infty}^{\infty} X[k] \phi_k(t) = \sum_{k=-\infty}^{\infty} X[k] e^{j2k\pi f_0 t} = \sum_{k=-\infty}^{\infty} X[k] e^{jk\omega_0 t} \quad (2.184)$$

where $X[k]$ ($k = 0, \pm 1, \pm 2, \dots$) are the coefficients given in Eq.2.132. This is the Fourier expansion, to be discussed in detail in the next chapter.

The expansion of a non-periodic function can be similarly obtained if we let $T \rightarrow \infty$ so that at the limit a periodic function $x_T(t)$ becomes non-periodic, and the following will take place:

- The discrete variables $k\omega_0 = 2k\pi/T$ ($k = 0, \pm 1, \pm 2, \dots$) becomes a continuous variable $-\infty < \omega < \infty$;
- The gap between two consecutive eigenvalues becomes zero, i.e., $\Delta\lambda_k \rightarrow 0$, the discrete eigenvalues $\lambda_k = -(2k\pi/T)^2$ become a continuous eigenvalue function $\lambda = -\omega^2$;
- The frequency gap ω_0 between two consecutive eigenfunctions becomes zero, the discrete eigenfunctions $\phi_k(t) = e^{j2k\pi t/T}$ ($k = 0, \pm 1, \pm 2, \dots$) become a set of uncountable non-periodic eigenfunctions $\phi_f(t) = e^{j2\pi ft}$ for all $-\infty < f < \infty$.

We see that the same self-adjoint operator D^2 is now defined over a different interval $(-\infty, \infty)$ and correspondingly its eigenfunctions $\phi(t) = e^{j\omega t} = e^{j2\pi ft} = \phi(t, f)$ become a continuous function of f as well as t and they form a complete orthogonal system spanning the function space of all non-periodic functions:

$$\langle \phi_f(t), \phi_{f'}(t) \rangle = \int_{-\infty}^{\infty} e^{j2\pi(f-f')t} dt = \delta(f - f') \quad (2.185)$$

This is actually Eq.2.133. Now $\phi_f(t)$ becomes a set of uncountably infinite basis functions and any non-periodic square integrable function $x(t)$ can be represented as:

$$x(t) = \int_{-\infty}^{\infty} X(f) \phi_f(t) df = \int_{-\infty}^{\infty} X(f) e^{j2\pi ft} df \quad (2.186)$$

where $X(f)$ is the weighting function given in Eq.2.135. This is the Fourier transform, to be discussed in detail in the next chapter.

2.2.4 Unitary Transformations

Definition 2.22. A linear transformation $U : V \rightarrow W$ is a unitary transformation if it conserves inner products:

$$\langle \mathbf{x}, \mathbf{y} \rangle = \langle U\mathbf{x}, U\mathbf{y} \rangle \quad (2.187)$$

In particular, if the vectors are real with symmetric inner product $\langle \mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{y}, \mathbf{x} \rangle$, then U is an orthogonal transformation.

Obviously a unitary transformation also conserves any measurement based on the inner product, such as the norm of a vector, the distance and angle between two vectors, and the projection of one vector on another. Also, if in particular $\mathbf{x} = \mathbf{y}$, we have

$$\langle \mathbf{x}, \mathbf{x} \rangle = \|\mathbf{x}\|^2 = \langle U\mathbf{x}, U\mathbf{x} \rangle = \|U\mathbf{x}\|^2 \quad (2.188)$$

i.e., the unitary transformation conserves vector norm (length). This is Parseval's identity for a generic unitary transformation $U\mathbf{x}$. Due to this property, a unitary operation $R : V \rightarrow V$ can be intuitively interpreted as a rotation in space V .¹

Theorem 2.5. A linear transformation U is unitary if and only if its adjoint U^* is equal to its inverse U^{-1} :

$$U^* = U^{-1}, \quad \text{i.e.} \quad U^*U = UU^* = I \quad (2.189)$$

Proof: We let $U\mathbf{y} = \mathbf{d}$, i.e., $\mathbf{y} = U^{-1}\mathbf{d}$ in Eq.2.187, and get

$$\langle U\mathbf{x}, \mathbf{d} \rangle = \langle \mathbf{x}, U^{-1}\mathbf{d} \rangle = \langle \mathbf{x}, U^*\mathbf{d} \rangle \quad (2.190)$$

i.e., $U^{-1} = U^*$. Q.E.D.

Due to this theorem, Eq.2.189 can be used as an alternative definition for the unitary operator.

In the generalized Fourier expansion in Eqs.2.88 and 2.89 based on the Plancherel Theorem (Thm.2.3), the coefficient vector $\mathbf{c} = [\dots, c[k], \dots]^T$ composed of $c[k] = \langle \mathbf{x}, \mathbf{u}_k \rangle$ can be considered as a transformation $\mathbf{c} = U\mathbf{x}$, and we can also have another transformation $\mathbf{d} = U\mathbf{y}$. Now we have (Eq.2.85):

$$\langle \mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{c}, \mathbf{d} \rangle = \langle U\mathbf{x}, U\mathbf{y} \rangle \quad (2.191)$$

indicating that the inner product is conserved by U , i.e., the generalized Fourier expansion $\mathbf{c} = U\mathbf{x}$ is actually a unitary transformation.

When a unitary operator U is applied to an orthonormal basis $\{\mathbf{u}_k\}$, the basis is rotated to become another orthonormal basis $\{\mathbf{v}_k = U\mathbf{u}_k\}$ that spans the same space:

$$\langle \mathbf{v}_k, \mathbf{v}_l \rangle = \langle U\mathbf{u}_k, U\mathbf{u}_l \rangle = \langle \mathbf{u}_k, \mathbf{u}_l \rangle = \delta[k - l] \quad (2.192)$$

Specially, when a unitary operator U is applied to the standard basis $\{\mathbf{e}_k\}$, this basis is rotated to become a unitary basis $\{\mathbf{u}_k = U\mathbf{e}_k\}$.

¹ Strictly speaking, a unitary transformation may also correspond to other norm-preserving operations such as reflection and inversion, which could be all treated as rotations in the most general sense.

2.2.5 Unitary Transformations in N-D Space

We consider specifically the unitary transformation in the N-D unitary space \mathbb{C}^N .

Definition 2.23. A matrix \mathbf{U} is unitary if it conserves inner products:

$$\langle \mathbf{U}\mathbf{x}, \mathbf{U}\mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{y} \rangle \quad (2.193)$$

Theorem 2.6. A matrix \mathbf{U} is unitary if and only if $\mathbf{U}^*\mathbf{U} = \mathbf{I}$, i.e., the following two statements are equivalent:

$$(a) \quad \langle \mathbf{U}\mathbf{x}, \mathbf{U}\mathbf{y} \rangle = \langle \mathbf{x}, \mathbf{y} \rangle \quad (2.194)$$

$$(b) \quad \mathbf{U}^*\mathbf{U} = \mathbf{U}\mathbf{U}^* = \mathbf{I}, \quad \text{i.e.,} \quad \mathbf{U}^{-1} = \mathbf{U}^* \quad (2.195)$$

Proof: We first show if (b) then (a):

$$\langle \mathbf{U}\mathbf{x}, \mathbf{U}\mathbf{y} \rangle = (\mathbf{U}\mathbf{x})^T \overline{\mathbf{U}\mathbf{y}} = \mathbf{x}^T \mathbf{U}^T \overline{\mathbf{U}\mathbf{y}} = \mathbf{x}^T \mathbf{I} \overline{\mathbf{y}} = \langle \mathbf{x}, \mathbf{y} \rangle \quad (2.196)$$

Next we show if (a) then (b). (a) can be written as:

$$(\mathbf{U}\mathbf{x})^* \mathbf{U}\mathbf{x} = \mathbf{x}^* \mathbf{U}^* \mathbf{U}\mathbf{x} = \mathbf{x}^* \mathbf{x} \quad (2.197)$$

i.e.,

$$\mathbf{x}^* (\mathbf{U}^* \mathbf{U} - \mathbf{I}) \mathbf{x} = 0 \quad (2.198)$$

Since in general $\mathbf{x} \neq 0$, we must have $\mathbf{U}^* \mathbf{U} = \mathbf{I}$. Post-multiplying this equation by \mathbf{U}^{-1} , we get $\mathbf{U}^* = \mathbf{U}^{-1}$. Pre-multiplying this new equation by \mathbf{U} , we get $\mathbf{U}\mathbf{U}^* = \mathbf{I}$. Q.E.D.

As (a) and (b) in the theorem above are equivalent, either of them can be used as the definition of a unitary matrix. If a unitary matrix $\overline{\mathbf{U}} = \mathbf{U}$ is real, i.e., $\mathbf{U}^{-1} = \mathbf{U}^T$, then it is called an *orthogonal matrix*.

A unitary matrix \mathbf{U} has the following properties:

- Unitary transformation $\mathbf{U}\mathbf{x}$ conserves vector norm, i.e., $\|\mathbf{U}\mathbf{x}\| = \|\mathbf{x}\|$ for any $\mathbf{x} \in \mathbb{C}^N$;
- All eigenvalues $\{\lambda_1, \dots, \lambda_N\}$ of \mathbf{U} have an absolute value of 1: $|\lambda_k| = 1$, i.e., they lie on the unit circle in the complex plain.
- The determinant of \mathbf{U} has an absolute value of 1: $|\det(\mathbf{U})| = 1$. This can be easily seen as $\det(\mathbf{U}) = \prod_{k=1}^N \lambda_k$.
- All column (or row) vectors of $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_N]$ are orthonormal:

$$\langle \mathbf{u}_k, \mathbf{u}_l \rangle = \delta[k - l] \quad (2.199)$$

The last property indicates that the column (row) vectors $\{\mathbf{u}_k\}$ form an orthogonal basis that spans \mathbb{C}^N . Any vector $\mathbf{x} = [x[1], \dots, x[N]]^T \in \mathbb{C}^N$ represented by

the standard basis $\mathbf{I} = [\mathbf{e}_1, \dots, \mathbf{e}_N]$ as:

$$\mathbf{x} = \begin{bmatrix} x[1] \\ \vdots \\ x[N] \end{bmatrix} = \sum_{n=1}^N x[n] \mathbf{e}_n = [\mathbf{e}_1, \dots, \mathbf{e}_N] \begin{bmatrix} x[1] \\ \vdots \\ x[N] \end{bmatrix} = \mathbf{Ix} \quad (2.200)$$

can also be represented by the basis $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_N]$ as:

$$\mathbf{x} = \mathbf{Ix} = \mathbf{U}\mathbf{U}^*\mathbf{x} = \mathbf{U}\mathbf{c} = [\mathbf{u}_1, \dots, \mathbf{u}_N] \begin{bmatrix} c[1] \\ \vdots \\ c[N] \end{bmatrix} = \sum_{k=1}^N c[k] \mathbf{u}_k \quad (2.201)$$

where we have defined

$$\mathbf{c} = \begin{bmatrix} c[1] \\ \vdots \\ c[N] \end{bmatrix} = \mathbf{U}^*\mathbf{x} = \begin{bmatrix} \mathbf{u}_1^* \\ \vdots \\ \mathbf{u}_N^* \end{bmatrix} \mathbf{x}, \quad \text{i.e.} \quad c[k] = \mathbf{u}_k^* \mathbf{x} = \langle \mathbf{x}, \mathbf{u}_k \rangle \quad (2.202)$$

Combining the two equations we get

$$\begin{cases} \mathbf{c} = \mathbf{U}^*\mathbf{x} \\ \mathbf{x} = \mathbf{U}\mathbf{c} \end{cases} \quad (2.203)$$

This is the generalized Fourier transform in Eqs.2.88 and 2.89, by which a vector \mathbf{x} is rotated to become another vector \mathbf{c} .

This result can be extended to the continuous transformation first given in Eqs.2.91 and 2.93 for signal vectors in the form of continuous functions. In general, corresponding to any given unitary transformation U , a signal vector $\mathbf{x} \in H$ can be alternatively represented by a coefficient vector $\mathbf{c} = U^*\mathbf{x}$ (where \mathbf{c} can be either a set of discrete coefficients $c[k]$ or a continuous function $c(f)$). The original signal vector \mathbf{x} can always be reconstructed from \mathbf{c} by applying U on both sides of $\mathbf{c} = U^*\mathbf{x}$ to get $U\mathbf{c} = UU^*\mathbf{x} = I\mathbf{x} = \mathbf{x}$, i.e., we get a unitary transform pair in the most general form:

$$\begin{cases} \mathbf{c} = U^*\mathbf{x} \\ \mathbf{x} = U\mathbf{c} \end{cases} \quad (2.204)$$

The first equation is the forward transform that maps the signal vector \mathbf{x} to a coefficient vector \mathbf{c} , while the second equation is the inverse transform by which the signal is reconstructed. In particular, when $U = I$ is an identity operator, both equations in Eq.2.204 become an identity $\mathbf{x} = I\mathbf{x} = \mathbf{x}$, i.e., no transformation is carried out.

Previously we considered the rotation of a given vector \mathbf{x} . We next consider the rotation of the basis that spans the space. Specifically let $\{\mathbf{a}_k\}$ be an arbitrary basis of \mathbb{C}^N (not necessarily orthogonal), then any vector \mathbf{x} can be represented in terms of a set of coefficients $c[k]$:

$$\mathbf{x} = \sum_{k=1}^N c[k] \mathbf{a}_k \quad (2.205)$$

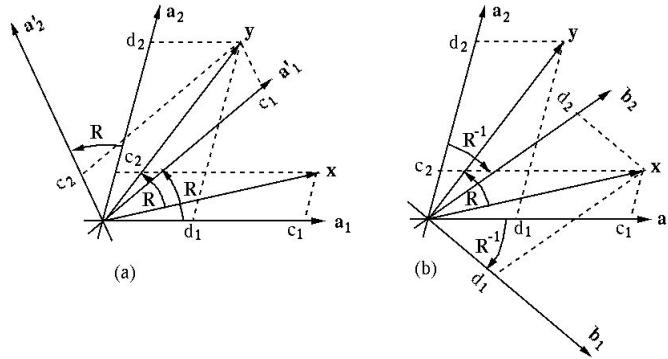


Figure 2.7 Rotation of vectors and bases

Rotating this vector by a unitary matrix \mathbf{U} , we get a new vector:

$$\mathbf{U}\mathbf{x} = \mathbf{U} \left[\sum_{k=1}^N c[k] \mathbf{a}_k \right] = \sum_{k=1}^N c[k] \mathbf{U} \mathbf{a}_k = \sum_{k=1}^N c[k] \mathbf{a}'_k = \mathbf{y} \quad (2.206)$$

This equation indicates that vector \mathbf{y} after the rotation can still be represented by the same set of coefficients $c[k]$, if the basis $\{\mathbf{a}_k\}$ is also rotated the same way to become $\mathbf{a}'_k = \mathbf{U} \mathbf{a}_k$, as illustrated in Fig.2.7(a) for the 2-D case.

Under the original basis $\{\mathbf{a}_k\}$, the rotated vector \mathbf{y} can be represented in terms of a set of new coefficients $\{\dots, d[k], \dots\}$:

$$\mathbf{y} = \sum_{k=1}^N d[k] \mathbf{a}_k = [\mathbf{a}_1, \dots, \mathbf{a}_N] \begin{bmatrix} d[1] \\ \vdots \\ d[N] \end{bmatrix} \quad (2.207)$$

The N new coefficients $d[n]$ can be obtained by solving this linear equation system with N equations (with $O(N^3)$ complexity).

On the other hand, if we rotate \mathbf{y} in the opposite direction by the inverse matrix $\mathbf{U}^{-1} = \mathbf{U}^*$, of course we get \mathbf{x} back:

$$\mathbf{U}^{-1} \mathbf{y} = \mathbf{U}^{-1} \left[\sum_{k=1}^N d[k] \mathbf{a}_k \right] = \sum_{k=1}^N d[k] \mathbf{U}^{-1} \mathbf{a}_k = \sum_{k=1}^N d[k] \mathbf{b}_k \quad (2.208)$$

where $\mathbf{b}_k = \mathbf{U}^{-1} \mathbf{a}_k = \mathbf{U}^* \mathbf{a}_k$ is the k th vector of a new basis obtained by rotating \mathbf{a}_k of the old basis in the opposite direction. In fact, as

$$P_{\mathbf{a}_k}(\mathbf{y}) = \frac{\langle \mathbf{y}, \mathbf{a}_k \rangle}{\|\mathbf{a}_k\|} = \frac{\langle \mathbf{U}\mathbf{x}, \mathbf{U}\mathbf{b}_k \rangle}{\|\mathbf{U}\mathbf{a}_k\|} = \frac{\langle \mathbf{x}, \mathbf{b}_k \rangle}{\|\mathbf{b}_k\|} = P_{\mathbf{b}_k}(\mathbf{x}) \quad (2.209)$$

we see that the scalar projection of the new vector $\mathbf{y} = \mathbf{U}\mathbf{x}$ onto the old basis \mathbf{a}_k is the same as that of the old vector \mathbf{x} onto the new basis $\mathbf{b}_k = \mathbf{U}^{-1} \mathbf{a}_k$. In other words, a rotation of the vector is equivalent to a rotation in the opposite direction of the basis, as one would intuitively expect. This is illustrated in Fig.2.7(b). A rotation in a 3-D space is illustrated in Fig.2.8.

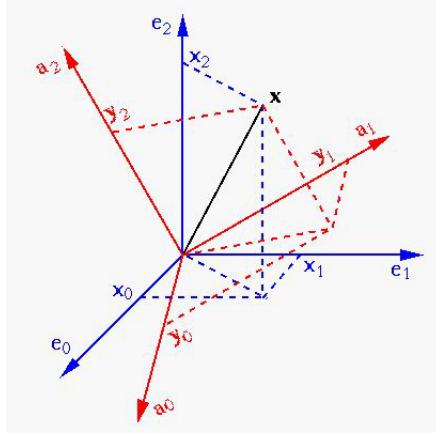


Figure 2.8 Rotation of coordinate system

In summary, multiplication of a vector $\mathbf{x} \in \mathbb{C}^N$ by a unitary matrix corresponds to a rotation of the vector. The transformation pair in Eq.2.203 can therefore be interpreted as a rotation of \mathbf{x} to get the coefficients $\mathbf{U}^*\mathbf{x} = \mathbf{c}$, and a rotation of \mathbf{c} in the opposite direction $\mathbf{x} = \mathbf{U}\mathbf{c}$ gets the original vector \mathbf{x} back. Moreover, a different rotation $\mathbf{d} = \mathbf{V}^*\mathbf{x}$ by another unitary matrix \mathbf{V} will result in a different set of coefficients \mathbf{d} , and these two sets of coefficients \mathbf{c} and \mathbf{d} are also related by a rotation corresponding to a unitary matrix $\mathbf{W} = \mathbf{V}^*\mathbf{U}$:

$$\mathbf{d} = \mathbf{V}^*\mathbf{x} = \mathbf{V}^*\mathbf{U}\mathbf{c} = \mathbf{W}\mathbf{c} \quad (2.210)$$

Example 2.5: In Example 2.2, a vector $\mathbf{x} = [1, 2]^T = 1\mathbf{e}_1 + 2\mathbf{e}_2$ is represented under a basis composed of $\mathbf{a}_1 = [1, 0]^T$ and $\mathbf{a}_2 = [-1, 2]^T$:

$$\mathbf{x} = 1\mathbf{a}_1 + 2\mathbf{a}_2 = 2 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + 1 \begin{bmatrix} -1 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad (2.211)$$

This basis $\{\mathbf{a}_1, \mathbf{a}_2\}$ can be rotated by $\theta = 45^\circ$ by a orthogonal matrix:

$$\mathbf{R} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} = 0.707 \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \quad (2.212)$$

to become a new basis $\{\mathbf{b}_1, \mathbf{b}_2\}$:

$$\mathbf{b}_1 = \mathbf{R}\mathbf{a}_1 = \mathbf{R} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = 0.707 \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad \mathbf{b}_2 = \mathbf{R}\mathbf{a}_2 = \mathbf{R} \begin{bmatrix} -1 \\ 2 \end{bmatrix} = 0.707 \begin{bmatrix} -3 \\ 1 \end{bmatrix} \quad (2.213)$$

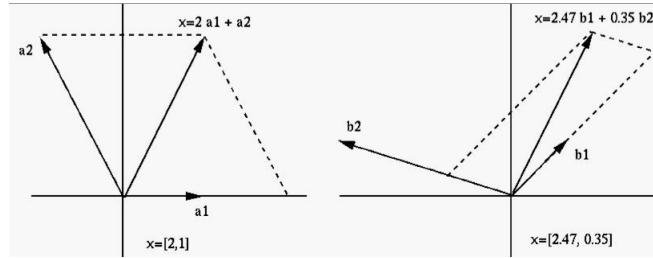


Figure 2.9 Rotation of a basis

Under this new basis, \mathbf{x} is represented as:

$$\begin{aligned}\mathbf{x} &= c'[1]\mathbf{b}_1 + c'[2]\mathbf{b}_2 = c'[1] 0.707 \begin{bmatrix} 1 \\ 1 \end{bmatrix} + c'[2] 0.707 \begin{bmatrix} -3 \\ 1 \end{bmatrix} \\ &= 0.707 \begin{bmatrix} 1 & -3 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} c'[1] \\ c'[2] \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}\end{aligned}\quad (2.214)$$

Solving this we get $c'[1] = 2.47$ and $c'[2] = 0.35$, i.e., $\mathbf{x} = 2.47\mathbf{b}_1 + 0.35\mathbf{b}_2$, as shown in Fig.2.9. In this case, the coefficients $c'[1]$ and $c'[2]$ cannot be found as the projections of \mathbf{x} onto basis vectors \mathbf{b}_1 and \mathbf{b}_2 as they are not orthogonal.

We see that the same vector \mathbf{x} can be equivalently represented by different bases:

$$\mathbf{x} = 1\mathbf{e}_1 + 2\mathbf{e}_2 = 2\mathbf{a}_1 + 1\mathbf{a}_2 = 2.47\mathbf{b}_1 + 0.35\mathbf{b}_2 \quad (2.215)$$

2.3 Projection Theorem and Signal Approximation

2.3.1 Projection Theorem and Pseudo-Inverse

A signal in a high dimensional space (possibly infinite dimensional) may need to be approximated in a lower dimensional subspace, for various reasons such as computational complexity reduction and data compression. Although a complete basis is necessary to represent any given vector in a vector space, it is still possible to approximate the vector in a subspace if certain error is allowed. Also, a continuous function may not be accurately representable in a finite dimensional space, but it may still be needed to approximate the function in such a space for certain signal processing desired. The issue in such approximation is how to minimize the error.

Let H be a Hilbert space (finite or infinite dimensional), and $U \subset H$ be an M-D subspace spanned by a set of M basis vectors $\{\mathbf{a}_1, \dots, \mathbf{a}_M\}$ (not necessarily orthogonal), and assume a given vector $\mathbf{x} \in H$ is approximated by a vector $\hat{\mathbf{x}} \in$

U :

$$\mathbf{x} \approx \hat{\mathbf{x}} = \sum_{k=1}^M c[k] \mathbf{a}_k \quad (2.216)$$

An error vector is defined as

$$\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}} = \mathbf{x} - \sum_{k=1}^M c[k] \mathbf{a}_k \quad (2.217)$$

The least squares error of the approximation is defined as:

$$\varepsilon = \|\tilde{\mathbf{x}}\|^2 = \langle \tilde{\mathbf{x}}, \tilde{\mathbf{x}} \rangle \quad (2.218)$$

The goal is to find a set of coefficients $c[1], \dots, c[M]$ so that the error ε is minimized.

Theorem 2.7. (*The projection theorem*) *The least squares error $\varepsilon = \|\tilde{\mathbf{x}}\|^2$ of the approximation in equation 2.216 is minimized if and only if the error vector $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$ is orthogonal to the subspace U :*

$$\tilde{\mathbf{x}} \perp U, \quad \text{i.e.,} \quad \tilde{\mathbf{x}} \perp \mathbf{a}_k, \quad (k = 1, \dots, M) \quad (2.219)$$

Proof: Let $\hat{\mathbf{x}}$ and $\hat{\mathbf{x}}'$ be two vectors both in the subspace U , where $\hat{\mathbf{x}}'$ is arbitrary but $\hat{\mathbf{x}}$ is the projection of \mathbf{x} onto U , i.e., $(\mathbf{x} - \hat{\mathbf{x}}) \perp U$. As $\hat{\mathbf{x}} - \hat{\mathbf{x}}'$ is also a vector in U , we have $(\mathbf{x} - \hat{\mathbf{x}}) \perp (\hat{\mathbf{x}} - \hat{\mathbf{x}}')$, i.e., $\langle \mathbf{x} - \hat{\mathbf{x}}, \hat{\mathbf{x}} - \hat{\mathbf{x}}' \rangle = 0$. Now consider the error associated with $\hat{\mathbf{x}}'$:

$$\begin{aligned} \|\mathbf{x} - \hat{\mathbf{x}}'\|^2 &= \|\mathbf{x} - \hat{\mathbf{x}} + \hat{\mathbf{x}} - \hat{\mathbf{x}}'\|^2 \\ &= \|\mathbf{x} - \hat{\mathbf{x}}\|^2 + \langle \mathbf{x} - \hat{\mathbf{x}}, \hat{\mathbf{x}} - \hat{\mathbf{x}}' \rangle + \langle \hat{\mathbf{x}} - \hat{\mathbf{x}}', \mathbf{x} - \hat{\mathbf{x}} \rangle + \|\hat{\mathbf{x}} - \hat{\mathbf{x}}'\|^2 \\ &= \|\mathbf{x} - \hat{\mathbf{x}}\|^2 + \|\hat{\mathbf{x}} - \hat{\mathbf{x}}'\|^2 \geq \|\mathbf{x} - \hat{\mathbf{x}}\|^2 \end{aligned} \quad (2.220)$$

We see that the error $\|\mathbf{x} - \hat{\mathbf{x}}'\|^2$ associated with $\hat{\mathbf{x}}'$ is always greater than the error $\|\mathbf{x} - \hat{\mathbf{x}}\|^2$ associated with $\hat{\mathbf{x}}$, unless $\hat{\mathbf{x}}' = \hat{\mathbf{x}}$, i.e., the error is minimized if and only if the approximation is $\hat{\mathbf{x}}$, the projection of \mathbf{x} onto the subspace U . Q.E.D.

This theorem can be understood intuitively as shown in Fig.2.10, where a vector \mathbf{x} in a 3-D space is approximated by a vector $\hat{\mathbf{x}}$ in a 2-D subspace $\hat{\mathbf{x}} = c[1]\mathbf{a}_1 + c[2]\mathbf{a}_2$. The error $\varepsilon = \|\tilde{\mathbf{x}}\|^2 = \|\mathbf{x} - \hat{\mathbf{x}}\|^2$ is indeed minimum if $\mathbf{x} - \hat{\mathbf{x}}$ is orthogonal to the 2-D plane spanned by the basis vectors \mathbf{a}_1 and \mathbf{a}_2 , as any other vector $\hat{\mathbf{x}}'$ in this plane would be associated with a larger error, i.e., the approximation $\hat{\mathbf{x}}$ is the *projection* of \mathbf{x} onto the subspace U .

The coefficients corresponding to the optimal approximation can be found based on the projection theorem. As the minimum error vector $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}}$ has

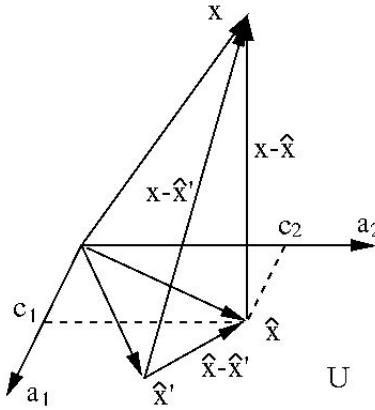


Figure 2.10 Projection theorem

to be orthogonal to each of the basis vectors that span the subspace U , we have:

$$\begin{aligned} \langle \tilde{x}, \mathbf{a}_l \rangle &= \langle \mathbf{x} - \sum_{k=1}^M c[k] \mathbf{a}_k, \mathbf{a}_l \rangle = \langle \mathbf{x}, \mathbf{a}_l \rangle - \sum_{k=1}^M c[k] \langle \mathbf{a}_k, \mathbf{a}_l \rangle = 0, \\ (l &= 1, \dots, M) \end{aligned} \quad (2.221)$$

i.e.

$$\langle \mathbf{x}, \mathbf{a}_l \rangle = \sum_{k=1}^M c[k] \langle \mathbf{a}_k, \mathbf{a}_l \rangle, \quad (m = 1, \dots, M) \quad (2.222)$$

These M equations can be written in matrix form:

$$\begin{bmatrix} \langle \mathbf{x}, \mathbf{a}_1 \rangle \\ \vdots \\ \langle \mathbf{x}, \mathbf{a}_M \rangle \end{bmatrix}_{M \times 1} = \begin{bmatrix} \langle \mathbf{a}_1, \mathbf{a}_1 \rangle & \cdots & \langle \mathbf{a}_M, \mathbf{a}_1 \rangle \\ \vdots & \ddots & \vdots \\ \langle \mathbf{a}_1, \mathbf{a}_M \rangle & \cdots & \langle \mathbf{a}_M, \mathbf{a}_M \rangle \end{bmatrix}_{M \times M} \begin{bmatrix} c[1] \\ \vdots \\ c[M] \end{bmatrix}_{M \times 1} \quad (2.223)$$

Solving this system of M equations and M unknowns, we get the optimal coefficients $c[k]$ and the vector \mathbf{x} can be approximated in the M-D subspace as shown in Eq.2.216.

In particular, if the basis vectors of the Hilbert space are orthogonal, i.e., $\langle \mathbf{a}_k, \mathbf{a}_l \rangle = 0$ for all $k \neq l$, then all off-diagonal components of the M by M matrix in Eq.2.223 above are zero, and each of the coefficients can be obtained independently:

$$c[k] = \frac{\langle \mathbf{x}, \mathbf{a}_k \rangle}{\langle \mathbf{a}_k, \mathbf{a}_k \rangle} = \frac{\langle \mathbf{x}, \mathbf{a}_k \rangle}{\|\mathbf{a}_k\|^2}, \quad (k = 1, \dots, M) \quad (2.224)$$

Eq.2.216 now becomes:

$$\hat{x} = \sum_{k=1}^M c[k] \mathbf{a}_k = \sum_{k=1}^M \frac{\langle \mathbf{x}, \mathbf{a}_k \rangle}{\|\mathbf{a}_k\|^2} \mathbf{a}_k = \sum_{k=1}^M p_{\mathbf{a}_k}(\mathbf{x}) \quad (2.225)$$

We see that $\hat{\mathbf{x}}$ is the vector sum of the projections of \mathbf{x} onto each of the basis vectors \mathbf{a}_k ($k = 1, \dots, M$) of the subspace U . Moreover, if the basis is also normalized, i.e. $\langle \mathbf{a}_k, \mathbf{a}_l \rangle = \delta[k - l]$, then we have

$$c[k] = \langle \mathbf{x}, \mathbf{a}_k \rangle, \quad (k = 1, \dots, M) \quad (2.226)$$

and Eq.2.216 becomes:

$$\hat{\mathbf{x}} = \sum_{k=1}^M \langle \mathbf{x}, \mathbf{a}_k \rangle \mathbf{a}_k \quad (2.227)$$

Consider for example the space \mathbb{C}^N spanned by a basis $\{\mathbf{a}_1, \dots, \mathbf{a}_N\}$ (not necessarily orthogonal). We wish to express a given vector $\mathbf{x} \in \mathbb{C}^N$ in an M-D subspace spanned by M basis vectors $\{\mathbf{a}_1, \dots, \mathbf{a}_M\}$ as:

$$\mathbf{x} = \begin{bmatrix} x[1] \\ \vdots \\ x[N] \end{bmatrix}_{N \times 1} = \sum_{k=1}^M c[k] \mathbf{a}_k = [\mathbf{a}_1, \dots, \mathbf{a}_M]_{N \times M} \begin{bmatrix} c[1] \\ \vdots \\ c[M] \end{bmatrix}_{M \times 1} = \mathbf{A}\mathbf{c} \quad (2.228)$$

This equation system is over-determined with only M unknowns $c[1], \dots, c[M]$ but $N > M$ equations. As the N by M non-square matrix \mathbf{A} is not invertible, the system has no solution in general, indicating the impossibility of representing the N-D vector \mathbf{x} in an M-D subspace. However, based on the projection theorem, we can find the optimal approximation of \mathbf{x} in the M-D subspace by solving Eq.2.223. In this case the inner products in the equation become $\langle \mathbf{x}, \mathbf{a}_k \rangle = \mathbf{a}_k^* \mathbf{x}$ and $\langle \mathbf{a}_k, \mathbf{a}_l \rangle = \mathbf{a}_l^* \mathbf{a}_k$, Eq. 2.223 can be written as:

$$\mathbf{A}^* \mathbf{x} = \mathbf{A}^* \mathbf{A} \mathbf{c} \quad (2.229)$$

where $\mathbf{A}^* \mathbf{A}$ is an M by M square matrix and therefore invertible. Pre-multiplying its inverse $(\mathbf{A}^T \mathbf{A})^{-1}$ on both sides, we can find the optimal solution for \mathbf{c} of the over-determined equation system corresponding to the minimum least square error:

$$\mathbf{c} = (\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}^* \mathbf{x} = \mathbf{A}^- \mathbf{x} \quad (2.230)$$

where

$$\mathbf{A}^- = (\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}^* \quad (2.231)$$

is an M by N matrix, known as the *generalized inverse or pseudo-inverse* of the N by M matrix \mathbf{A} (Appendix 1), and we have: $\mathbf{A}^- \mathbf{A} = \mathbf{I}$.² If all N basis vectors can be used, then \mathbf{A} becomes an N by N square matrix and the pseudo-inverse

² The pseudo-inverse in Eq.2.231 is for the case where \mathbf{A} has more columns than rows ($M < N$ in this case). If \mathbf{A} has more rows than columns ($M > N$ in this case), the pseudo-inverse becomes:

$$\mathbf{A}^- = \mathbf{A}^* (\mathbf{A} \mathbf{A}^*)^{-1} \quad (2.232)$$

becomes the regular inverse:

$$\mathbf{A}^- = \mathbf{A}^{-1}(\mathbf{A}^*)^{-1}\mathbf{A}^* = \mathbf{A}^{-1} \quad (2.233)$$

and the coefficients can be found simply by:

$$\mathbf{c} = \mathbf{A}^{-1}\mathbf{x} \quad (2.234)$$

If the basis are orthogonal, i.e., $\langle \mathbf{a}_k, \mathbf{a}_l \rangle = 0$ for all $k \neq l$, the M coefficients can be found as

$$c[k] = \frac{\langle \mathbf{x}, \mathbf{a}_k \rangle}{\langle \mathbf{a}_k, \mathbf{a}_k \rangle} = \frac{\langle \mathbf{x}, \mathbf{a}_k \rangle}{\|\mathbf{a}_k\|^2}, \quad (k = 1, \dots, M) \quad (2.235)$$

with complexity $O(M^2)$. Moreover, if the basis is orthonormal with $\|\mathbf{a}_k\|^2 = 1$, the coefficients become:

$$c[k] = \langle \mathbf{x}, \mathbf{a}_k \rangle = \mathbf{a}_k^* \mathbf{x}, \quad (k = 1, \dots, M) \quad (2.236)$$

and the approximation becomes:

$$\hat{\mathbf{x}}_M = \sum_{k=1}^M c[k] \mathbf{a}_k = \sum_{k=1}^M \langle \mathbf{x}, \mathbf{a}_k \rangle \mathbf{a}_k \quad (2.237)$$

This is actually the unitary transformation in Eq.2.203. We see that under any orthonormal basis $\{\mathbf{a}_k\}$ of \mathbb{C}^N , a given vector \mathbf{x} can always be optimally approximated in the M-D subspace ($M < N$) with least square error:

$$\begin{aligned} \varepsilon &= \|\tilde{\mathbf{x}}\|^2 = \langle \mathbf{x} - \hat{\mathbf{x}}_M, \mathbf{x} - \hat{\mathbf{x}}_M \rangle \\ &= \langle \mathbf{x}, \mathbf{x} \rangle - \langle \mathbf{x}, \hat{\mathbf{x}}_M \rangle - \langle \hat{\mathbf{x}}_M, \mathbf{x} \rangle + \langle \hat{\mathbf{x}}_M, \hat{\mathbf{x}}_M \rangle \\ &= \|\mathbf{x}\|^2 - \sum_{k=1}^M \langle \mathbf{x}, \mathbf{a}_k \rangle \bar{c}[k] - \sum_{k=1}^M c[k] \langle \mathbf{a}_k, \mathbf{x} \rangle + \sum_{k=1}^M |c[k]|^2 \\ &= \|\mathbf{x}\|^2 - \sum_{k=1}^M |c[k]|^2 = \sum_{k=M+1}^N |c[k]|^2 \geq 0 \end{aligned} \quad (2.238)$$

The last equation is due to Parseval's identity $\|\mathbf{x}\|^2 = \|\mathbf{c}\|^2 = \sum_{k=1}^N |c[k]|^2$. When $M \rightarrow N$, the sequence $\hat{\mathbf{x}}_M$ converges to \mathbf{x} :

$$\lim_{M \rightarrow N} \hat{\mathbf{x}}_M = \lim_{M \rightarrow N} \sum_{k=1}^M c[k] \mathbf{a}_k = \sum_{k=1}^N c[k] \mathbf{a}_k = \mathbf{x} \quad (2.239)$$

and Eq.2.238 becomes:

$$\lim_{M \rightarrow N} \varepsilon = \|\mathbf{x}\|^2 - \sum_{k=1}^N |c[k]|^2 = 0 \quad (2.240)$$

This is of course Parseval's identity $\|\mathbf{x}\|^2 = \|\mathbf{c}\|^2$

Example 2.6: Consider a 3-D Euclidean space \mathbb{R}^3 spanned by a set of three linearly independent vectors:

$$\mathbf{a}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{a}_2 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{a}_3 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad (2.241)$$

We want to find two coefficients $c[1]$ and $c[2]$ so that a given vector $\mathbf{x} = [1, 2, 3]^T$ can be optimally approximated as $\hat{\mathbf{x}} = c[1]\mathbf{a}_1 + c[2]\mathbf{a}_2$ in the 2-D subspace spanned by \mathbf{a}_1 and \mathbf{a}_2 . First we construct a matrix composed of \mathbf{a}_1 and \mathbf{a}_2 :

$$\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2] = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \quad (2.242)$$

Next we find the pseudo inverse of \mathbf{A} :

$$\mathbf{A}^- = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (2.243)$$

The two coefficients can then be obtained as:

$$\mathbf{c} = \begin{bmatrix} c[1] \\ c[2] \end{bmatrix} = \mathbf{A}^- \mathbf{x} = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} -1 \\ 2 \\ 3 \end{bmatrix} \quad (2.244)$$

The optimal approximation is therefore

$$\hat{\mathbf{x}} = c[1]\mathbf{a}_1 + c[2]\mathbf{a}_2 = -1 \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + 2 \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix} \quad (2.245)$$

which is indeed the projection of $\mathbf{x} = [1, 2, 3]^T$ onto the 2-D subspace spanned by \mathbf{a}_1 and \mathbf{a}_2 .

Alternatively if we want to approximate \mathbf{x} by \mathbf{a}_2 and \mathbf{a}_3 as $\hat{\mathbf{x}} = c[2]\mathbf{a}_2 + c[3]\mathbf{a}_3$, we have:

$$\mathbf{A} = [\mathbf{a}_2, \mathbf{a}_3] = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{A}^- = \frac{1}{2} \begin{bmatrix} 1 & 1 & -2 \\ 0 & 0 & 2 \end{bmatrix} \quad (2.246)$$

and

$$\mathbf{c} = \mathbf{A}^- \mathbf{x} = \begin{bmatrix} -1.5 \\ 3 \end{bmatrix}, \quad \hat{\mathbf{x}} = c[2]\mathbf{a}_2 + c[3]\mathbf{a}_3 = -1.5 \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} + 3 \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1.5 \\ 1.5 \\ 3 \end{bmatrix} \quad (2.247)$$

If all three basis vectors can be used, then the coefficients can be found as:

$$\mathbf{c} = \mathbf{A}^{-1} \mathbf{x} = [\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3]^{-1} \mathbf{x} = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} -1 \\ -1 \\ 3 \end{bmatrix} \quad (2.248)$$

and \mathbf{x} can be precisely represented as:

$$\mathbf{x} = c[1]\mathbf{a}_1 + c[2]\mathbf{a}_2 + c[3]\mathbf{a}_3 = \mathbf{A}\mathbf{c} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} \quad (2.249)$$

2.3.2 Signal Approximation

As discussed above, a signal vector can be represented equivalently under different bases that span the space, in terms of the total energy (Parseval's equality). However, these representations may differ drastically in term of how different types of information contained in the signal is concentrated in different signal components and represented by the coefficients. Sometimes certain advantages can be gained from one particular basis compared to another, depending on the specific application. In the following we consider two simple examples to illustrate such issues.

Example 2.7: Given a signal $x(t) = t$ defined over $0 \leq t < 2$ (undefined outside the range), we want to optimally approximate it in a subspace spanned by the following two bases.

- First we use the standard functions $e_1(t)$ and $e_2(t)$:

$$\hat{x}(t) = c[1]e_1(t) + c[2]e_2(t) \quad (2.250)$$

where $e_1(t)$ and $e_2(t)$ are defined as:

$$e_1(t) = \begin{cases} 1, & 0 \leq t < 1 \\ 0, & 1 \leq t < 2 \end{cases}, \quad e_2(t) = \begin{cases} 0, & 0 \leq t < 1 \\ 1, & 1 \leq t < 2 \end{cases} \quad (2.251)$$

These two basis functions are obviously orthonormal $\langle e_i(t), e_j(t) \rangle = \delta[i - j]$. Following the projection theorem, the coefficients $c[1]$ and $c[2]$ can be found by solving these two simultaneous equations (Eq.2.222):

$$\begin{aligned} c[1] \int_0^2 e_1(t)e_1(t)dt + c[2] \int_0^2 e_2(t)e_1(t)dt &= \int_0^2 x(t)e_1(t)dt \\ c[1] \int_0^2 e_1(t)e_2(t)dt + c[2] \int_0^2 e_2(t)e_2(t)dt &= \int_0^2 x(t)e_2(t)dt \end{aligned}$$

As $e_1(t)$ and $e_2(t)$ are orthonormal, the equation system becomes decoupled and the two coefficients $c[1]$ and $c[2]$ can be obtained independently as the projections of $x(t)$ onto each of the basis functions.

$$c[1] = \int_0^2 x(t)e_1(t)dt = \int_0^1 t dt = 0.5, \quad c[2] = \int_0^2 x(t)e_2(t)dt = \int_1^2 t dt = 1.5 \quad (2.252)$$

Now the signal $x(t)$ can be approximated as:

$$\hat{x}(t) = 0.5e_1(t) + 1.5e_2(t) = \begin{cases} 0.5, & 0 \leq t < 1 \\ 1.5, & 1 \leq t < 2 \end{cases} \quad (2.253)$$

- Next, we use two different basis functions $u_1(t)$ and $u_2(t)$:

$$\hat{x}(t) = d[1]u_1(t) + d[2]u_2(t) \quad (2.254)$$

where

$$u_1(t) = \frac{1}{\sqrt{2}}[e_1(t) + e_2(t)] = \frac{1}{\sqrt{2}}$$

$$u_2(t) = \frac{1}{\sqrt{2}}[e_1(t) - e_2(t)] = \begin{cases} 1/\sqrt{2}, & 0 \leq t < 1 \\ -1/\sqrt{2}, & 1 \leq t < 2 \end{cases}$$

Again these two basis functions are orthonormal $\langle u_i(t), u_j(t) \rangle = \delta[i - j]$, and the two coefficients $d[1]$ and $d[2]$ can be obtained independently as:

$$d[1] = \int_0^2 x(t)u_1(t)dt = \sqrt{2}, \quad d[2] = \int_0^2 x(t)u_2(t)dt = -\frac{1}{\sqrt{2}} \quad (2.255)$$

The approximation is:

$$\hat{x}(t) = \sqrt{2}u_1(t) - \frac{1}{\sqrt{2}}u_2(t) = \begin{cases} 0.5, & 0 \leq t < 1 \\ 1.5, & 1 \leq t < 2 \end{cases} \quad (2.256)$$

We see that the approximations based on these two different bases happen to be identical as illustrated in Fig.2.11. We can make the following observations:

- The first basis $\{e_1(t), e_2(t)\}$ is the standard basis, the two coefficients $c[1]$ and $c[2]$ represent the average values of the signal during two consecutive time segments.
- The second basis $\{u_1(t), u_2(t)\}$ represents the signal $x(t)$ in a totally different way. The first coefficient $d[1]$ represents the average of the signal (0 frequency), while the second coefficient $d[2]$ represents the variation of the signal in terms of the difference between the first half and the second. (In fact they correspond to the first two frequency components in several orthogonal transforms, including the discrete Fourier transform, discrete cosine transform, Walsh-Hadamard transform, etc.)
- The second basis $\{u_1(t), u_2(t)\}$ is a rotated version of the first basis $\{e_1(t), e_2(t)\}$, as shown in Fig.2.12, and naturally they produce the same approximation $\hat{x}(t)$. Consequently, the two sets of coefficients $\{c[1], c[2]\}$ and $\{d[1], d[2]\}$ are related by an orthogonal matrix representing the rotation by an angle $\theta = -45^\circ$:

$$\begin{bmatrix} d[2] \\ d[1] \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} c[2] \\ c[1] \end{bmatrix} = \begin{bmatrix} \sqrt{2}/2 & -\sqrt{2}/2 \\ \sqrt{2}/2 & \sqrt{2}/2 \end{bmatrix} \begin{bmatrix} 1/2 \\ 3/2 \end{bmatrix} = \begin{bmatrix} -1/\sqrt{2} \\ \sqrt{2} \end{bmatrix} \quad (2.257)$$

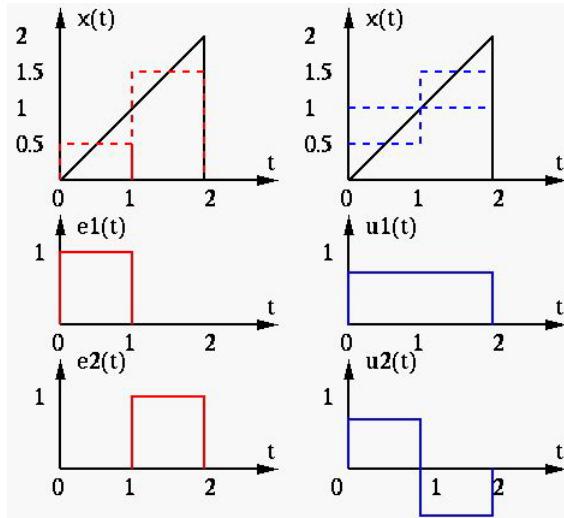


Figure 2.11 Approximation of a signal by two different basis functions

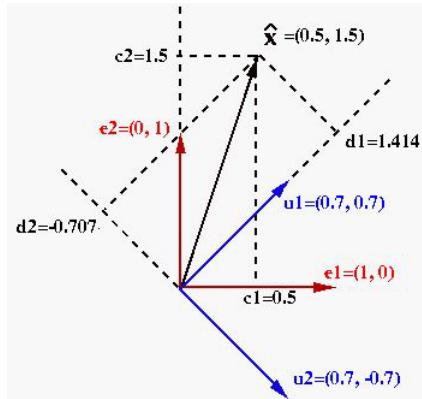


Figure 2.12 Representation of a signal vector under two different bases

Example 2.8: The temperature is measured every 3 hours in a day to obtain 8 samples as shown below:

Time (hours)	0	3	6	9	12	15	18	21
Temperature (F)	65	60	65	70	75	80	75	70

These time samples can be considered as a vector $\mathbf{x} = [x[1], \dots, x[8]]^T = [65, 60, 65, 70, 75, 80, 75, 70]^T$ in \mathbb{R}^8 space under the standard basis implicitly used, i.e., the nth element $x[n]$ is the coefficient for the nth standard basis vector

$\mathbf{e}_k = [0, \dots, 0, 1, 0, \dots, 0]^T$ (all elements are zero except the nth one), i.e.,

$$\mathbf{x} = \sum_{k=1}^8 x[k] \mathbf{e}_k \quad (2.258)$$

This 8-D signal vector \mathbf{x} is approximated in an M-D subspace ($M < 8$) as shown below for different M values:

- $M = 1$: \mathbf{x} is approximated as $\hat{\mathbf{x}} = c[1]\mathbf{b}_1$ in a 1-D subspace spanned by $\mathbf{b}_1 = [1, 1, 1, 1, 1, 1, 1, 1]^T$. Here the coefficient can be obtained as:

$$c[1] = \frac{\langle \mathbf{x}, \mathbf{b}_1 \rangle}{\langle \mathbf{b}_1, \mathbf{b}_1 \rangle} = \frac{560}{8} = 70 \quad (2.259)$$

which represents the average or DC component of the daily temperature. The approximation is:

$$\hat{\mathbf{x}} = c[1]\mathbf{b}_1 = [70, 70, 70, 70, 70, 70, 70, 70]^T \quad (2.260)$$

The error vector is $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}} = [-5, -10, -5, 0, 5, 10, 5, 0]^T$ and the error is $\|\tilde{\mathbf{x}}\|^2 = 300$.

- $M = 2$: \mathbf{x} can be better approximated in a 2-D subspace spanned by the same \mathbf{b}_1 and a second basis vector $\mathbf{b}_2 = [1, 1, 1, 1, -1, -1, -1, -1]^T$. As \mathbf{b}_2 is orthogonal to \mathbf{b}_1 , its coefficient $c[2]$ can be found independently:

$$c[2] = \frac{\langle \mathbf{x}, \mathbf{b}_2 \rangle}{\langle \mathbf{b}_2, \mathbf{b}_2 \rangle} = \frac{-40}{8} = -5 \quad (2.261)$$

which represents the temperature difference between morning and afternoon. The approximation is:

$$\hat{\mathbf{x}} = c[1]\mathbf{b}_1 + c[2]\mathbf{b}_2 = [65.65, 65, 65, 75, 75, 75, 75, 75]^T \quad (2.262)$$

The error vector is $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}} = [0, -5, 0, 5, 0, 5, 0, -5]^T$ and the error is $\|\tilde{\mathbf{x}}\|^2 = 100$.

- $M = 3$: The approximation can be further improved if a third basis vector $\mathbf{b}_3 = [1, 1, -1, -1, -1, -1, 1, 1]^T$ is added. As all three basis vectors are orthogonal to each other, the coefficient $c[3]$ can also be independently obtained:

$$c[3] = \frac{\langle \mathbf{x}, \mathbf{b}_3 \rangle}{\langle \mathbf{b}_3, \mathbf{b}_3 \rangle} = \frac{-20}{8} = -2.5 \quad (2.263)$$

which represents the temperature difference between day-time and night-time. The approximation can be expressed as:

$$\hat{\mathbf{x}} = c[1]\mathbf{b}_1 + c[2]\mathbf{b}_2 + c[3]\mathbf{b}_3 = [62.5, 62, 5, 67.5, 67.5, 77.5, 77.5, 72.5, 72.5]^T \quad (2.264)$$

The error vector is $\tilde{\mathbf{x}} = \mathbf{x} - \hat{\mathbf{x}} = [2.5, -2.5, -2.5, 2.5, -2.5, 2.5, 2.5, -2.5]^T$ and the error is $\|\tilde{\mathbf{x}}\|^2 = 50$.

We can now make the following observations:

- The original 8-D signal vector \mathbf{x} can be approximated by $M < 8$ basis vectors spanning a M-D subspace. As more basis vectors are included in the approximation, the error becomes progressively smaller.
 - A typical signal contains both slow-varying or low-frequency components and fast-varying or high-frequency components, and the former is likely to contain more energy compared to the latter. In order to reduce error when approximating the signal, basis functions representing lower frequencies should be used first.
 - When progressively more basis functions representing more details or subtle variations in the signal are added in the signal approximation, their coefficients are likely to have lower values compared to those for the slow-varying basis functions, and they are more likely to be affected by noise such as some random fluctuation, therefore they are less significant and could be neglected without losing much essential information.
 - The three basis vectors \mathbf{b}_1 , \mathbf{b}_2 and \mathbf{b}_3 used above are actually the first three basis vectors of the sequency-ordered Hadamard transform to be discussed in Chapter 8.
-

2.4 Frames and Biorthogonal Bases

2.4.1 Frames

Previously we considered the representation of a signal vector $\mathbf{x} \in H$ as some linear combination of an orthogonal basis $\{\mathbf{u}_k\}$ that spans the space:

$$\mathbf{x} = \sum_k c[k] \mathbf{u}_k = \sum_k \langle \mathbf{x}, \mathbf{u}_k \rangle \mathbf{u}_k \quad (2.265)$$

and Parseval's identity $\|\mathbf{x}\|^2 = \|\mathbf{c}\|^2$ indicates that \mathbf{x} is equivalently represented by the coefficients \mathbf{c} without any redundancy. However, sometimes it may not be easy or even possible to identify a set of linearly independent and orthogonal basis vectors in the space. In such cases we could still consider representing a signal vector \mathbf{x} by a set of vectors $\{\mathbf{f}_k\}$ which may not be linearly independent and therefore do not form a basis of the space. A main issue is the redundancy that exists among such a set of non-independent vectors. As it is now possible to find a set of coefficients $d[k]$ so that $\sum_k d[k] \mathbf{f}_k = 0$, an immediate consequence is that the representation is no longer unique:

$$\mathbf{x} = \sum_k c[k] \mathbf{f}_k = \sum_k c[k] \mathbf{f}_k + \sum_k d[k] \mathbf{f}_k = \sum_k (c[k] + d[k]) \mathbf{f}_k \quad (2.266)$$

One consequence of the redundancy is that Parseval's identify no longer holds. The energy contained in the coefficients $\|\mathbf{c}\|^2$ may be either higher or lower than

the actual energy $\|\mathbf{x}\|^2$ in the signal. We therefore need to develop some theory to address this issue when using non-independent vectors for signal representation.

First, in order for the expansion $\mathbf{x} = \sum_k c[k] \mathbf{f}_k$ to be a precise representation of the signal vector \mathbf{x} in terms of a set of coefficients $c[k] = \langle \mathbf{x}, \mathbf{f}_k \rangle$, we need to guarantee that for any vectors $\mathbf{x}, \mathbf{y} \in H$, the following always holds:

$$\langle \mathbf{x}, \mathbf{f}_k \rangle = \langle \mathbf{y}, \mathbf{f}_k \rangle \quad \text{iff} \quad \mathbf{x} = \mathbf{y} \quad (2.267)$$

Moreover, these representations also need to be stable in the following two aspects.

- **Stable representation:**

If the difference between two vectors is small, the difference between their corresponding coefficients should also be small:

$$\text{if } \|\mathbf{x} - \mathbf{y}\|^2 \rightarrow 0, \quad \text{then} \quad \sum_k |\langle \mathbf{x}, \mathbf{f}_k \rangle - \langle \mathbf{y}, \mathbf{f}_k \rangle|^2 \rightarrow 0 \quad (2.268)$$

i.e.,

$$\sum_k |\langle \mathbf{x}, \mathbf{f}_k \rangle - \langle \mathbf{y}, \mathbf{f}_k \rangle|^2 \leq B \|\mathbf{x} - \mathbf{y}\|^2 \quad (2.269)$$

where $0 < B < \infty$ is a positive real constant. In particular if $\mathbf{y} = \mathbf{0}$ and therefore $\langle \mathbf{y}, \mathbf{f}_k \rangle = 0$, we have:

$$\sum_k |\langle \mathbf{x}, \mathbf{f}_k \rangle|^2 \leq B \|\mathbf{x}\|^2 \quad (2.270)$$

- **Stable reconstruction:**

If the difference between two sets of coefficients is small, the difference between the reconstructed vectors should also be small:

$$\text{if } \sum_k |\langle \mathbf{x}, \mathbf{f}_k \rangle - \langle \mathbf{y}, \mathbf{f}_k \rangle|^2 \rightarrow 0, \quad \text{then} \quad \|\mathbf{x} - \mathbf{y}\|^2 \rightarrow 0 \quad (2.271)$$

i.e.,

$$A \|\mathbf{x} - \mathbf{y}\|^2 \leq \sum_k |\langle \mathbf{x}, \mathbf{f}_k \rangle - \langle \mathbf{y}, \mathbf{f}_k \rangle|^2 \quad (2.272)$$

where $0 < A < \infty$ is also a positive real constant. Again if $\mathbf{y} = \mathbf{0}$ and $\langle \mathbf{y}, \mathbf{f}_k \rangle = 0$, we have:

$$A \|\mathbf{x}\|^2 \leq \sum_k |\langle \mathbf{x}, \mathbf{f}_k \rangle|^2 \quad (2.273)$$

Combining Eqs.2.270 and 2.273, we have the following definition:

Definition 2.24. *A family of finite or infinite vectors $\{\mathbf{f}_k\}$ in Hilbert space H is a frame if there exist two real constants $0 < A \leq B < \infty$, called the lower and upper bounds of the frame, such that for any $\mathbf{x} \in H$, the following holds:*

$$A \|\mathbf{x}\|^2 \leq \sum_k |\langle \mathbf{x}, \mathbf{f}_k \rangle|^2 \leq B \|\mathbf{x}\|^2 \quad (2.274)$$

In particular, if $A = B$, i.e.,

$$A\|\mathbf{x}\|^2 = \sum_k |\langle \mathbf{x}, \mathbf{f}_k \rangle|^2 \quad (2.275)$$

then the frame is tight.

2.4.2 Signal Expansion by Frames and Riesz Bases

Our purpose here is to represent a given signal vector $\mathbf{x} \in H$ as a linear combination $\mathbf{x} = \sum_k c[k] \mathbf{f}_k$ of a set of frame vectors $\{\mathbf{f}_k\}$. The process of finding the coefficients $c[k]$ needed in the combination can be considered as a *frame transformation*, denoted by F^* , that maps the given \mathbf{x} to a coefficient vector \mathbf{c} :

$$\mathbf{c} = F^* \mathbf{x} = [\dots, c[k], \dots]^T = [\dots, \langle \mathbf{x}, \mathbf{f}_k \rangle, \dots]^T \quad (2.276)$$

where we have defined $c[k] = \langle \mathbf{x}, \mathbf{f}_k \rangle$, following the unitary transformation in Eq.2.204. Here F^* is the adjoint of another transformation F , which can be found from the following inner product in the definition of a unitary transformation (Eq.2.142):

$$\begin{aligned} \langle \mathbf{c}', F^* \mathbf{x} \rangle &= \sum_k c'[k] \overline{\langle \mathbf{x}, \mathbf{f}_k \rangle} \\ &= \sum_k c'[k] \langle \mathbf{f}_k, \mathbf{x} \rangle = \langle \sum_k c'[k] \mathbf{f}_k, \mathbf{x} \rangle = \langle F \mathbf{c}', \mathbf{x} \rangle \end{aligned} \quad (2.277)$$

We see that F is a transformation that constructs a vector as a linear combination of the frame $\{\mathbf{f}_k\}$ based on a given set of coefficients \mathbf{c}' :

$$\mathbf{x}' = F \mathbf{c}' = \sum_k c'[k] \mathbf{f}_k \quad (2.278)$$

We further define an operator FF^* :

$$FF^* \mathbf{x} = F(F^* \mathbf{x}) = F \mathbf{c} = \sum_k \langle \mathbf{x}, \mathbf{f}_k \rangle \mathbf{f}_k \quad (2.279)$$

Note that different from the unitary transform satisfying $UU^* = UU^{-1} = I$, here $FF^* \neq I$ is in general not an identity operator. Applying its inverse $(FF^*)^{-1}$ to both sides of the equation above, we get:

$$\begin{aligned} \mathbf{x} &= (FF^*)^{-1} F \mathbf{c} = (FF^*)^{-1} \left[\sum_k \langle \mathbf{x}, \mathbf{f}_k \rangle \mathbf{f}_k \right] = \sum_k \langle \mathbf{x}, \mathbf{f}_k \rangle (FF^*)^{-1} \mathbf{f}_k \\ &= \sum_k \langle \mathbf{x}, \mathbf{f}_k \rangle \tilde{\mathbf{f}}_k = \sum_k c[k] \tilde{\mathbf{f}}_k \end{aligned} \quad (2.280)$$

where we have defined $\tilde{\mathbf{f}}_k$, called the *dual vector* of \mathbf{f}_k , as:

$$\tilde{\mathbf{f}}_k = (FF^*)^{-1} \mathbf{f}_k, \quad \text{i.e.} \quad \mathbf{f}_k = (FF^*) \tilde{\mathbf{f}}_k \quad (2.281)$$

Note that $(FF^*)^{-1}F = (F^*)^-$ above is actually the pseudo-inverse of F^* satisfying:

$$(F^*)^-F^* = (FF^*)^{-1}FF^* = I \quad (2.282)$$

We can define $(F^*)^-$ as another transformation:

$$\tilde{F} = (FF^*)^{-1}F = (F^*)^- \quad (2.283)$$

and rewrite Eq.2.280 as:

$$\mathbf{x} = \tilde{F}\mathbf{c} = \tilde{F}[\dots, c[k], \dots]^T = \sum_k \langle \mathbf{x}, \mathbf{f}_k \rangle \tilde{\mathbf{f}}_k = \sum_k c[k] \tilde{\mathbf{f}}_k \quad (2.284)$$

This is the inverse frame transformation which reconstructs the vector \mathbf{x} based on the coefficients \mathbf{c} obtained by the forward frame transformation in Eq.2.276. Eqs.2.284 and 2.276 form a frame transformation pair, similar to the unitary transformation pair in Eq.2.204.

We can find the adjoint of \tilde{F} from the following inner product (by reversing the steps in Eq.2.277):

$$\begin{aligned} \langle \tilde{F}\mathbf{c}, \mathbf{x} \rangle &= \langle \sum_k c[k] \tilde{\mathbf{f}}_k, \mathbf{x} \rangle = \sum_k c[k] \langle \tilde{\mathbf{f}}_k, \mathbf{x} \rangle \\ &= \sum_k c[k] \overline{\langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle} = \langle \mathbf{c}, \tilde{F}^* \mathbf{x} \rangle \end{aligned} \quad (2.285)$$

Here \tilde{F}^* is the adjoint of \tilde{F} :

$$\tilde{F}^*\mathbf{x} = [\dots, \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle, \dots]^T = [\dots, d[k], \dots]^T = \mathbf{d} \quad (2.286)$$

where we have defined $d[k] = \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle$. Replacing F by F^* in Eq. 2.283, we get

$$\tilde{F}^* = (F^*F)^{-1}F^* = F^- \quad (2.287)$$

which is the pseudo-inverse of F satisfying:

$$\tilde{F}^*F = (F^*F)^{-1}F^*F = F^-F = I \quad (2.288)$$

Theorem 2.8. A vector $\mathbf{x} \in H$ can be equivalently represented by either of the two dual frames $\{\mathbf{f}_k\}$ or $\{\tilde{\mathbf{f}}_k\}$:

$$\mathbf{x} = \sum_k \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle \mathbf{f}_k = \sum_k \langle \mathbf{x}, \mathbf{f}_k \rangle \tilde{\mathbf{f}}_k \quad (2.289)$$

Proof: Consider the inner product $\langle \mathbf{x}, \mathbf{x} \rangle$, with the first \mathbf{x} replaced by the expression in Eq.2.280:

$$\begin{aligned} \langle \mathbf{x}, \mathbf{x} \rangle &= \langle \sum_k \langle \mathbf{x}, \mathbf{f}_k \rangle \tilde{\mathbf{f}}_k, \mathbf{x} \rangle = \sum_k \langle \mathbf{x}, \mathbf{f}_k \rangle \langle \tilde{\mathbf{f}}_k, \mathbf{x} \rangle \\ &= \langle \mathbf{x}, \sum_k \overline{\langle \tilde{\mathbf{f}}_k, \mathbf{x} \rangle} \mathbf{f}_k \rangle = \langle \mathbf{x}, \sum_k \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle \mathbf{f}_k \rangle \end{aligned} \quad (2.290)$$

Comparing the two sides of the equation, we get:

$$\mathbf{x} = \sum_k \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle \mathbf{f}_k = \sum_k d[k] \mathbf{f}_k \quad (2.291)$$

Combining this result with Eq.2.280, we get Eq.2.289. Q.E.D.

Note that according to Eq.2.278 Eq.2.291 can also be written as:

$$\mathbf{x} = \sum_k \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle \mathbf{f}_k = \sum_k d[k] \mathbf{f}_k = F\mathbf{d} \quad (2.292)$$

We can combine Eqs.2.276 and 2.286 together with Eq.2.289 to form two alternative frame transformation pairs based on either frame $\{\mathbf{f}_k\}$ or its dual $\{\tilde{\mathbf{f}}_k\}$:

$$\begin{cases} c[k] = \langle \mathbf{x}, \mathbf{f}_k \rangle \\ \mathbf{x} = \sum_k c[k] \mathbf{f}_k = \sum_k \langle \mathbf{x}, \mathbf{f}_k \rangle \tilde{\mathbf{f}}_k \end{cases} \quad \begin{cases} d[k] = \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle \\ \mathbf{x} = \sum_k d[k] \mathbf{f}_k = \sum_k \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle \mathbf{f}_k \end{cases} \quad (2.293)$$

These equations are respectively the forward and inverse frame transformation of \mathbf{x} based on the frame $\{\mathbf{f}_k\}$ and its dual $\{\tilde{\mathbf{f}}_k\}$, which can also be expressed (due to Eqs.2.284 and 2.292) more concisely as:

$$\begin{cases} \mathbf{c} = F^* \mathbf{x} \\ \mathbf{x} = \tilde{F} \mathbf{c} = (F^*)^- \mathbf{c} \end{cases} \quad \begin{cases} \mathbf{d} = \tilde{F}^* \mathbf{x} = F^- \mathbf{x} \\ \mathbf{x} = F \mathbf{d} \end{cases} \quad (2.294)$$

The frame transformation pairs in Eqs.2.294 can be considered as a generalization of the unitary transformation given in Eq.2.204, which is carried out by U and its inverse U^{-1} , while the frame transformation pairs are carried out by F^* (or F) and its pseudo-inverse $\tilde{F} = (F^*)^-$ (or $\tilde{F}^* = F^-$). We also see from Eq.2.294 that

$$\tilde{F} F^* \mathbf{x} = F \tilde{F}^* \mathbf{x} = \mathbf{x} \quad (2.295)$$

i.e., $\tilde{F} F^* = (F^*)^- F^* = I$ and $F \tilde{F}^* = F^- F = I$, similar to $U^{-1} U = U^* U = I$.

Also, similar to the unitary transformation, the signal energy is conserved by the frame transformation:

$$\begin{aligned} \|\mathbf{x}\|^2 &= \langle \mathbf{x}, \mathbf{x} \rangle = \langle \tilde{F} \mathbf{c}, \mathbf{x} \rangle = \langle \mathbf{c}, \tilde{F}^* \mathbf{x} \rangle = \langle \mathbf{c}, \mathbf{d} \rangle \\ &= \langle F \mathbf{d}, \mathbf{x} \rangle = \langle \mathbf{d}, F^* \mathbf{x} \rangle = \langle \mathbf{d}, \mathbf{c} \rangle \end{aligned} \quad (2.296)$$

This relationship can be considered as the generalized version of Parseval's identity. However, we note that:

$$\begin{aligned} \|\mathbf{c}\|^2 &= \langle \mathbf{c}, \mathbf{c} \rangle = \langle F^* \mathbf{x}, F^* \mathbf{x} \rangle = \langle F F^* \mathbf{x}, \mathbf{x} \rangle \neq \langle \mathbf{x}, \mathbf{x} \rangle = \|\mathbf{x}\|^2 \\ \|\mathbf{d}\|^2 &= \langle \mathbf{d}, \mathbf{d} \rangle = \langle \tilde{F}^* \mathbf{x}, \tilde{F}^* \mathbf{x} \rangle = \langle \tilde{F} \tilde{F}^* \mathbf{x}, \mathbf{x} \rangle \neq \langle \mathbf{x}, \mathbf{x} \rangle = \|\mathbf{x}\|^2 \end{aligned} \quad (2.297)$$

To find out how the signal energy is related to the energy contained in either of the two sets of coefficients, we need to study further the operator FF^* . Consider

the inner product of Eq.2.279 and another vector \mathbf{y} :

$$\begin{aligned} \langle FF^*\mathbf{x}, \mathbf{y} \rangle &= \sum_k \langle \mathbf{x}, \mathbf{f}_k \rangle \langle \mathbf{f}_k, \mathbf{y} \rangle = \langle \mathbf{x}, \sum_k \overline{\langle \mathbf{f}_k, \mathbf{y} \rangle} \mathbf{f}_k \rangle \\ &= \langle \mathbf{x}, \sum_k \langle \mathbf{y}, \mathbf{f}_k \rangle \mathbf{f}_k \rangle = \langle \mathbf{x}, FF^*\mathbf{y} \rangle \end{aligned} \quad (2.298)$$

which indicates that FF^* is a self-adjoint operator. If we let $\{\lambda_k\}$ and $\{\phi_k\}$ be the eigenvalues and eigenvectors of FF^* , i.e.,

$$FF^*\phi_k = \lambda_k \phi_k, \quad (\text{for all } k) \quad (2.299)$$

then all $\{\lambda_k\}$ are real, and all $\{\phi_k\}$ are orthogonal $\langle \phi_k, \phi_l \rangle = \delta[k - l]$ and they form a complete orthogonal system (Theorem 2.4). Now \mathbf{x} can also be expanded in terms of these eigenvectors as:

$$\mathbf{x} = \sum_k \langle \mathbf{x}, \phi_k \rangle \phi_k \quad (2.300)$$

and the energy contained in \mathbf{x} is:

$$\begin{aligned} \|\mathbf{x}\|^2 &= \langle \mathbf{x}, \mathbf{x} \rangle = \langle \sum_k \langle \mathbf{x}, \phi_k \rangle \phi_k, \sum_l \langle \mathbf{x}, \phi_l \rangle \phi_l \rangle \\ &= \sum_k \sum_l \langle \mathbf{x}, \phi_k \rangle \overline{\langle \mathbf{x}, \phi_l \rangle} \langle \phi_k, \phi_l \rangle = \sum_k |\langle \mathbf{x}, \phi_k \rangle|^2 \end{aligned} \quad (2.301)$$

For the dual frame transformation \tilde{F} , we have:

$$\tilde{F}\tilde{F}^* = [(FF^*)^{-1}F][(FF^*)^{-1}F]^* = (FF^*)^{-1}FF^*(FF^*)^{-1} = (FF^*)^{-1} \quad (2.302)$$

which is also a self-adjoint operator whose eigenvalues and eigenvectors are respectively $\{1/\lambda_k\}$ and $\{\phi_k\}$, i.e.,:

$$\tilde{F}\tilde{F}^*\phi_k = (FF^*)^{-1}\phi_k = \frac{1}{\lambda_k}\phi_k, \quad (\text{for all } k) \quad (2.303)$$

Theorem 2.9. *The frame transformation coefficients $c[k] = \langle \mathbf{x}, \mathbf{f}_k \rangle$ and $d[k] = \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle$ satisfy respectively the following inequalities:*

$$\lambda_{min}\|\mathbf{x}\|^2 \leq \sum_k |\langle \mathbf{x}, \mathbf{f}_k \rangle|^2 = \|\mathbf{c}\|^2 = \|F^*\mathbf{x}\|^2 \leq \lambda_{max}\|\mathbf{x}\|^2 \quad (2.304)$$

$$\frac{1}{\lambda_{max}}\|\mathbf{x}\|^2 \leq \sum_k |\langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle|^2 = \|\mathbf{d}\|^2 = \|\tilde{F}^*\mathbf{x}\|^2 \leq \frac{1}{\lambda_{min}}\|\mathbf{x}\|^2 \quad (2.305)$$

where λ_{min} and λ_{max} are respectively the smallest and largest eigenvalues of the self-adjoint operator FF^* . When all eigenvalues are the same, then $\lambda_{max} = \lambda_{min} = \lambda$ and the frame is tight:

$$\sum_k |\langle \mathbf{x}, \mathbf{f}_k \rangle|^2 = \lambda\|\mathbf{x}\|^2, \quad \sum_k |\langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle|^2 = \frac{1}{\lambda}\|\mathbf{x}\|^2 \quad (2.306)$$

Proof: Applying $(FF^*)^{-1}$ to both sides of Eq.2.292 we get:

$$(FF^*)^{-1}\mathbf{x} = \sum_k \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle (FF^*)^{-1}\mathbf{f}_k = \sum_k \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle \tilde{\mathbf{f}}_k \quad (2.307)$$

This result and Eq.2.279 form a symmetric pair:

$$(FF^*)\mathbf{x} = \sum_k \langle \mathbf{x}, \mathbf{f}_k \rangle \mathbf{f}_k \quad (2.308)$$

$$(FF^*)^{-1}\mathbf{x} = \sum_k \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle \tilde{\mathbf{f}}_k \quad (2.309)$$

Taking the inner product of each of these equations with \mathbf{x} , we get:

$$\begin{aligned} \langle (FF^*)\mathbf{x}, \mathbf{x} \rangle &= \sum_k \langle \mathbf{x}, \mathbf{f}_k \rangle \langle \mathbf{f}_k, \mathbf{x} \rangle = \sum_k |\langle \mathbf{x}, \mathbf{f}_k \rangle|^2 \\ &= \sum_k |c[k]|^2 = \|\mathbf{c}\|^2 \end{aligned} \quad (2.310)$$

$$\begin{aligned} \langle (FF^*)^{-1}\mathbf{x}, \mathbf{x} \rangle &= \sum_k \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle \langle \tilde{\mathbf{f}}_k, \mathbf{x} \rangle = \sum_k |\langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle|^2 \\ &= \sum_k |d[k]|^2 = \|\mathbf{d}\|^2 \end{aligned} \quad (2.311)$$

These two expressions represent the energy contained in each of the two sets of coefficients $c[k] = \langle \mathbf{x}, \mathbf{f}_k \rangle$ and $d[k] = \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle$.

We will now carry out the following two parallel steps. First, we apply FF^* to both sides of Eq.2.300:

$$\begin{aligned} FF^*\mathbf{x} &= FF^*\left(\sum_k \langle \mathbf{x}, \phi_k \rangle \phi_k\right) = \sum_k \langle \mathbf{x}, \phi_k \rangle FF^*\phi_k \\ &= \sum_k \langle \mathbf{x}, \phi_k \rangle \lambda_k \phi_k \end{aligned} \quad (2.312)$$

and take inner product with \mathbf{x} on both sides:

$$\begin{aligned} \langle FF^*\mathbf{x}, \mathbf{x} \rangle &= \langle \sum_k \langle \mathbf{x}, \phi_k \rangle \lambda_k \phi_k, \mathbf{x} \rangle = \sum_k \langle \mathbf{x}, \phi_k \rangle \lambda_k \langle \phi_k, \mathbf{x} \rangle \\ &= \sum_k \lambda_k |\langle \mathbf{x}, \phi_k \rangle|^2 \end{aligned} \quad (2.313)$$

Replacing the left hand side by Eq.2.310, we get:

$$\sum_k |\langle \mathbf{x}, \mathbf{f}_k \rangle|^2 = \sum_k \lambda_k |\langle \mathbf{x}, \phi_k \rangle|^2 \quad (2.314)$$

Applying Eq.2.301 to the right-hand side we get:

$$\lambda_{min} \|\mathbf{x}\|^2 \leq \sum_k |\langle \mathbf{x}, \mathbf{f}_k \rangle|^2 \leq \lambda_{max} \|\mathbf{x}\|^2 \quad (2.315)$$

Next, we apply $(FF^*)^{-1}$ to both sides of Eq.2.300:

$$(FF^*)^{-1}\mathbf{x} = \sum_k \langle \mathbf{x}, \phi_k \rangle (FF^*)^{-1}\phi_k = \sum_k \langle \mathbf{x}, \phi_k \rangle \frac{1}{\lambda_k} \phi_k \quad (2.316)$$

and take inner product with \mathbf{x} on both sides:

$$\langle (FF^*)^{-1}\mathbf{x}, \mathbf{x} \rangle = \sum_k \langle \mathbf{x}, \phi_k \rangle \frac{1}{\lambda_k} \langle \phi_k, \mathbf{x} \rangle = \sum_k \frac{1}{\lambda_k} |\langle \mathbf{x}, \phi_k \rangle|^2 \quad (2.317)$$

Replacing the left hand side by Eq.2.311, we get:

$$\sum_k |\langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle|^2 = \sum_k \frac{1}{\lambda_k} |\langle \mathbf{x}, \phi_k \rangle|^2 \quad (2.318)$$

Applying Eq.2.301 to the right-hand side we get:

$$\frac{1}{\lambda_{max}} \|\mathbf{x}\|^2 \leq \sum_k |\langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle|^2 \leq \frac{1}{\lambda_{min}} \|\mathbf{x}\|^2 \quad (2.319)$$

Q.E.D.

This theorem indicates that the frame transformation associated with either F or \tilde{F} does not conserve signal energy, due obviously to the redundancy of the non-independent frame vectors. However, as shown in Eq.2.296, the energy is conserved when both sets of coefficients are involved.

Theorem 2.10. *Let λ_k and ϕ_k be the k th eigenvalue and the corresponding eigenvector of operator FF^* : $FF^*\phi_k = \lambda_k\phi_k$ for all k , Then*

$$\sum_k \lambda_k = \sum_k \|\mathbf{f}_k\|^2, \quad \sum_k \frac{1}{\lambda_k} = \sum_k \|\tilde{\mathbf{f}}_k\|^2 \quad (2.320)$$

Proof: As FF^* is self-adjoint its eigenvalues λ_k 's are real and its eigenfunctions are orthogonal $\langle \phi_k, \phi_l \rangle = \delta[k - l]$, we therefore have:

$$\begin{aligned} \sum_k \lambda_k &= \sum_k \lambda_k \langle \phi_k, \phi_k \rangle = \sum_k \langle FF^*\phi_k, \phi_k \rangle \\ &= \sum_k \left\langle \sum_k \langle \phi_k, \mathbf{f}_k \rangle \mathbf{f}_k, \phi_k \right\rangle = \sum_k \sum_k |\langle \mathbf{f}_k, \phi_k \rangle|^2 \end{aligned} \quad (2.321)$$

On the other hand:

$$\begin{aligned} \|\mathbf{f}_k\|^2 &= \langle \mathbf{f}_k, \mathbf{f}_k \rangle = \left\langle \sum_k \langle \mathbf{f}_k, \phi_k \rangle \phi_k, \sum_l \langle \mathbf{f}_k, \phi_l \rangle \phi_l \right\rangle \\ &= \sum_k \sum_l \langle \mathbf{f}_k, \phi_k \rangle \overline{\langle \mathbf{f}_k, \phi_l \rangle} \langle \phi_k, \phi_l \rangle = \sum_k |\langle \mathbf{f}_k, \phi_k \rangle|^2 \end{aligned} \quad (2.322)$$

Therefore we get

$$\sum_k \|\mathbf{f}_k\|^2 = \sum_k \sum_k |\langle \mathbf{f}_k, \phi_k \rangle|^2 = \sum_k \lambda_k \quad (2.323)$$

The second equation in the theorem can be similarly proved. Q.E.D.

Definition 2.25. If the vectors in a frame are linearly independent, the frame is called a Riesz basis.

Theorem 2.11. (Biorthogonality of Riesz basis) A Riesz basis $\{\mathbf{f}_k\}$ and its dual $\{\tilde{\mathbf{f}}_k\}$ form a pair of biorthogonal bases satisfying

$$\langle \mathbf{f}_k, \tilde{\mathbf{f}}_l \rangle = \delta[k-l], \quad k, l \in \mathbb{Z} \quad (2.324)$$

Proof: We let $\mathbf{x} = \mathbf{f}_l$ in Eq.2.289 and get:

$$\mathbf{f}_l = \sum_k \langle \mathbf{f}_l, \tilde{\mathbf{f}}_k \rangle \mathbf{f}_k \quad (2.325)$$

Since these vectors are linearly independent, i.e., \mathbf{f}_l cannot be expressed as a linear combination of the rest of the frame vectors, the equation above has only one interpretation: all coefficients $\langle \mathbf{f}_l, \tilde{\mathbf{f}}_k \rangle = 0$ for all $k \neq l$ except when $k = l$ and $\langle \mathbf{f}_k, \tilde{\mathbf{f}}_k \rangle = 1$. In other words, these frame vectors are orthogonal to their dual vectors, i.e., Eq.2.325 holds. Q.E.D.

If the dual frames \mathbf{f} and $\tilde{\mathbf{f}}$ in Theorem 2.8 are a pair of biorthogonal bases, then Eq.2.289 is a *biorthogonal transformation*:

$$\mathbf{x} = \sum_k \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle \mathbf{f}_k = \sum_k \langle \mathbf{x}, \mathbf{f}_k \rangle \tilde{\mathbf{f}}_k \quad (2.326)$$

In summary, we see that signal representation by a set of linearly independent and orthogonal basis vectors $\mathbf{x} = \sum_k c[k] \phi_k = \sum_k \langle \mathbf{x}, \mathbf{b}_k \rangle \mathbf{x}_k$ (Eq.2.88) is now much generalized so that the signal is represented by a set of frame vectors, which are in general neither linearly independent nor orthogonal. The representation can be in either of the two dual frames, and the frame transformation and its inverse are pseudo-inverse of each other. Moreover, the signal energy is no longer conserved by the transformation, as Parseval's identity is invalid due to the redundancy in the frame. Instead, the signal energy and the energy in the coefficients are related by Eqs.2.304, 2.305, and 2.296.

On the other hand, we can consider the unitary transformation U as a special kind of frame transformation F . As $UU^* = U^*U = I$ in Eq.2.281, the pseudo-inverse in Eq.2.288 $U^- = (U^*U)^{-1}U^* = U^* = U^{-1}$ becomes a regular inverse, $U = \tilde{U}$ becomes the same as its dual, i.e., $\mathbf{u}_k = \tilde{\mathbf{u}}_k$, and the biorthogonality in Eq.2.324 becomes regular orthogonality. Consequently, the two dual transformation pairs in Eq.2.293 (or 2.294) become identical, a unitary transformation pair.

Also, corresponding to the eigenequations of operators FF^* (Eq.2.299) and $\tilde{F}\tilde{F}^*$ (Eq.2.303), the eigenequation of operator $UU^* = I$ becomes a trivial case:

$$UU^*\phi_k = U^*U\phi_k = I\phi_k = \lambda_k\phi_k = \phi_k \quad (2.327)$$

with $\lambda_{max} = \lambda_{min} = \lambda_k = 1$ (for all k) and both Eqs.2.304 and 2.305 (as well as Eqs.2.296) become Parseval's identity (Eq.2.188):

$$\|\mathbf{x}\|^2 = \sum_k |\langle \mathbf{x}, \mathbf{u}_k \rangle|^2 \quad (2.328)$$

2.4.3 Frames in Finite-Dimensional Space

Here we consider the frame transformation in \mathbb{C}^N . Let $\mathbf{F} = [\mathbf{f}_1, \dots, \mathbf{f}_M]$ be a matrix composed of a set of M frame vectors as its columns. We assume $M > N$, and the M frame vectors are obviously not independent. The dual frame is also an matrix composed of M dual vectors as its columns $\tilde{\mathbf{F}} = [\tilde{\mathbf{f}}_1, \dots, \tilde{\mathbf{f}}_M]$. Any given vector $\mathbf{x} \in \mathbb{C}^N$ can now be represented by either the frame \mathbf{F} (second transformation pair in Eq.2.294) or its dual $\tilde{\mathbf{F}}$ (first transformation pair in Eq.2.294), in the form of a matrix multiplication (e.g., the generic operator F becomes a matrix \mathbf{F}):

$$\begin{cases} \mathbf{c} = \mathbf{F}^* \mathbf{x} \\ \mathbf{x} = \tilde{\mathbf{F}} \mathbf{c} = (\mathbf{F}^*)^- \mathbf{c} \end{cases} \quad \begin{cases} \mathbf{d} = \tilde{\mathbf{F}}^* \mathbf{x} = \mathbf{F}^- \mathbf{x} \\ \mathbf{x} = \mathbf{F} \mathbf{d} \end{cases} \quad (2.329)$$

These frame transformations are in the same form as the unitary transformations in Eq.2.100. However, different from matrices \mathbf{U} and $\mathbf{U}^* = \mathbf{U}^{-1}$ in Eq.2.100, here matrices \mathbf{F} and $\tilde{\mathbf{F}}$ in Eq.2.329 are not invertible as they are not square matrices (\mathbf{F} and $\tilde{\mathbf{F}}$ are N by M while \mathbf{F}^* and $\tilde{\mathbf{F}}^*$ are M by N). Consequently, the matrices used in the forward and inverse frame transformations are pseudo-inverse of each other:

$$\mathbf{F}^- = (\mathbf{F}^* \mathbf{F})^{-1} \mathbf{F}^* = \tilde{\mathbf{F}}^*, \quad (\mathbf{F}^*)^- = (\mathbf{F} \mathbf{F}^*)^{-1} \mathbf{F} = \tilde{\mathbf{F}} \quad (2.330)$$

We first represent \mathbf{x} in terms of \mathbf{F} . Based on the second transformation in Eq.2.329, the coefficients \mathbf{d} can obtained as:

$$\mathbf{d} = \tilde{\mathbf{F}}^* \mathbf{x} = \begin{bmatrix} \tilde{\mathbf{f}}_1^* \\ \vdots \\ \tilde{\mathbf{f}}_M^* \end{bmatrix} \mathbf{x} = \begin{bmatrix} \langle \mathbf{x}, \tilde{\mathbf{f}}_1 \rangle \\ \vdots \\ \langle \mathbf{x}, \tilde{\mathbf{f}}_M \rangle \end{bmatrix} \quad (2.331)$$

and \mathbf{x} is reconstructed by the inverse transformation:

$$\mathbf{x} = \mathbf{F} \mathbf{d} = [\mathbf{f}_1, \dots, \mathbf{f}_M] \begin{bmatrix} \langle \mathbf{x}, \tilde{\mathbf{f}}_1 \rangle \\ \vdots \\ \langle \mathbf{x}, \tilde{\mathbf{f}}_M \rangle \end{bmatrix} = \sum_{k=1}^M \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle \mathbf{f}_k \quad (2.332)$$

Alternatively, we can also represent \mathbf{x} in terms of the dual frame $\tilde{\mathbf{F}}$. Based on the first transformation in Eq.2.329, the coefficients \mathbf{c} can be obtained as:

$$\mathbf{c} = \mathbf{F}^* \mathbf{x} = \begin{bmatrix} \mathbf{f}_1^* \\ \vdots \\ \mathbf{f}_M^* \end{bmatrix} \mathbf{x} = \begin{bmatrix} \langle \mathbf{x}, \mathbf{f}_1 \rangle \\ \vdots \\ \langle \mathbf{x}, \mathbf{f}_M \rangle \end{bmatrix} \quad (2.333)$$

and \mathbf{x} is reconstructed by the inverse transformation:

$$\mathbf{x} = \tilde{\mathbf{F}} \mathbf{c} = [\tilde{\mathbf{f}}_1, \dots, \tilde{\mathbf{f}}_M] \begin{bmatrix} \langle \mathbf{x}, \mathbf{f}_1 \rangle \\ \vdots \\ \langle \mathbf{x}, \mathbf{f}_M \rangle \end{bmatrix} = \sum_{k=1}^M \langle \mathbf{x}, \mathbf{f}_k \rangle \tilde{\mathbf{f}}_k \quad (2.334)$$

Theorem 2.12. If a frame $\mathbf{F} = [\mathbf{f}_1, \dots, \mathbf{f}_M]$ in \mathbb{C}^N is tight, i.e., all eigenvalues $\lambda_k = \lambda$ of $\mathbf{F}\mathbf{F}^*$ are the same, and all frame vectors are normalized $\|\mathbf{f}_k\| = 1$, then the frame bound is M/N .

Proof: As $\mathbf{F}\mathbf{F}^*$ is an N by N matrix, it has N eigenvalues $\lambda_k = \lambda$ ($k = 1, \dots, N$). Then Theorem 2.10 becomes:

$$\sum_{k=1}^N \lambda_k = N\lambda = \sum_{k=1}^M \|\mathbf{f}_k\|^2 = M \quad (2.335)$$

i.e., $\lambda = M/N$. Q.E.D.

In particular, if $M = N$ linearly independent frame vectors are used, then they form a Riesz basis in \mathbb{C}^N , and $\mathbf{F} = [\mathbf{f}_1, \dots, \mathbf{f}_N]$ becomes an N by N invertible matrix, and its pseudo-inverse is just a regular inverse, and the second equation in Eq.2.330 becomes $(\mathbf{F}^*)^{-1} = \tilde{\mathbf{F}}$, i.e.,

$$\mathbf{F}^* \tilde{\mathbf{F}} = \begin{bmatrix} \mathbf{f}_1^* \\ \vdots \\ \mathbf{f}_N^* \end{bmatrix} [\tilde{\mathbf{f}}_1, \dots, \tilde{\mathbf{f}}_N] = \mathbf{I} \quad (2.336)$$

which indicates that these Riesz vectors are indeed biorthogonal:

$$\langle \mathbf{f}_k, \tilde{\mathbf{f}}_l \rangle = \delta[k - l], \quad (k, l = 1, \dots, N) \quad (2.337)$$

Moreover, if these N vectors are also orthogonal, i.e., $\langle \mathbf{f}_k, \mathbf{f}_l \rangle = \delta[k - l]$, then $\mathbf{F} = \mathbf{U}$ becomes a unitary matrix satisfying $\mathbf{U}^* = \mathbf{U}^{-1}$, and $\tilde{\mathbf{U}} = (\mathbf{U}^*)^{-1} = \mathbf{U}$, i.e., the vectors are the dual of their own, and they form an orthonormal basis of \mathbb{C}^N . Now the frame transformation becomes a unitary transformation $\mathbf{U}^* \mathbf{x} = \mathbf{c}$ and the inverse is simply $\mathbf{U}\mathbf{c} = \mathbf{x}$. Also the eigenvalues of $\mathbf{U}\mathbf{U}^* = \mathbf{I}$ are all $\lambda_k = 1$, and $\|\mathbf{u}_k\|^2 = 1$, Theorem 2.10 holds trivially.

Example 2.9: Three normalized vectors in \mathbb{R}^2 form a frame:

$$\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3] = \begin{bmatrix} -1 & 1/2 & 1/2 \\ 0 & \sqrt{3}/2 & -\sqrt{3}/2 \end{bmatrix} \quad (2.338)$$

Note that these frame vectors are normalized $\|\mathbf{f}_k\| = 1$. We also have:

$$\mathbf{F}\mathbf{F}^T = \frac{3}{2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad (\mathbf{F}\mathbf{F}^T)^{-1} = \frac{2}{3} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (2.339)$$

The eigenvalues of these two matrices are obviously $\lambda_1 = \lambda_2 = 3/2$ and $1/\lambda_1 = 1/\lambda_2 = 2/3$, respectively, indicating this is a tight frame $A = B$. The dual frame $\tilde{\mathbf{F}}$ can be found as the pseudo-inverse of \mathbf{F}^T :

$$\tilde{\mathbf{F}} = [\tilde{\mathbf{f}}_1, \tilde{\mathbf{f}}_2, \tilde{\mathbf{f}}_3] = (\mathbf{F}\mathbf{F}^T)^{-1}\mathbf{F} = \frac{2}{3}\mathbf{F} = \begin{bmatrix} -2/3 & 1/3 & 1/3 \\ 0 & \sqrt{3}/3 & -\sqrt{3}/3 \end{bmatrix} \quad (2.340)$$

Any given $\mathbf{x} = [x[1], x[2]]^T$ can be expanded in terms of either of the two frames:

$$\mathbf{x} = \sum_{k=1}^3 c[k]\mathbf{f}_k = \sum_{k=1}^3 \langle \mathbf{x}, \mathbf{f}_k \rangle \tilde{\mathbf{f}}_k = \sum_{k=1}^3 d[k]\mathbf{f}_k = \sum_{k=1}^3 \langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle \mathbf{f}_k \quad (2.341)$$

where $\mathbf{c} = \mathbf{F}^*\mathbf{x}$ or

$$c[1] = -x[1], \quad c[2] = \frac{1}{2}[x[1] + \sqrt{3}x[2]], \quad c[3] = \frac{1}{2}[x[1] - \sqrt{3}x[2]] \quad (2.342)$$

and $\mathbf{d} = \tilde{\mathbf{F}}^*\mathbf{x}$ or

$$d[1] = -\frac{2}{3}x[1], \quad d[2] = \frac{1}{3}[x[1] + \sqrt{3}x[2]], \quad d[3] = \frac{1}{3}[x[1] - \sqrt{3}x[2]] \quad (2.343)$$

The energy contained in the coefficients \mathbf{c} and \mathbf{d} is respectively:

$$\|\mathbf{c}\|^2 = \sum_{k=1}^3 |\langle \mathbf{x}, \mathbf{f}_k \rangle|^2 = \frac{3}{2}\|\mathbf{x}\|^2 = \lambda\|\mathbf{x}\|^2 \quad (2.344)$$

and

$$\|\mathbf{d}\|^2 = \sum_{k=1}^3 |\langle \mathbf{x}, \tilde{\mathbf{f}}_k \rangle|^2 = \frac{2}{3}\|\mathbf{x}\|^2 = \frac{1}{\lambda}\|\mathbf{x}\|^2 \quad (2.345)$$

Specifically if we let $\mathbf{x} = [1, 2]^T$, then

$$\mathbf{c} = \mathbf{F}^T\mathbf{x} = \begin{bmatrix} \mathbf{f}_1^T \\ \mathbf{f}_2^T \\ \mathbf{f}_3^T \end{bmatrix} \mathbf{x} = \begin{bmatrix} \langle \mathbf{x}, \mathbf{f}_1 \rangle \\ \langle \mathbf{x}, \mathbf{f}_2 \rangle \\ \langle \mathbf{x}, \mathbf{f}_3 \rangle \end{bmatrix} = \begin{bmatrix} -1 \\ 1 + \sqrt{3} \\ 1 - \sqrt{3} \end{bmatrix} \quad (2.346)$$

and

$$\mathbf{d} = \tilde{\mathbf{F}}^T\mathbf{x} = \begin{bmatrix} \tilde{\mathbf{f}}_1^T \\ \tilde{\mathbf{f}}_2^T \\ \tilde{\mathbf{f}}_3^T \end{bmatrix} \mathbf{x} = \begin{bmatrix} \langle \mathbf{x}, \tilde{\mathbf{f}}_1 \rangle \\ \langle \mathbf{x}, \tilde{\mathbf{f}}_2 \rangle \\ \langle \mathbf{x}, \tilde{\mathbf{f}}_3 \rangle \end{bmatrix} = \frac{2}{3} \begin{bmatrix} -1 \\ 1 + \sqrt{3} \\ 1 - \sqrt{3} \end{bmatrix} \quad (2.347)$$

Example 2.10: Vectors \mathbf{f}_1 and \mathbf{f}_2 form a basis that spans the 2-D space:

$$\mathbf{f}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \mathbf{f}_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \quad \mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2] = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \quad (2.348)$$

$$\mathbf{F}\mathbf{F}^T = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}, \quad (\mathbf{F}\mathbf{F}^T)^{-1} = \begin{bmatrix} 1 & -1 \\ -1 & 2 \end{bmatrix} \quad (2.349)$$

The dual frame can be found to be:

$$\tilde{\mathbf{F}} = (\mathbf{F}\mathbf{F}^T)^{-1}\mathbf{F} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \quad \text{i.e.} \quad \tilde{\mathbf{f}}_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \quad \tilde{\mathbf{f}}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (2.350)$$

Obviously the biorthogonality condition in Eq.2.324 is satisfied by these two sets of bases. Next, to represent a vector $\mathbf{x} = [0, 2]^T$ by each of the two bases, we find the coefficients as:

$$\begin{aligned} c[1] &= \langle \mathbf{x}, \tilde{\mathbf{f}}_1 \rangle = 2; & c[1] &= \langle \mathbf{x}, \tilde{\mathbf{f}}_2 \rangle = -2 \\ d[1] &= \langle \mathbf{x}, \mathbf{f}_1 \rangle = 0; & d[2] &= \langle \mathbf{x}, \mathbf{f}_2 \rangle = -2 \end{aligned}$$

Now we have:

$$\mathbf{x} = c[1]\mathbf{f}_1 + c[2]\mathbf{f}_2 = 2 \begin{bmatrix} 1 \\ 0 \end{bmatrix} - 2 \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -2 \end{bmatrix} \quad (2.351)$$

or

$$\mathbf{x} = d[1]\tilde{\mathbf{f}}_1 + d[2]\tilde{\mathbf{f}}_2 = -2 \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -2 \end{bmatrix} \quad (2.352)$$

2.5 Kernel Function and Mercer's Theorem

Definition 2.26. A kernel is a function that maps two continuous variable t, τ to a complex value $K(t, \tau) \in \mathbb{C}$. If the two variables are truncated and sampled to become discrete t_m, t_n ($m, n = 1, \dots, N$), then the kernel can be represented by an N by N matrix with the mn -th element being $K(t_m, t_n) = K[m, n]$. If $K(t, \tau) = \overline{K(\tau, t)}$ or $K[m, n] = \overline{K[n, m]}$, the kernel is Hermitian (self-adjoint).

Definition 2.27. A continuous kernel $K(t, \tau)$ is positive definite if the following holds for any function $x(t)$ defined over $[a, b]$:

$$\int_a^b \int_a^b x(t)K(t, \tau)\overline{x}(\tau)d\tau dt > 0 \quad (2.353)$$

A discrete kernel $K[m, n]$ is positive definite if the following holds for any vector $\mathbf{x} = [x[1], \dots, x[N]]$:

$$\sum_{m=1}^N \sum_{n=1}^N x[m] K[m, n] \bar{x}[n] > 0 \quad (2.354)$$

Definition 2.28. An operator T_K associated with a continuous kernel $K(t, \tau)$ is defined as:

$$T_K \mathbf{x}(t) = \int_a^b K(t, \tau) x(\tau) d\tau = \mathbf{y}(t) \quad (2.355)$$

An operator T_K associated with a discrete kernel $K[m, n]$ is a matrix:

$$T_K = \mathbf{T} = \begin{bmatrix} K[1, 1] & K[1, 2] & \cdots & K[1, N] \\ K[2, 1] & K[2, 2] & \cdots & K[2, N] \\ \vdots & \vdots & \ddots & \vdots \\ K[N, 1] & K[N, 2] & \cdots & K[N, N] \end{bmatrix} \quad (2.356)$$

which can be applied to a vector \mathbf{x} to generate $T_K \mathbf{x} = \mathbf{T} \mathbf{x} = \mathbf{y}$, or in component form:

$$\sum_{m=1}^N K[m, n] x[m] = y[n], \quad (n = 1, \dots, N) \quad (2.357)$$

Theorem 2.13. The operator T_K associated with a Hermitian kernel is Hermitian (self-adjoint):

$$\langle T_K \mathbf{x}(t), \mathbf{y}(t) \rangle = \langle \mathbf{x}(t), T_K \mathbf{y}(t) \rangle \quad (2.358)$$

Proof: For operator T_K associated with a continuous kernel, we have:

$$\begin{aligned} \langle T_K \mathbf{x}(t), \mathbf{y}(t) \rangle &= \int_a^b T_K \mathbf{x}(t) \bar{\mathbf{y}}(t) dt = \int_a^b \left[\int_a^b K(t, \tau) x(\tau) d\tau \right] \bar{\mathbf{y}}(t) dt \\ &= \int_a^b \left[\int_a^b \bar{K}(\tau, t) \bar{\mathbf{y}}(t) d\tau \right] x(\tau) d\tau = \int_a^b x(\tau) \bar{T_K \mathbf{y}(\tau)} d\tau = \langle \mathbf{x}(t), T_K \mathbf{y}(t) \rangle \end{aligned} \quad (2.359)$$

For operator $T_K = \mathbf{T}$ associated with a discrete kernel, we have:

$$\begin{aligned} \langle \mathbf{T} \mathbf{x}, \mathbf{y} \rangle &= \sum_{n=1}^N \left[\sum_{m=1}^N K[m, n] x[m] \right] \bar{y}[n] \\ &= \sum_{m=1}^N x[m] \left[\sum_{n=1}^N \bar{K}[m, n] \bar{y}[n] \right] = \langle \mathbf{x}, \mathbf{T} \mathbf{y} \rangle \end{aligned} \quad (2.360)$$

Q.E.D.

A self-adjoint operator T_K has all the properties stated in Theorem 2.4. Specifically, let λ_k be the k th eigenvalue of a self-adjoint operator T_K and $\phi_k(t)$ or ϕ_k be the corresponding eigenfunction or eigenvector:

$$\int_a^b K(t, \tau) \phi_k(\tau) d\tau = \lambda_k \phi_k(t), \quad \text{or} \quad T_K \phi_k = \mathbf{T} \phi_k = \lambda_k \phi_k \quad (2.361)$$

then we have:

1. All eigenvalues λ_k are real;
2. All eigenfunctions/eigenvectors are mutually orthogonal:

$$\langle \phi_k(t), \phi_l(t) \rangle = \langle \phi_k, \phi_l \rangle = \delta[k - l] \quad (2.362)$$

3. All eigenfunctions/eigenvectors form a complete orthogonal system, i.e., they form a basis that spans the function/vector space.

Theorem 2.14. (*Mercer's Theorem*) Let λ_k and $\phi_k(t)$ ($k = 1, 2, \dots$) be respectively the k th eigenvalue and the corresponding eigenfunction of the operator T_K associated with a positive definite Hermitian kernel $K(t, \tau)$, then the kernel can be expanded as:

$$K(t, \tau) = \sum_{k=1}^{\infty} \lambda_k \phi_k(t) \bar{\phi}_k(\tau) \quad (2.363)$$

Let λ_k and ϕ_k ($k = 1, 2, \dots$) be the k th eigenvalue and the corresponding eigenvector of the operator \mathbf{T} associated with a positive definite Hermitian kernel $K[m, n]$, then the kernel can be expanded as:

$$K[m, n] = \sum_{k=1}^N \lambda_k \phi[m, k] \bar{\phi}[n, k], \quad (m, n = 1, \dots, N) \quad (2.364)$$

where $\phi[m, k]$ is the m th element of the k th eigenvector $\phi_k = [\phi[1, k], \dots, \phi[N, k]]^T$.

The general proof of Mercer's theorem in Hilbert space is beyond the scope of this book and therefore omitted, but the discrete version in \mathbb{C}^N given in Eq.2.364 is simply the element form of Eq.2.164 for any Hermitian matrix:

$$\mathbf{T} = \sum_{k=1}^N \lambda_k \phi_k \phi_k^* \quad (2.365)$$

Note that given Eq.2.363 in Mercer's theorem, Eq.2.361 can be easily derived:

$$\begin{aligned} \int_a^b K(t, \tau) \phi_l(\tau) d\tau &= \int_a^b \left[\sum_{k=1}^{\infty} \lambda_k \phi_k(t) \bar{\phi}_k(\tau) \right] \phi_l(\tau) d\tau \\ &= \sum_{k=1}^{\infty} \lambda_k \phi_k(t) \int_a^b \bar{\phi}_k(\tau) \phi_l(\tau) d\tau = \sum_{k=1}^{\infty} \lambda_k \phi_k(t) \delta[k - l] = \lambda_l \phi_l(t) \end{aligned} \quad (2.366)$$

For example, consider the covariance of a centered stochastic process $x(t)$ with $\mu_x(t) = 0$:

$$\sigma_x^2(t, \tau) = E[x(t)\bar{x}(\tau)] = \overline{E[x(\tau)\bar{x}(t)]} = \bar{\sigma}_x^2(\tau, t) \quad (2.367)$$

which is a Hermitian kernel $K(t, \tau) = \sigma_x^2(t, \tau)$ that maps two variables t and τ to a complex value. Moreover, we can show that it is also positive definite:

$$\begin{aligned} &\int_a^b \int_a^b f(t) \sigma_x^2(t, \tau) \bar{f}(\tau) dt d\tau = \int_a^b \int_a^b E[f(t)x(t) \bar{f}(\tau)\bar{x}(\tau)] dt d\tau \\ &= E \left[\int_a^b f(t)x(t) dt \int_a^b \bar{f}(\tau)\bar{x}(\tau) d\tau \right] = E \left| \int_a^b f(t)x(t) dt \right|^2 > 0 \end{aligned} \quad (2.368)$$

Let T_K be the Hermitian integral operator associated with $\sigma_x^2(t, \tau) = \bar{\sigma}_x^2(\tau, t)$, its eigenequation is:

$$T_k \phi_k(t) = \int_a^b \sigma_x^2(t, \tau) \phi_k(\tau) dt = \lambda_k \phi_k(t), \quad k = 1, 2, \dots \quad (2.369)$$

where all eigenvalues $\lambda_k > 0$ are real and positive, and the eigenfunctions $\phi_k(t)$ are orthogonal:

$$\langle \phi_k(t), \phi_l(t) \rangle = \int_a^b \phi_k(t) \bar{\phi}_l(t) dt = \delta[k - l] \quad (2.370)$$

and they form a complete orthogonal basis that spans the vector space.

If the stochastic process $x(t)$ is truncated and sampled, it become a random vector $\mathbf{x} = [x[1], \dots, x[N]]^T$. The covariance between any two components $x[m]$ and $x[n]$ is

$$\sigma_{mn}^2 = E[x[m]\bar{x}[n]] = \overline{E[x[n]\bar{x}[m]]} = \bar{\sigma}_{nm}^2, \quad (m, n = 1, \dots, N) \quad (2.371)$$

which is a discrete Hermitian kernel, and the associated operator is the N by N covariance matrix:

$$\Sigma_x = E(\mathbf{x}\mathbf{x}^*) = \begin{bmatrix} \sigma_{11}^2 & \sigma_{12}^2 & \cdots & \sigma_{1N}^2 \\ \sigma_{21}^2 & \sigma_{22}^2 & \cdots & \sigma_{2N}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{N1}^2 & \sigma_{N2}^2 & \cdots & \sigma_{NN}^2 \end{bmatrix} \quad (2.372)$$

The eigenequation of this operator is:

$$\Sigma_x \phi_k = \lambda_k \phi_k, \quad (k = 1, \dots, N) \quad (2.373)$$

As $\Sigma^* = \Sigma$ is Hermitian (symmetric if x is real) and positive definite, its eigenvalues λ_k are all real positive, and the eigenvectors are orthogonal:

$$\langle \phi_k, \phi_l \rangle = \phi_k^T \bar{\phi}_l = \delta[k - l], \quad (k, l = 1, \dots, N) \quad (2.374)$$

and they form a unitary matrix $\Phi = [\phi_1, \dots, \phi_N]$ satisfying $\Phi^{-1} = \Phi^*$ i.e., $\Phi^* \Phi = I$. Eq.2.373 can also be written in the following forms:

$$\Sigma_x \Phi = \Phi \Lambda, \quad \Phi^* \Sigma_x \Phi = \Lambda, \quad \Sigma_x = \Phi \Lambda \Phi^* = \sum_{k=1}^N \lambda_k \phi_k \phi_k^* \quad (2.375)$$

Theorem 2.15. (*Karhunen-Loeve Theorem, continuous*) Let $\sigma_x^2(t, \tau)$ be the covariance of a centered stochastic process $x(t)$ with $\mu_x = E(x(t)) = 0$, and λ_k and $\phi_k(t)$ be respectively the k th eigenvalue and the corresponding eigenfunction of the integral operator associated with $\sigma_x^2(t, \tau)$ as a kernel:

$$T_K \phi_k(t) = \int_a^b \sigma_x^2(t, \tau) \phi_k(\tau) d\tau = \lambda_k \phi_k(t), \quad \text{for all } k \quad (2.376)$$

then $x(t)$ can be series expanded as:

$$x(t) = \sum_{k=1}^{\infty} c[k] \phi_k(t) \quad (2.377)$$

where $c[k]$ is the k th random coefficients given by

$$c[k] = \langle x(t), \phi_k(t) \rangle = \int_a^b x(t) \bar{\phi}_k(t) dt, \quad k = 1, 2, \dots \quad (2.378)$$

which are centered (zero mean) $E(c[k]) = 0$ and uncorrelated:

$$Cov(c[k], c[l]) = \lambda_k \delta[k - l] \quad (2.379)$$

Proof: As $\sigma_x^2(t, \tau)$ is self-adjoint, the eigenfunctions $\phi_k(t)$ of the associated operator T_K form a complete orthogonal basis, therefore any given stochastic process $x(t)$ can be represented as a linear combination of $\phi_k(t)$, i.e., Eq.2.377 holds.

Taking an inner product with $\phi_l(t)$ on both sides of Eq.2.377, we get:

$$\begin{aligned} \langle x(t), \phi_l(t) \rangle &= \int_a^b x(t) \bar{\phi}_l(t) dt = \sum_{k=1}^{\infty} c[k] \langle \phi_k(t), \phi_l(t) \rangle \\ &= \sum_{k=1}^{\infty} c[k] \delta[k - l] = c[l] \end{aligned} \quad (2.380)$$

This is Eq.2.378. The expectation of this equation is indeed zero:

$$E(c[k]) = E\left[\int_a^b x(t) \bar{\phi}_k(t) dt\right] = \int_a^b E[x(t)] \bar{\phi}_k(t) dt = 0 \quad (2.381)$$

Finally we show Eq.2.379 holds:

$$\begin{aligned}
 Cov(c[k], c[l]) &= E(c[k]\bar{c}[l]) = E\left[\int_a^b x(t)\bar{\phi}_k(t)dt \int_a^b \bar{x}(\tau)\phi_l(\tau)d\tau\right] \\
 &= \int_a^b \left[\int_a^b \phi_l(\tau)E[x(t)\bar{x}(\tau)]d\tau\right] \bar{\phi}_k(t)dt = \int_a^b \left[\int_a^b \phi_l(\tau)\sigma_x^2(t, \tau)d\tau\right] \bar{\phi}_k(t)dt \\
 &= \int_a^b \lambda_l \phi_l(t) \bar{\phi}_k(t)dt = \lambda_l \int_a^b \phi_l(t) \bar{\phi}_k(t)dt = \lambda_l \delta[k-l] = \lambda_k \delta[k-l]
 \end{aligned} \quad (2.382)$$

Q.E.D.

When the centered stochastic process $x(t)$ is truncated and sampled to become a finite random vector $\mathbf{x} = [x[1], \dots, x[N]]^T$ with $E(\mathbf{x}) = \boldsymbol{\mu}_x = 0$, the Karhunen-Loeve theorem takes the following discrete form:

Theorem 2.16. (*Karhunen-Loeve Theorem, discrete*) Let Σ_x be the covariance matrix of a centered random vector \mathbf{x} with $\boldsymbol{\mu}_x = E(\mathbf{x}) = \mathbf{0}$, and λ_k and ϕ_k be respectively the k th eigenvalue and the corresponding eigenvector of Σ_x :

$$\Sigma_x \phi_k = \lambda_k \phi_k, \quad k = 1, \dots, N \quad (2.383)$$

then \mathbf{x} can be series expanded as:

$$\mathbf{x} = \sum_{k=1}^N c[k] \phi_k \quad (2.384)$$

where $c[k]$ is the k th random coefficients given by

$$c[k] = \langle \mathbf{x}, \phi_k \rangle = \phi_k^* \mathbf{x}, \quad k = 1, \dots, N \quad (2.385)$$

which are centered (zero mean) $E(c[k]) = 0$ and uncorrelated:

$$Cov(c[k], c[l]) = \lambda_k \delta[k-l] \quad (2.386)$$

Proof: As the covariance matrix Σ_x is Hermitian and positive definite, its eigenvalues λ_k are all real positive and eigenvectors ϕ_k form a complete orthogonal system by which any \mathbf{x} can be series expanded as:

$$\mathbf{x} = \sum_{k=1}^N c[k] \phi_k = \Phi \mathbf{c} \quad (2.387)$$

where $\mathbf{c} = [c[1], \dots, c[N]]^T$ is a random vector formed by the N coefficients, and $\Phi = [\phi_1, \dots, \phi_N]^T$, i.e., Eq.2.387 holds.

To obtain these coefficients, we pre-multiply both sides by $\Phi^{-1} = \Phi^*$ to get:

$$\Phi^* \mathbf{x} = \Phi^* \Phi \mathbf{c} = \mathbf{c}, \quad \text{i.e.} \quad c[k] = \langle \mathbf{x}, \phi_k \rangle = \phi_k^* \mathbf{x}, \quad (k = 1, \dots, N) \quad (2.388)$$

The mean vector of \mathbf{c} is indeed zero:

$$\boldsymbol{\mu}_c = E(\mathbf{c}) = E(\Phi^* \mathbf{x}) = \Phi^* E(\mathbf{x}) = \mathbf{0} \quad (2.389)$$

and the covariance matrix of \mathbf{c} is:

$$\begin{aligned}\Sigma_c &= E(\mathbf{cc}^*) = E[(\Phi^*\mathbf{x})(\Phi^*\mathbf{x})^*] = E[\Phi^*\mathbf{x}\mathbf{x}^*\Phi] \\ &= \Phi^*E(\mathbf{x}\mathbf{x}^*)\Phi = \Phi^*\Sigma_x\Phi = \Lambda\end{aligned}\quad (2.390)$$

The covariance matrix $\Sigma_c = \Lambda$ is diagonalized:

$$\sigma_{kl}^2 = \lambda_k \delta[k - l], \quad (k, l = 1, \dots, N) \quad (2.391)$$

This is Eq.2.386. Q.E.D.

We see that the variance σ_k^2 of the kth coefficient $c[k]$ is the kth eigenvalue λ_k corresponding to the kth eigenvector ϕ_k , and the random signal \mathbf{x} is decorrelated by the transformation $\mathbf{c} = \Phi^*\mathbf{x}$ in Eq.2.388, as the components $c[k]$ and $c[l]$ of the resulting random signal \mathbf{c} are no longer correlated with $E(c[k]\bar{c}[l]) = 0$.

Comparing the generalized Fourier expansion in Eqs.2.107 and 2.109 with the Karhunen-Loeve series expansion in Eqs.2.377 and 2.378 we see that they are identical in form. However, we need to make it clear that the former is for a deterministic signal with a set of pre-determined basis functions $\phi_k(t)$; while the latter is for a stochastic signal, and the basis functions $\phi_k(t)$, are the eigenfunctions of the integral operator associated with the covariance function of the stochastic process, which are dependent on the specific signal being considered. Also note that Eqs.2.387 and 2.388 are simply the discrete versions of Eqs.2.377 and 2.378. The Karhunen-Loeve theorem and the associated series expansion will be considered in Chapter 9.

2.6 Summary

We summarize below the most essential points discussed in this chapter based on which the various orthogonal transform methods to be specifically discussed in following chapters will all be looked at from a unified point of view.

- A time signal can be considered as a vector $\mathbf{x} \in H$ in a Hilbert space, the specific type of which depends on the nature of the signal. For example, a continuous signal $x(t)$ over time interval $a < t < b$ is a vector $\mathbf{x} = x(t)$ in L^2 space; and its discrete samples is a vector $\mathbf{x} = [\dots, x[n], \dots]^T$ is a vector in l^2 space. Moreover, if the signal is truncated to become a set of N samples, then $\mathbf{x} = [x[1], \dots, x[N]]^T$ is a vector in \mathbb{C}^N space.
- The signal vector \mathbf{x} can be represented as a linear combination of a set of either countable basis \mathbf{b}_k or uncountable basis $\mathbf{b}(t)$ spanning the space in which it resides. In particular, if the basis is orthonormal, we have:

$$\mathbf{x} = \sum_k c[n]\mathbf{b}_k = \sum_k \langle \mathbf{x}, \mathbf{b}_k \rangle \mathbf{b}_k \quad (2.392)$$

or

$$\mathbf{x} = \int c(f)\mathbf{b}(f)df = \int \langle \mathbf{x}, \mathbf{b}(f) \rangle \mathbf{b}(f)df \quad (2.393)$$

Here $c[n] = \langle \mathbf{x}, \mathbf{b}_k \rangle$ or $c(f) = \langle \mathbf{x}, \mathbf{b}(f) \rangle$ is the weighting coefficient or function, representing the *analysis* of the signal by which the signal is decomposed into a set of components $c[n]\mathbf{b}_k$ or $c(f)\mathbf{b}(t)$, and the summation or integration is the *synthesis* of the signal by which the signal is reconstructed from its components.

- A signal vector given in the default form of a time function or a sequence of discrete values can be considered as a sequence of weighted and shifted time impulses (Eqs.1.3 and 1.9):

$$x(t) = \int x(\tau)\delta(t - \tau)d\tau, \quad (\text{for all } t) \quad (2.394)$$

or

$$x[n] = \sum_m x[m]\delta[m - n], \quad (\text{for all } m) \quad (2.395)$$

where $\delta(t - \tau)$ and $\delta[m - n]$ can be considered respectively as the standard basis of the corresponding signal space, which is always implicitly used in the default representation of a time signal. In other words, the default form of a signal $x(t)$ or $x[n]$ is actually a set of coefficients (countable) or weighting function (uncountable) of the standard basis vectors.

- The signal vector can be represented by any of the infinite bases all spanning the same space. For example, any unitary transformation of the standard basis will result a particular orthogonal basis, a rotated version of the standard basis. (The standard basis itself corresponds to an identity transformation.)

We here only consider orthogonal bases. For a continuous signal $x(t)$, we have:

$$\begin{aligned} x(t) &= \int c(f)\phi_f(t)df, \quad (\text{for all } t) \\ c(f) &= \langle x(t), \phi_f(t) \rangle = \int x(t)\overline{\phi}_f(t) dt, \quad (\text{for all } f) \end{aligned} \quad (2.396)$$

The first equation expresses the signal $x(t)$ as a linear combination of a set of uncountable basis functions $\phi_f(t)$ (sometimes also expressed as $\phi(t, f)$). The second equation, also called an *integral transform* of $x(t)$, gives the coefficient function $c(f)$ of the linear combination as the projection of $x(t)$ onto the basis function $\phi_f(t)$, also called the *kernel function* of the transform.

For a discrete signal $\mathbf{x} = [\dots, x[n], \dots]^T$, we have

$$\begin{aligned} \mathbf{x} &= \sum_k c[n]\mathbf{b}_k, \quad \text{or} \quad x[m] = \sum_k c[n]b[m, n], \quad (\text{for all } m) \\ c[n] &= \langle \mathbf{x}, \mathbf{b}_n \rangle = \sum_m x[m]\overline{b}[m, n], \quad (\text{for all } n) \end{aligned} \quad (2.397)$$

where $x[m]$ is the m th element of \mathbf{x} , and $b[m, n]$ is the m th element of the n th basis vector \mathbf{b}_k . The first equation expresses the signal vector as a linear combination of a set of countable basis vectors \mathbf{b}_k (or in component form

$b[m, n]$) for all n ; the second equation gives the n th coefficient $c[n]$ as the projection of the signal \mathbf{x} onto the corresponding basis vector \mathbf{b}_k .

Both of the two pairs of equations above are unitary (orthogonal if real) transformations. In either case, the second equation is the forward transform that converts the time signal given under the implicit standard basis to a continuous coefficient function or a set of discrete coefficients for a new basis; while the first equation is the inverse transform that represents the signal as a linear combination of the new basis weighted by the coefficients.

- The representations of the signal under different bases are equivalent, in the sense that the total amount of energy or information contained in the signal, represented by its norm of the vector, is conserved. This is because any two orthogonal bases are always related by a unitary transformation (a rotation), which conserves vector norms according to Parseval's equality.
- In the rest of the book we will study various orthogonal transforms each representing a given signal vector as the weighting coefficients or function of the corresponding basis used. The topics of interest in the future discussion include: why such a unitary transformation is desirable to start with; why it could represent a given signal in such a way that it can be most effectively and conveniently processed, analyzed, compressed for transmission and storage, and the information of interest extracted; and how to find the optimal transformation according to certain quantifiable criteria.
- In addition to the orthogonal transformations based on orthogonal basis vector or functions, each of which carries some independent information of the signal, we will also consider certain non-orthogonal basis functions, or even non-independent vectors. Specifically the frames discussed previously will be used in wavelet transforms. In such cases, the vectors used for representing the signal may be correlated and there may exist certain redundancy in terms of the signal information they each carry. There are both pros and cons in such signal representations with redundancy.

2.7 Homework Problems

1. Approximate a given 3-D vector $\mathbf{x} = [1, 2, 3]^T$ in an 2-D subspace spanned by the two standard basis vectors $\mathbf{e}_1 = [1, 0, 0]^T$ and $\mathbf{e}_2 = [0, 1, 0]^T$. Obtain the error vector $\tilde{\mathbf{x}}$ and verify that it is orthogonal to both \mathbf{e}_1 and \mathbf{e}_2 .
2. Repeat the problem above but now approximate the same 3-D vector $\mathbf{x} = [1, 2, 3]^T$ above but now in a different 2-D subspace spanned by two basis vectors $\mathbf{a}_1 = [1, 0, -1]^T$ and $\mathbf{a}_2 = [-1, 2, 0]^T$. Find a vector in this 2-D subspace $\tilde{\mathbf{x}} = c[1]\mathbf{a}_1 + c[2]\mathbf{a}_2$ so that the error $\|\mathbf{x} - \tilde{\mathbf{x}}\|$ is minimized.
3. Given two vectors $\mathbf{u}_1 = [2, 1]^T / \sqrt{5}$ and $\mathbf{u}_2 = [-1, 2]^T / \sqrt{5}$ in \mathbb{R}^2 , do the following:
 - a. Verify that they are orthogonal;
 - b. Normalized them;

- c. Use them as an orthonormal basis to represent a vector $\mathbf{x} = [1, 2]^T$.
4. Use the Gram-Schmidt orthogonalization process to construct two new orthonormal basis vectors \mathbf{b}_1 and \mathbf{b}_2 from the two vectors \mathbf{a}_1 and \mathbf{a}_2 used in the previous problem, so that they span the same 2-D space, and then approximate the vector $\mathbf{x} = [1, 2, 3]^T$ above. Note that as the off-diagonal elements of the 2 by 2 matrix are zero, and both elements on the main diagonal are one, the coefficients $c[1]$ and $c[2]$ can be easily found without solving a linear equation system.
5. Approximate a function $x(t) = t^2$ defined over an interval $[0, 1]$ in a 2-D space spanned by two basis functions $a_1(t)$ and $a_1(t)$:

$$a_1(t) = 1, \quad a_2(t) = \begin{cases} 0 & (0 \leq t < 1/2) \\ 1 & (1/2 \leq t < 1) \end{cases} \quad (2.398)$$

6. Repeat the problem above with the same \mathbf{a}_1 but a different \mathbf{a}_2 defined as:

$$a_2(t) = \begin{cases} -1 & (0 \leq t < 1/2) \\ 1 & (1/2 \leq t < 1) \end{cases} \quad (2.399)$$

Note that \mathbf{a}_1 and \mathbf{a}_2 are orthogonal $\langle a_1(t), a_2(t) \rangle = 0$ (they are actually the first two basis function of an orthogonal Walsh-Hadamard transform to be discussed in details later).

7. Repeat the problem above, but now with an additional basis function \mathbf{a}_3 defined as:

$$a_3(t) = \begin{cases} 1 & (0 \leq t < 1/4) \\ -1 & (1/4 \leq t < 3/4) \\ 1 & (3/4 \leq t < 1) \end{cases} \quad (2.400)$$

so that the 2-D space is expanded to a 3-D space spanned by $a_1(t)$, $a_2(t)$ and $a_3(t)$ (they are actually the first three basis functions of the Walsh-Hadamard transform).

8. Approximate the same function $x(t) = t^2$ above in a 3-D space spanned by three basis functions $a_0(t) = 1$, $a_1(t) = \sqrt{2} \cos(\pi t)$, and $a_2(t) = \sqrt{2} \cos(2\pi t)$, defined over the same time period. These happen to be the first three basis functions of the cosine transform.

Hint: The following integral may be needed:

$$\int x^2 \cos(ax) dx = \frac{2x \cos(ax)}{a^2} + \frac{a^2 x^2 - 2}{a^3} \sin(ax) + C \quad (2.401)$$

9. Consider a 2-D space spanned by two orthonormal basis vectors:

$$\mathbf{a}_1 = \frac{1}{2} \begin{bmatrix} \sqrt{3} \\ 1 \end{bmatrix}, \quad \mathbf{a}_2 = \frac{1}{2} \begin{bmatrix} -1 \\ \sqrt{3} \end{bmatrix} \quad (2.402)$$

- a. Represent vector $\mathbf{x} = [1, 2]^T$ under this basis as $\mathbf{x} = c[1]\mathbf{a}_1 + c[2]\mathbf{a}_2$. Find $c[1]$ and $c[2]$.
- b. Represent a counter clockwise rotation of $\theta = 30^\circ$ by a 2 by 2 matrix \mathbf{R} .

- c. Rotate vector \mathbf{x} to get $\mathbf{y} = \mathbf{Rx}$.
 - d. Represent \mathbf{y} above under basis $\{\mathbf{a}_1, \mathbf{a}_2\}$ by $\mathbf{y} = d[1]\mathbf{a}_1 + d[2]\mathbf{a}_2$. Find the two coefficients $d[1]$ and $d[2]$.
 - e. Rotate the basis $\{\mathbf{a}_1, \mathbf{a}_2\}$ in the opposite direction $-\theta = -30^\circ$ represented by $\mathbf{R}^{-1} = \mathbf{R}^T$ to get $\mathbf{b}_1 = \mathbf{Ra}_1$ and $\mathbf{b}_2 = \mathbf{Ra}_2$.
 - f. Represent \mathbf{x} under this new basis $\{\mathbf{b}_1, \mathbf{b}_2\}$ (which happens to be the standard basis).
 - g. Verify that $d'[1] = d[1]$ and $d'[2] = d[2]$, in other words, the representation $\{d[1], d[2]\}$ of the rotated vector \mathbf{y} under the original basis $\{\mathbf{a}_1, \mathbf{a}_2\}$ is equivalent to the representation $\{d'[1], d'[2]\}$ of the original vector \mathbf{x} under the inversely rotated basis $\{\mathbf{b}_1, \mathbf{b}_2\}$.
10. In Example 2.8 we approximated the temperature signal, a 8-D vector $\mathbf{x} = [65, 60, 65, 70, 75, 80, 75, 70]^T$, in a 3-D subspace spanned by three orthogonal basis vectors. This process can be continued by increasing the dimensionality from 3 to 8, so that the approximation error will be progressively reduced to reach zero, when eventually the signal vector is represented in the entire 8-D vector space. Consider the 8 orthogonal basis vectors shown below as the row vectors in this matrix (Walsh-Hadamard transform matrix):

$$\mathbf{H}_w = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix} \quad (2.403)$$

Note that the first three rows are used in the example. Now approximate the same signal by using 1 to all 8 rows as the basis vectors. Plot the original signal and the approximation in k-D subspaces for $k = 1, 2, \dots, 8$, adding one dimension at a time for more detailed variations in the signal. Find the coefficients $c[k]$ and the error in each case. Consider using some software tool such as Matlab.

11. The same temperature signal in Example 2.8 $\mathbf{x} = [65, 60, 65, 70, 75, 80, 75, 70]^T$ can also be approximated using a set of different basis vectors obtained by sampling the following cosine functions:

$$a_0(t) = 1, \quad a_1(t) = \sqrt{2} \cos(\pi t), \quad a_2(t) = \sqrt{2} \cos(2\pi t) \quad (2.404)$$

at 8 equally points $n_k = 1/16 + n/8 = 0.0625 + n \times 0.125$, ($n = 1, 2, \dots, 8$). The resulting vectors are actually used in the discrete cosine transform to be discussed later. Find the coefficients $c[k]$ and error for each approximation in a k-D subspace ($k = 1, 2, \dots, 8$), and plot the original signal together with the approximation for each case. Use a software tool such as Matlab.

12. Consider a frame in \mathbb{R}^2 containing three vectors that form a frame matrix:

$$\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3] = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \quad (2.405)$$

- Find the eigenvalues of $\mathbf{F}\mathbf{F}^T$ and its inverse $(\mathbf{F}\mathbf{F}^T)^{-1}$.
- Find the dual frame $\tilde{\mathbf{F}} = [\tilde{\mathbf{f}}_1, \tilde{\mathbf{f}}_2, \tilde{\mathbf{f}}_3]$.
- Find the coefficient vectors $\mathbf{c} = [c[1], c[2], c[3]]$ and $\mathbf{d} = [d[1], d[2], d[3]]$ for representing $\mathbf{x} = [1, 2]^T$ so that

$$\mathbf{x} = \sum_k c[k] \tilde{\mathbf{f}}_k = \sum_k d[k] \mathbf{f}_k \quad (2.406)$$

Verify that \mathbf{x} can be indeed perfectly reconstructed.

- Verify Eqs.2.304 and 2.305

13. Consider a frame in \mathbb{R}^2 containing two vectors that form a frame matrix:

$$\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2] = \begin{bmatrix} 2 & -1 \\ 1 & -2 \end{bmatrix} \quad (2.407)$$

As \mathbf{f}_1 and \mathbf{f}_2 are linearly independent, they form a Riesz basis.

- Find the dual frame and verify they are biorthonormal as shown in Eq.2.324.
- Given $\mathbf{x} = [2, 3]^T$, find the coefficient vectors \mathbf{c} and \mathbf{d}

$$\mathbf{x} = \sum_k c[k] \tilde{\mathbf{f}}_k = \sum_k d[k] \mathbf{f}_k \quad (2.408)$$

Verify that \mathbf{x} can be indeed perfectly reconstructed.

- Verify Eqs.2.304 and 2.305.

14. Given a basis in \mathbb{R}^3 :

$$\mathbf{f}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{f}_2 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \quad \mathbf{f}_3 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad (2.409)$$

Find its biorthogonal dual $\tilde{\mathbf{f}}_1, \tilde{\mathbf{f}}_2, \tilde{\mathbf{f}}_3$, and two sets of coefficients $c[k]$ and $d[k]$ ($k = 1, 2, 3$) to represent a vector $\mathbf{x} = [1, 2, 3]^T$.

3 Continuous-Time Fourier Transform

3.1 The Fourier Series Expansion of Periodic Signals

3.1.1 Formulation of The Fourier Expansion

As considered in the previous chapter, the second-order differential operator D^2 over the interval $[0, T]$ is a self-adjoint operator, and its eigenfunctions $\phi_k(t) = e^{j2k\pi t/T}/\sqrt{T}$ ($k = 0, \pm 1, \pm 2, \dots$) are orthonormal (Eq.2.183 i.e. Eq.2.130):

$$\langle \phi_k(t), \phi_l(t) \rangle = \frac{1}{T} \int_T e^{j2k\pi t/T} e^{-jln\pi t/T} dt = \frac{1}{T} \int_T e^{j2(f-l)\pi t/T} dt = \delta[k - l] \quad (3.1)$$

and they form a complete orthogonal system that spans a function space over interval $[0, T]$. Any periodic signal $x_T(t) = x_T(t + T)$ in the space can be expressed as a linear combination of these basis functions:

$$x_T(t) = \sum_{k=-\infty}^{\infty} X[k] \phi_k(t) = \frac{1}{\sqrt{T}} \sum_{k=-\infty}^{\infty} X[k] e^{j2k\pi t/T} \quad (3.2)$$

Here a periodic signal is denoted by $x_T(t)$ with a subscript T for its period. However, this subscript may be dropped for simplicity when no confusion will be caused. Note that at the two boundary points $t = 0$ and $t = T$ the summation on the right hand side is always equal to $\sum_{k=-\infty}^{\infty} X[k]/\sqrt{T}$, i.e., the condition in Eq. 2.167 is guaranteed. Consequently at these points $t = 0$ and $t = T$ the reconstructed signal in Eq.3.2 may not be the same as the original signal $x_T(t)$ if $x_T(0) \neq x_T(T)$.

Due to the orthogonality of these basis functions, the l th coefficient $X[l]$ can be found by taking an inner product with $\phi_l(t) = e^{j2l\pi t/T}/\sqrt{T}$ on both sides of the equation above:

$$\begin{aligned} \langle x_T(t), \phi_l(t) \rangle &= \langle x_T(t), e^{j2l\pi t/T}/\sqrt{T} \rangle = \frac{1}{T} \sum_{k=0}^{\infty} X[k] \langle e^{j2k\pi t/T}, e^{j2l\pi t/T} \rangle \\ &= \sum_{k=-\infty}^{\infty} X[k] \delta[k - n] = X[l] \end{aligned} \quad (3.3)$$

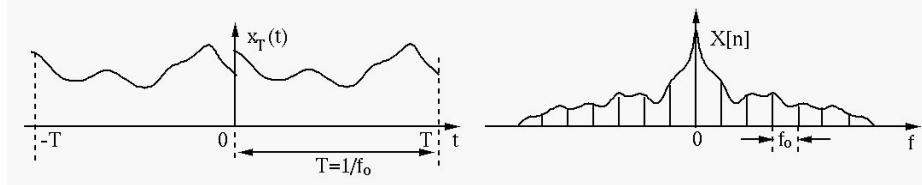


Figure 3.1 Fourier series expansion of periodic signals

i.e., the k th coefficient $X[k]$ is the projection of function $x_T(t)$ onto the k th basis function $\phi_k(t)$:

$$X[k] = \langle x_T(t), \phi_k(t) \rangle = \frac{1}{\sqrt{T}} \int_T x_T(t) e^{-j2k\pi t/T} dt \quad (3.4)$$

Equations 3.2 and 3.4 is the Fourier series expansion which can also be written as the following pair:

$$\begin{aligned} X[k] &= \mathcal{F}[x_T(t)] = \frac{1}{\sqrt{T}} \int_T x_T(t) e^{-j2k\pi t/T} dt = \langle x_T(t), e^{j2k\pi t/T} / \sqrt{T} \rangle, \\ &\quad (k = 0, \pm 1, \pm 2, \dots) \\ x_T(t) &= \mathcal{F}^{-1}[X[k]] = \frac{1}{\sqrt{T}} \sum_{k=-\infty}^{\infty} X[k] e^{j2k\pi t/T} \\ &= \sum_{k=-\infty}^{\infty} \langle x_T(t), e^{j2k\pi t/T} / \sqrt{T} \rangle e^{j2k\pi t/T} / \sqrt{T} \end{aligned} \quad (3.5)$$

As the signal and the basis functions are both periodic, the integral above can be over any interval of T , such as $[0, T]$ and $[-T/2, t < T/2]$.

As defined in Eq.2.175, we have $1/T = f_0$ and $2\pi/T = 2\pi f_0 = \omega_0$, so that the basis function can also be written as $\phi_k(t) = e^{j2k\pi f_0 t} / \sqrt{T} = e^{jk\omega_0 t} / \sqrt{T}$. We will use any of the equivalent expressions interchangeably, whichever most convenient in the specific discussion. Moreover, in practice, the constant scaling factor $1/\sqrt{T}$ in the equations above has little significance, therefore the Fourier series expansion pair could be expressed in some alternative forms such as:

$$\begin{aligned} x_T(t) &= \sum_{k=-\infty}^{\infty} X[k] e^{j2k\pi f_0 t} = \sum_{k=-\infty}^{\infty} X[k] e^{jk\omega_0 t} \\ X[k] &= \frac{1}{T} \int_T x_T(t) e^{-j2k\pi f_0 t} dt = \frac{1}{T} \int_T x_T(t) e^{-jk\omega_0 t} dt \end{aligned} \quad (3.6)$$

In this form, $X[0] = \int_T x_T(t) dt / T$ has a clear interpretation, it is the average, offset, or the DC (direct current) component of the signal.

The Fourier series expansion is a unitary transformation that converts a function $x_T(t)$ in the vector space of all periodic time functions into a vector $[\dots, X[-1], X[0], X[1], \dots]^T$ in another space of all vectors of infinite elements (or components). Also, the inner product of any two functions $x_T(t)$ and $y_T(t)$

remains the same before and after the unitary transformation:

$$\begin{aligned}
 < x_T(t), y_T(t) > &= \int_T x_T(t) \bar{y}_T(t) dt \\
 &= \frac{1}{T} \int_T \sum_{k=-\infty}^{\infty} X[k] e^{j2k\pi f_0 t} \sum_{l=-\infty}^{\infty} \bar{Y}[l] e^{-j2n\pi f_0 t} dt \\
 &= \frac{1}{T} \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} X[k] \bar{Y}[l] \int_T e^{j2(k-l)\pi f_0 t} dt \\
 &= \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} X[k] \bar{Y}[l] \delta[k - l] = \sum_{k=-\infty}^{\infty} X[k] \bar{Y}[k] = < \mathbf{X}, \mathbf{Y} >
 \end{aligned} \tag{3.7}$$

In particular, if $y_T(t) = x_T(t)$, the above becomes Parseval's identity

$$||x_T(t)||^2 = < x_T(t), x_T(t) > = < \mathbf{X}, \mathbf{X} > = ||\mathbf{X}||^2 \tag{3.8}$$

indicating that the total energy or information contained in the signal is conserved by the Fourier series expansion, therefore the signal can be equivalently represented in either time or frequency domain.

3.1.2 Physical Interpretation

The Fourier series expansion of a periodic signal $x_T(t)$ can also be expressed in terms of sine and cosine functions:

$$\begin{aligned}
 x_T(t) &= \sum_{k=-\infty}^{\infty} X[k] e^{jk\omega_0 t} = X[0] + \sum_{k=1}^{\infty} [X[-k] e^{-jk\omega_0 t} + X[k] e^{jk\omega_0 t}] \\
 &= X[0] + \sum_{k=1}^{\infty} [X[-k] (\cos k\omega_0 t - j \sin k\omega_0 t) + X[k] (\cos k\omega_0 t + j \sin k\omega_0 t)] \\
 &= X[0] + \sum_{k=1}^{\infty} [(X[k] + X[-k]) \cos k\omega_0 t + j(X[k] - X[-k]) \sin k\omega_0 t] \\
 &= X[0] + 2 \sum_{k=1}^{\infty} (a_k \cos k\omega_0 t + b_k \sin k\omega_0 t)
 \end{aligned} \tag{3.9}$$

Here we have defined $a_k = (X[k] + X[-k])/2$ and $b_k = (X[k] - X[-k])/2$, which can also be expressed as (Eq.3.6):

$$\begin{aligned}
 a_k &= \frac{1}{2T} \int_T x_T(t) [e^{-jk\omega_0 t} + e^{jk\omega_0 t}] dt = \frac{1}{T} \int_T x_T(t) \cos k\omega_0 t dt \\
 b_k &= \frac{j}{2T} \int_T x_T(t) [e^{-jk\omega_0 t} - e^{jk\omega_0 t}] dt = \frac{1}{T} \int_T x_T(t) \sin k\omega_0 t dt \\
 (k &= 1, 2, \dots)
 \end{aligned} \tag{3.10}$$

The two equations above are the alternative form of the Fourier series expansion of $x_T(t)$.

In particular, if $x_T(t)$ is real, we have

$$X[-k] = \frac{1}{T} \int_T x_T(t) e^{j2k\pi f_0 t} dt = \overline{X}[k] \quad (3.11)$$

which means

$$\operatorname{Re}[X[-k]] = \operatorname{Re}[X[k]], \quad \operatorname{Im}[X[-k]] = -\operatorname{Im}[X[k]] \quad (3.12)$$

i.e., the real part of $X[k]$ is even and the imaginary part is odd. Now we have:

$$\begin{aligned} a_k &= \frac{X[k] + X[-k]}{2} = \frac{X[k] + \overline{X}[k]}{2} = \operatorname{Re}[X[k]] \\ b_k &= \frac{j(X[k] - X[-k])}{2} = \frac{j(X[k] - \overline{X}[k])}{2} = -\operatorname{Im}[X[k]] \end{aligned} \quad (3.13)$$

i.e.,

$$\begin{cases} |X[k]| = \sqrt{a_k^2 + b_k^2} \\ \angle X[k] = -\tan^{-1}(b_k/a_k) \end{cases} \quad \begin{cases} a_k = |X[k]| \cos \angle X[k] \\ b_k = -|X[k]| \sin \angle X[k] \end{cases} \quad (3.14)$$

and the Fourier series expansion of a real signal $x_T(t)$ (Eq. 3.9) can be rewritten as:

$$\begin{aligned} x_T(t) &= X[0] + 2 \sum_{k=1}^{\infty} (a_k \cos k\omega_0 t + b_k \sin k\omega_0 t) \\ &= X[0] + 2 \sum_{k=1}^{\infty} |X[k]| (\cos \angle X[k] \cos k\omega_0 t - \sin \angle X[k] \sin k\omega_0 t) \\ &= X[0] + 2 \sum_{k=1}^{\infty} |X[k]| \cos(k\omega_0 t + \angle X[k]) \end{aligned} \quad (3.15)$$

This is yet another form of the Fourier expansion, which indicates that a real periodic signal $x_T(t)$ can be constructed as a superposition of infinite sinusoids of (a) different frequencies $k\omega_0$, (b) different amplitudes $|X[k]|$, and (c) different phases $\angle X[k]$. In particular, consider the following values for k :

- $k = 0$, the coefficient $X[0] = \int_T x_T(t) dt / T$ is the average or DC component of the signal $x_T(t)$;
- $k = 1$, the sinusoid $\cos(\omega_0 t + \angle X[1])$ has the same period T as the signal $x_T(t)$ and is therefore called the *fundamental frequency* of the signal;
- $k > 1$, the frequency of the sinusoidal function $\cos(k\omega_0 t + \angle X[k])$ is k times the frequency of the fundamental and is called the k th *harmonic* of the signal.

3.1.3 Properties of The Fourier Series Expansion

Here is a set of properties of the Fourier series expansion:

- **Linearity:**

$$\mathcal{F}[a x(t) + b y(t)] = a \mathcal{F}[x(t)] + b \mathcal{F}[y(t)] \quad (3.16)$$

As an integral operator which is by definition linear, the Fourier expansion is obviously linear.

- **Time scaling:** When $x_T(t)$ is scaled in time by a factor of $a > 0$ to become $x(at)$, its period becomes T/a and its fundamental frequency becomes $a/T = a\omega_0$. If $a > 1$, the signal is compressed by a factor a and the frequencies of its fundamental and harmonics become a times higher; if $a < 1$, the signal is expanded and the frequencies of its fundamental and harmonics are a times lower. In either case, the coefficients $X[k]$ remain the same:

$$x(at) = \sum_{k=-\infty}^{\infty} X[k] e^{j2ka\pi f_0 t} = \sum_{k=-\infty}^{\infty} X[k] e^{jka\omega_0 t} \quad (3.17)$$

- **Time shift:** A time signal $x(t)$ shifted in time by τ becomes $y(t) = x(t - \tau)$. Defining $t' = t - \tau$ we can get its Fourier coefficient as:

$$\begin{aligned} Y[k] &= \frac{1}{T} \int_T x(t - \tau) e^{-jk\omega_0 t} dt = \frac{1}{T} \int_T x(t') e^{-jk\omega_0 (t'+\tau)} dt' \\ &= X[k] e^{-jk\omega_0 \tau} = X[k] e^{-j2k\pi f\tau} \end{aligned} \quad (3.18)$$

- **Differentiation:** Fourier coefficients of the time derivative $y(t) = d x(t)/dt$ can be found to be:

$$\begin{aligned} Y[k] &= \frac{1}{T} \int_T \left[\frac{d}{dt} x(t) \right] e^{-jk\omega_0 t} dt \\ &= \frac{1}{T} \left[e^{-jk\omega_0 t} x(t) \Big|_0^T + jk\omega_0 \int_T x(t) e^{-jk\omega_0 t} dt \right] = jk\omega_0 X[k] = jk \frac{2\pi}{T} X[k] \end{aligned} \quad (3.19)$$

- **Integration:** The time integration of $x(t)$ is

$$y(t) = \int_{-\infty}^t x(\tau) d\tau \quad (3.20)$$

Note that $y(t)$ is periodic only if the DC component or average of $x(t)$ is zero, i.e., $X[0] = 0$ (otherwise it would accumulate over time by the integration to form a ramp). Since $x(t) = dy(t)/dt$, according to the differentiation property above, we have

$$X[k] = jk \frac{2\pi}{T} Y[k], \quad \text{i.e.} \quad Y[k] = \frac{T}{j2k\pi} X[k] \quad (3.21)$$

Note that $Y[0]$ can not be obtained from this formula as when $k = 0$, both the numerator and the denominator of $Y[k]$ are zero. However, as the DC component of $y(t)$, $Y[0]$ can be found by the definition:

$$Y[0] = \frac{1}{T} \int_T y(t) dt \quad (3.22)$$

- Parseval's theorem:

$$\frac{1}{T} \int_T |x_T(t)|^2 dt = \sum_{k=-\infty}^{\infty} |X[k]|^2 \quad (3.23)$$

This is already given in Eq.3.8. The left-hand side of the equation represents the average power in $x_T(t)$. The left-hand side can be written as

$$\frac{1}{T} \int_T |X[k]e^{j2\pi kf_0 t}|^2 dt = \frac{1}{T} \int_T |X[k]|^2 dt = |X[k]|^2 \quad (3.24)$$

which represents the average power contained in the k th frequency component.

Therefore Eq.3.23 states that the average power in one period of the signal is the sum of the average power of all of its frequency components, i.e., the power in the signal is conserved in either time or frequency domain.

3.1.4 The Fourier Expansion of Typical Functions

Here we consider the Fourier expansion of a set of typical periodic signals.

- Constant:

A constant $x(t) = 1$ can be expressed as a complex exponential $x(t) = e^{j0t}$ with arbitrary period T , i.e., it is a zero-frequency signal. The Fourier coefficient for this zero frequency is $X[0] = 1$, while all other coefficients for nonzero frequencies are zero. Alternatively, following the definition, we get (Eq.1.33):

$$X[k] = \frac{1}{T} \int_T e^{-jk\omega_0 t} dt = \delta[k] \quad (3.25)$$

- Complex exponential:

A complex exponential $x(t) = e^{j2\pi f_0 t} = e^{j\omega_0 t}$ (with period $T = 1/f_0 = 2\pi/\omega_0$) with a coefficient $X[1] = 1$. We can also find $X[k]$ by definition:

$$c_k = \frac{1}{T} \int_T e^{j\omega_0 t} e^{-jk\omega_0 t} dt = \frac{1}{T} \int_T e^{j\omega_0(1-k)t} dt = \delta[k-1] = \begin{cases} 1 & k=0 \\ 0 & k \neq 0 \end{cases} \quad (3.26)$$

- Sinusoids:

The cosine function $x(t) = \cos(2\pi f_0 t) = (e^{j2\pi f_0 t} + e^{-j2\pi f_0 t})/2$ of frequency f_0 is periodic with $T = 1/f_0$, and its Fourier coefficients are

$$\begin{aligned} X[k] &= \frac{1}{T} \int_T \cos(2\pi f_0 t) e^{-j2\pi kf_0 t} dt \\ &= \frac{1}{2} \left[\frac{1}{T} \int_T e^{-j2\pi(k-1)f_0 t} dt + \frac{1}{T} \int_T e^{-j2\pi(k+1)f_0 t} dt \right] \\ &= \frac{1}{2} (\delta[k-1] + \delta[k+1]) \end{aligned} \quad (3.27)$$

In particular, when $f_0 = 0$, $x(t) = 1$ and $X[k] = \delta[k]$, an impulse at zero, representing the constant (zero frequency) value. Similarly, the Fourier coefficient

of $x(t) = \sin(2\pi f_0 t)$ is:

$$\begin{aligned} X[k] &= \frac{1}{T} \int_T \sin(2\pi f_0 t) e^{-j2\pi k f_0 t} dt \\ &= \frac{1}{2j} \left[\frac{1}{T} \int_T e^{-j2\pi(k-1)f_0 t} dt - \frac{1}{T} \int_T e^{-j2\pi(k+1)f_0 t} dt \right] \\ &= \frac{1}{2j} (\delta[k-1] - \delta[k+1]) \end{aligned} \quad (3.28)$$

Alternatively, as

$$\cos(2\pi f_0 t) = \frac{1}{2} [e^{j2\pi f_0 t} + e^{-j2\pi f_0 t}] = \sum_{k=-\infty}^{\infty} X[k] e^{j2\pi f_0 t} \quad (3.29)$$

Comparing the two sides of the last equal sign we see that all $X[k] = 0$ except $X[1] = X[-1] = 1/2$, i.e., $X[k] = (\delta[k-1] + \delta[k+1])/2$. Similarly, comparing the two sides of the following

$$\sin(2\pi f_0 t) = \frac{1}{2j} [e^{j2\pi f_0 t} - e^{-j2\pi f_0 t}] = \sum_{k=-\infty}^{\infty} X[k] e^{j2\pi f_0 t} \quad (3.30)$$

we see that all $X[k] = 0$ except $X[1] = 1/2j$ and $X[-1] = -1/2j$, i.e., $X[k] = (\delta[k-1] - \delta[k+1])/2j$. This method can be used to find the Fourier coefficients of signals containing a small number of complex exponential terms.

- **Square wave:**

Let $x(t)$ be an odd square wave:

$$x(t) = \begin{cases} 1 & 0 < t < \tau \\ 0 & \tau < t < T \end{cases} \quad (3.31)$$

The Fourier coefficients of this function are

$$\begin{aligned} X[k] &= \frac{1}{T} \int_0^T x(t) e^{-j2k\pi f_0 t} dt = \frac{1}{T} \int_0^\tau e^{-j2k\pi f_0 t} dt = \frac{1}{j2k\pi} (1 - e^{-j2k\pi f_0 \tau}) \\ &= \frac{e^{-jk\pi f_0 \tau}}{k\pi} \frac{(e^{jk\pi f_0 \tau} - e^{-j2k\pi f_0 \tau})}{2j} = \frac{e^{-jk\pi f_0 \tau}}{k\pi} \sin(k\pi f_0 \tau) \end{aligned} \quad (3.32)$$

A *sinc function* is commonly defined as:

$$\text{sinc}(x) = \frac{\sin(\pi x)}{\pi x}, \quad \text{and} \quad \lim_{x \rightarrow 0} \text{sinc}(x) = 1 \quad (3.33)$$

and the expression above for $X[k]$ can be further written as:

$$X[k] = f_0 \tau \frac{\sin(k\pi f_0 \tau)}{k\pi f_0 \tau} e^{-jk\pi f_0 \tau} = \frac{\tau}{T} \text{sinc}(k f_0 \tau) e^{-jk\pi f_0 \tau} \quad (3.34)$$

In particular, the DC component is $X[0] = \tau/T$.

Also if $\tau = T/2$, then $X[0] = 1/2$ and $X[k]$ above becomes:

$$X[k] = \frac{1}{j2k\pi} (1 - e^{-jk\pi}) = \frac{e^{-jk\pi/2}}{k\pi} \sin(k\pi/2) \quad (3.35)$$

Moreover, since $e^{\pm j2k\pi} = 1$ and $e^{\pm j(2k-1)\pi} = -1$, all even terms $X[\pm 2k] = 0$ become zero and the odd terms become:

$$X[\pm(2k-1)] = \pm 1/j\pi(2k-1), \quad (k = 1, 2, \dots) \quad (3.36)$$

and the Fourier series expansion of the square wave becomes a linear combination of sinusoids:

$$\begin{aligned} x(t) &= \sum_{k=-\infty}^{\infty} X[k] e^{j2k\pi f_0 t} \\ &= X[0] + \sum_{k=1}^{\infty} \left[\frac{1}{j\pi(2k-1)} e^{j(2k-1)\omega_0 t} + \frac{1}{-j\pi(2k-1)} e^{-j(2k-1)\omega_0 t} \right] \\ &= \frac{1}{2} + \frac{2}{\pi} \sum_{k=1}^{\infty} \frac{\sin((2k-1)\omega_0 t)}{2k-1} \\ &= \frac{1}{2} + \frac{2}{\pi} \left[\frac{\sin(\omega_0 t)}{1} + \frac{\sin(3\omega_0 t)}{3} + \frac{\sin(5\omega_0 t)}{5} + \dots \right] \end{aligned} \quad (3.37)$$

As the function $x(t)$ is odd (except the DC), it is composed of only odd sine functions.

- **Triangle wave:** A triangle wave is defined as an even function:

$$x(t) = 2|t|/T, \quad (|t| \leq T/2) \quad (3.38)$$

First, the DC offset $X[0]$ can be found from the definition:

$$X[k] = \frac{1}{T} \int_T x(t) dt = \frac{T}{2} \quad (3.39)$$

For $k \neq 0$, we realize that this triangle wave can be obtained as an integral of the square wave defined in Eq. 3.31 with these modifications: (a) $\tau = T/2$, (b) DC offset is zero $X[0] = 0$, and (c) vertically scaled by $4/T$. Now according to the integration property, the Fourier coefficients can be easily obtained from Eq.3.35 as

$$X[k] = \frac{4}{T} \frac{T}{j2k\pi} \frac{e^{-jk\pi/2}}{k\pi} \sin(k\pi/2) = \frac{2 \sin(k\pi/2)}{j (k\pi)^2} e^{-jk\pi/2} = \frac{2 \sin(k\pi/2)}{(k\pi)^2} (-j)^{k+1} \quad (3.40)$$

This is a real and even $X[k] = X[-k]$ with respect to k ($k = 0, \pm 1, \pm 2, \dots$). According to the time shift property, the complex exponential $e^{jk\pi/2}$ corresponds to a right-shifted by $T/4$. If we shift the signal left by $T/4$, the triangle wave $x(t)$ becomes odd, the complex exponential term in the expression of $X[k]$ disappears:

$$X[k] = \frac{2 \sin(k\pi/2)}{j (k\pi)^2} \quad (3.41)$$

This is imaginary and odd $X[k] = -X[-k]$ with respect to k ($k = 0, \pm 1, \pm 2, \dots$).

The Fourier series expansion of such an odd triangle wave can be written as below. As the function $x(t)$ is odd (except the DC), it is composed of only odd sine functions.

$$\begin{aligned}
 x(t) &= \sum_{k=-\infty}^{\infty} X[k] e^{j2k\pi f_0 t} = \frac{1}{2} + \sum_{k=1}^{\infty} [X[k] e^{j2k\pi f_0 t} + X[-k] e^{-j2k\pi f_0 t}] \\
 &= \frac{1}{2} + \sum_{k=1}^{\infty} \left(\frac{2 \sin(k\pi/2)}{(k\pi)^2} e^{j2k\pi f_0 t} - \frac{2 \sin(k\pi/2)}{(k\pi)^2} e^{-j2k\pi f_0 t} \right) \\
 &= \frac{1}{2} + \frac{4}{\pi^2} \sum_{k=1}^{\infty} \frac{\sin(k\pi/2)}{k^2} \sin(2k\pi f_0 t) \\
 &= \frac{1}{2} + \frac{4}{\pi^2} [\sin(2\pi f_0 t) - \frac{1}{9} \sin(6\pi f_0 t) + \frac{1}{25} \sin(10\pi f_0 t) - \dots] \quad (3.42)
 \end{aligned}$$

- **Sawtooth:**

A sawtooth function is defined as

$$x(t) = t/T, \quad (0 < t < T) \quad (3.43)$$

First find $X[0]$, the average or DC component:

$$X[0] = \frac{1}{T} \int_T \frac{t}{T} e^{-j0\omega_0 t} dt = \frac{1}{2} \quad (3.44)$$

Next we find all remaining coefficients $X[k]$ ($k \neq 0$):

$$X[k] = \frac{1}{T} \int_T \frac{t}{T} e^{-jk\omega_0 t} dt \quad (3.45)$$

In general, this type of integrals can be found using integration by parts:

$$\int t e^{at} dt = \frac{1}{a^2} (at - 1) e^{at} + C \quad (3.46)$$

Here $a = -jk\omega_0 = -j2k\pi/T \neq 0$ and we get

$$X[k] = \frac{1}{T^2(jk\omega_0)^2} [(-jk\omega_0 t - 1) e^{-jk\omega_0 t}]_0^T = \frac{j}{2k\pi} \quad (3.47)$$

The Fourier series expansion of the function is

$$x(t) = \frac{1}{2} + \sum_{k=1}^{\infty} \left[\frac{j}{2k\pi} e^{j\omega_0 t} - \frac{j}{2k\pi} e^{-j\omega_0 t} \right] = \frac{1}{2} - \frac{1}{\pi} \sum_{k=1}^{\infty} \frac{1}{k} \sin(k\omega_0 t) \quad (3.48)$$

Note that this sawtooth wave is an odd function and therefore it is composed of only odd sine functions.

Some different versions of the square, triangle and sawtooth waveforms are shown in Fig.3.2. The corresponding Fourier series expansions of these waveforms are illustrated in Fig.3.3. The first ten basis functions for the DC component, fundamental frequency and progressively higher harmonics are shown on the left, and the reconstructions by inverse transform of the square, triangle and sawtooth waveforms are shown in the remaining three columns.

As we can see, the accuracy of the reconstruction of a waveform improves continuously as more basis functions of higher frequencies are included in the reconstruction so that finer details (corresponding to rapid changes in time) can be better represented.

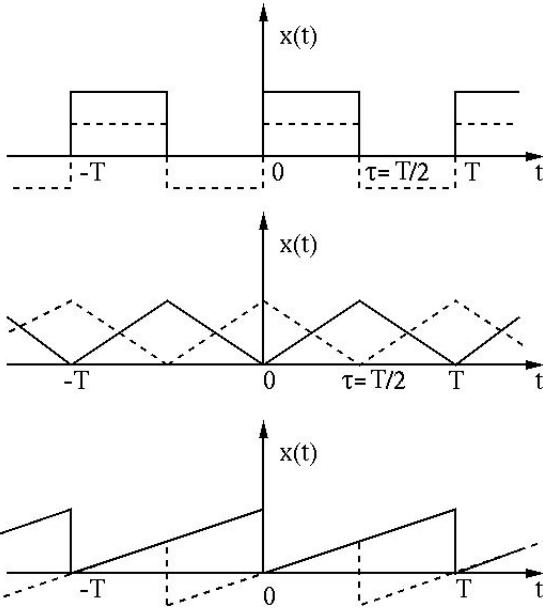


Figure 3.2 Square wave (top), triangle wave (middle) and sawtooth wave (bottom)

- **Impulse Train:**

An impulse train, also called a comb function or sampling function, is a sequence of infinite unit impulse separated by a time interval T :

$$\text{comb}(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT) \quad (3.49)$$

As a function with period T , this impulse train can be Fourier expanded:

$$\text{comb}(t) = \sum_{k=-\infty}^{\infty} \text{Comb}[k] e^{j2k\pi t/T} \quad (3.50)$$

with coefficients:

$$\begin{aligned} \text{Comb}[k] &= \frac{1}{T} \int_{-T/2}^{T/2} \text{comb}(t) e^{-j2k\pi t/T} dt = \frac{1}{T} \int_{-T/2}^{T/2} \sum_{n=-\infty}^{\infty} \delta(t - nT) e^{-j2k\pi t/T} dt \\ &= \frac{1}{T} \int_{-T/2}^{T/2} \delta(t) e^{-j2k\pi t/T} dt = \frac{1}{T}, \quad (k = 0, \pm 1, \pm 2, \dots) \end{aligned} \quad (3.51)$$

The last equation is due to Eq. 1.9. Substituting $\text{Comb}[k] = 1/T$ back into the Fourier series expansion of $\text{comb}(t)$, we can also express the impulse train

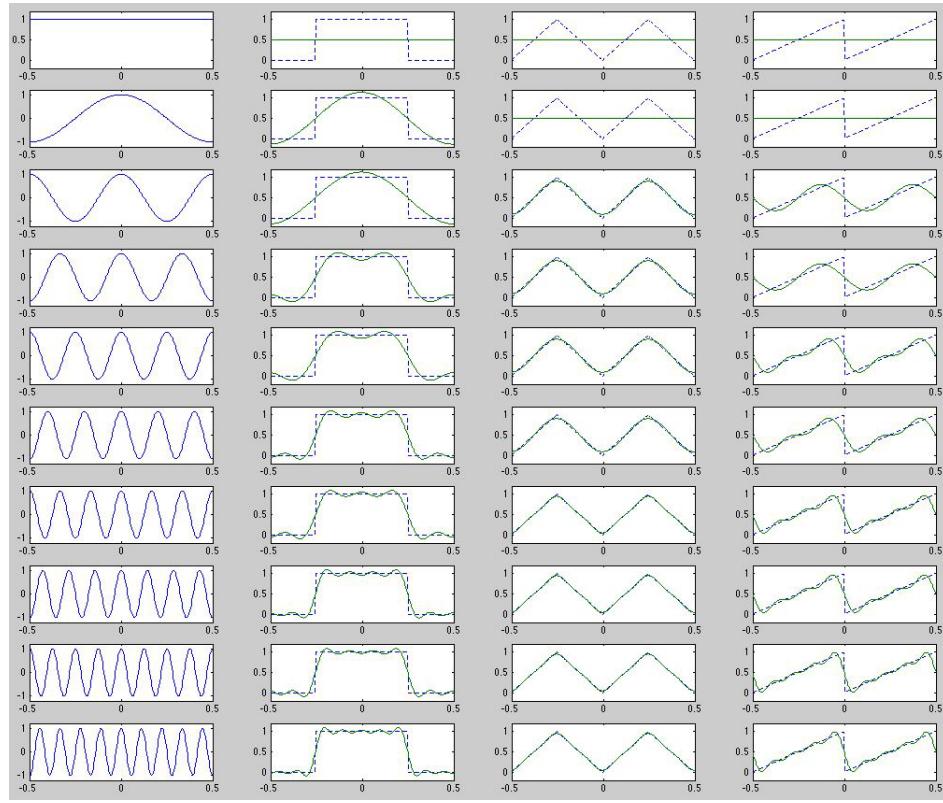


Figure 3.3 Fourier reconstructions of square, triangle, and sawtooth waveforms (2nd, 3rd and 4th columns) with progressively more higher harmonics included

as:

$$\text{comb}(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT) = \frac{1}{T} \sum_{k=-\infty}^{\infty} e^{j2k\pi t/T} \quad (3.52)$$

This is actually the same as Eq.1.35.

Fig.3.4 shows a set of periodic signals (left) and their corresponding Fourier coefficients (right).

To carry out the Fourier series expansion of a given signal function $x(t)$, it is necessary to first find its fundamental frequency f_0 or equivalently its period $T = 1/f_0$, which sometime is not explicitly available and therefore needs to be found.

Example 3.1: Consider a signal function $x(t) = \cos(2\pi 4t) + \cos(2\pi 6t)$ containing two sinusoids of frequencies $f_1 = 4$ and $f_2 = 6$ or periods $T_1 = 1/f_1 = 1/4$ and $T_2 = 1/f_2 = 1/6$, respectively. The fundamental frequency f_0 of the sum of these

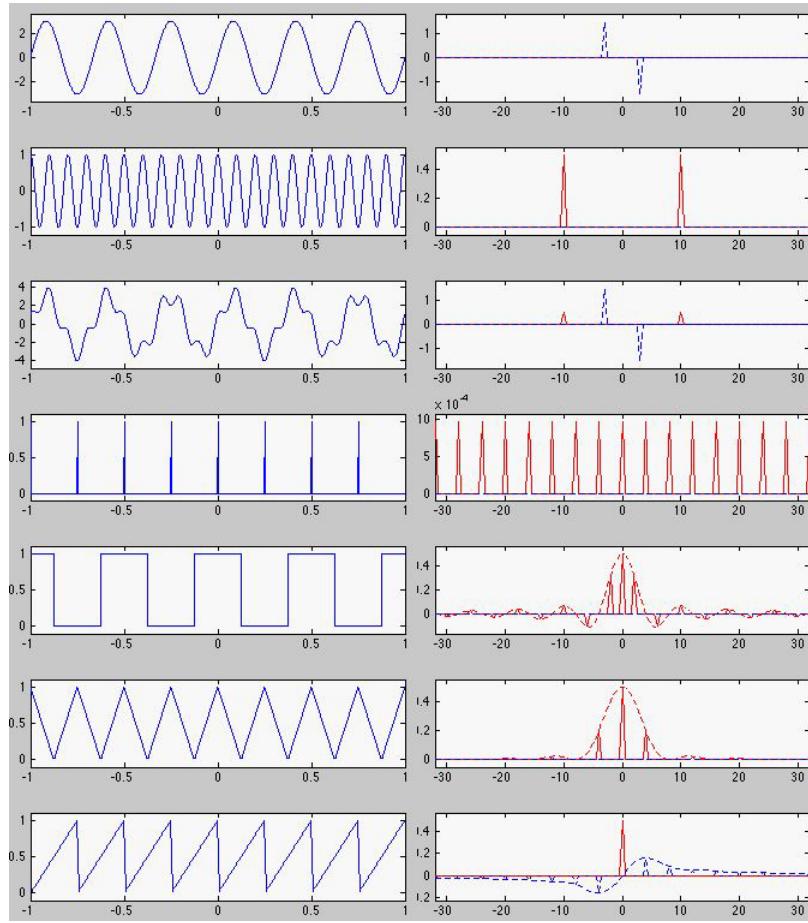


Figure 3.4 Examples of Fourier series expansions

A set of periodic signals (left) and their Fourier expansion coefficients (right) as a function of frequency f (real and imaginary parts are shown in solid and dashed lines, respectively). The first three rows show two sinusoids $x_1(t) = \sin(2\pi 3t)$ and $x_2(t) = \cos(2\pi 10t)$, and their weighted sum $x_1(t) + x_2(t)/5$. The following four rows are for the impulse train, square wave, triangle wave, and sawtooth wave, respectively.

component sinusoids can be found as the *greatest common divisor (GCD)* of the individual frequency components:

$$f_0 = GCD(f_1, f_2) = GCD(4, 6) = 2 \quad (3.53)$$

Or, equivalently, the period T of the sum can be found as the *least common multiple (LCM)* of the periods of the individual components:

$$T = LCM(T_1, T_2) = LCM(1/4, 1/6) = 1/2 \quad (3.54)$$

Now the signal can be expressed in terms of its fundamental frequency as $x(t) = \cos(2\pi 2f_0 t) + \cos(2\pi 3f_0 t)$ and its Fourier series coefficients can be found to be $X[k] = (\delta(k - 2) + \delta[k + 2] + \delta[k - 3] + \delta[k + 3])/2$.

3.2 The Fourier Transform of Non-Periodic Signals

3.2.1 Formulation

The Fourier series expansion does not apply to any non-periodic signal. To process and analyze such signals in frequency domain, the concept of the Fourier series expansion needs to be generalized. To do so, we first make some minor modification of the Fourier series expansion pair in Eq. 3.5 by moving the factor $1/T$ from the second equation to the first one:

$$\begin{aligned} x_T(t) &= \sum_{k=-\infty}^{\infty} \frac{1}{T} X[k] e^{jk\omega_0 t} = \sum_{k=-\infty}^{\infty} \frac{1}{T} X[k] e^{j2k\pi f_0 t} \\ X[k] &= \int_T x_T(t) e^{-jk\omega_0 t} dt = \int_T x_T(t) e^{-j2k\pi f_0 t} dt \end{aligned} \quad (3.55)$$

Here the coefficient $X[k]$ is redefined so that its value is scaled by T , and its dimension becomes that of the signal $x_T(t)$ multiplied by time, or divided by frequency (the exponential term $\exp(\pm j2\pi f_0 t)$ is dimensionless).

Next we convert a periodic signal $x_T(t)$ into a non-periodic signal $x(t)$ simply by increasing its period T to approach infinity $T \rightarrow \infty$. At the limit the following changes take place:

- The gap between two consecutive frequency components approaches zero $f_0 = 1/T \rightarrow 0$, and the discrete frequencies kf_0 for all integers $-\infty < k < \infty$ can be replaced by a continuous variable $-\infty < f < \infty$;
- The discrete and periodic basis functions $\phi_k(t) = e^{jk\omega_0 t}$ for all k become uncountable and non-periodic $\phi_f(t) = e^{j2\pi ft}$ for all f , as an orthogonal basis that spans the function space over $(-\infty, \infty)$ (Eq.1.28):

$$\langle \phi_f(t), \phi_{f'}(t) \rangle = \int_{-\infty}^{\infty} e^{j2\pi(f-f')t} dt = \delta(f - f') \quad (3.56)$$

- The coefficients $X[k]$ for the k th basis function, or the k th frequency component, $\phi_k(t) = e^{jk\omega_0 t}$ for all k is replaced by a continuous weight function $X(f)$ for the continuous and uncountable basis function $\phi_f(t) = e^{j2\pi ft}$ for all f ;
- Let $\Delta f = f_0 = 1/T$, then $1/T = \Delta f \rightarrow df$ when $T \rightarrow \infty$, and the summation in the first equation in Eq. 3.55 becomes an integral.

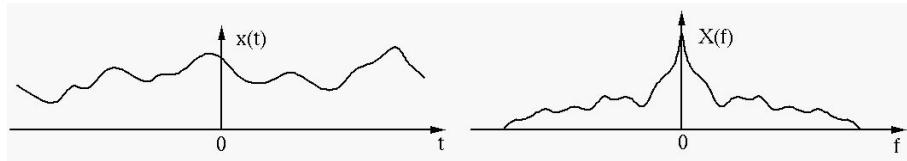


Figure 3.5 Fourier transform of non-periodic and continuous signals

When the time signal is no longer periodic, its discrete spectrum represented by the Fourier series coefficients becomes a continuous function.

Due to the changes above, when $T \rightarrow \infty$, the two equations in Eq. 3.55 become:

$$\begin{aligned} x(t) &= \lim_{T \rightarrow \infty} \left[\frac{1}{T} \sum_{k=-\infty}^{\infty} X[k] e^{j2k\pi f_0 t} \right] = \int_{-\infty}^{\infty} X(f) e^{j2\pi f t} df \\ X(f) &= \lim_{T \rightarrow \infty} \left[\int_T x(t) e^{-j2k\pi f_0 t} dt \right] = \int_{-\infty}^{\infty} x(t) e^{-j2\pi f t} dt \end{aligned} \quad (3.57)$$

These two equations can be rewritten as the *continuous-time Fourier transform (CTFT)* pair:

$$\begin{aligned} X(f) &= \mathcal{F}[x(t)] = \int_{-\infty}^{\infty} x(t) e^{-j2\pi f t} dt \\ x(t) &= \mathcal{F}^{-1}[X(f)] = \int_{-\infty}^{\infty} X(f) e^{j2\pi f t} df \end{aligned} \quad (3.58)$$

The first and second equations are the forward and inverse CTFT, respectively, which can be more concisely represented as:

$$x(t) \xleftrightarrow{\mathcal{F}} X(f) \quad (3.59)$$

The weighting function $X(f)$ in Eq.3.58 is called the *Fourier spectrum* of $x(t)$, representing how the signal energy is distributed over frequency, in comparison with $x(t)$ representing how the signal energy is distributed over time. A non-periodic signal and its continuous spectrum are illustrated in Fig.3.5, in comparison to a periodic signal and its discrete spectrum shown in Fig.3.1.

Eq.3.58 can be considered as the most generic form of the forward and inverse Fourier transform pair, generally denoted by $\mathcal{F}[\cdot]$ and $\mathcal{F}^{-1}[\cdot]$, with different variations depending on the specific nature of the signal $x(t)$, such as whether it is periodic or aperiodic, continuous or discrete (to be considered in the next chapter). For example, the Fourier series expansion in Eq.3.5 is just a special case of Eq.3.58, where the Fourier transform is applied to a periodic signal $x(t) = x_T(t)$, and the Fourier coefficients $X[k]$ are just the discrete spectrum $X(f) = \mathcal{F}[x_T(t)]$ of the periodic signal, as shown in the following subsection (Eq.3.78).

Comparing Eq.3.58 with Eqs.2.134 and 2.135, we see that the CTFT is actually the representation of a signal function $x(t)$ by an uncountably infinite set of orthonormal basis functions (Eq.2.133) defined as:

$$\phi_f(t) = e^{j2\pi f t}, \quad (-\infty < f < \infty) \quad (3.60)$$

so that the function $x(t)$ can be expressed as a linear combination, an integral, of these basis functions $\phi_f(t)$ over all frequencies f :

$$x(t) = \int_{-\infty}^{\infty} X(f)\phi_f(t)df = \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df \quad (3.61)$$

This is the second equation in Eq.3.58, and the coefficient function $X(f)$ can be found as the projection of the signal function $x(t)$ onto the basis function $\phi_f(t)$:

$$X(f) = \langle x(t), \phi_f(t) \rangle = \frac{1}{\sqrt{T}} \int_0^T x_T(t)e^{-j2\pi kt/T} dt \quad (3.62)$$

This is the forward CTFT in Eq.3.58.

The Fourier transform pair in Eq.3.58 can also be equivalently represented in terms of the angular frequency $\omega = 2\pi f$:

$$\begin{aligned} X(\omega) &= \mathcal{F}[x(t)] = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt \\ x(t) &= \mathcal{F}^{-1}[X(\omega)] = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega)e^{j\omega t}d\omega \end{aligned} \quad (3.63)$$

In some literatures, the CTFT spectrum $X(f)$ or $X(\omega)$ is also denoted by $X(j\omega)$, as it takes this form when treated as a special case of the *Laplace transform*, to be considered in Chapter 6. However, all these different forms are just some notational variations of the same spectrum, a function of frequency f or angular frequency $\omega = 2\pi f$. We will use these notations interchangeably, whichever most convenient and suitable in the specific discussion, as no confusion should be caused given the context. Moreover, we also note that when the spectrum is denoted by $X(f)$, the Fourier transform pair in Eq.3.58 appears symmetric between time and frequency domains so that the time-frequency duality is more clearly revealed.

In order for the integral in Eq.3.58 to converge, i.e., for $X(f)$ to exist, the signal $x(t)$ needs to satisfy the following Dirichlet conditions:

1. $x(t)$ is absolutely integrable:

$$\int_{-\infty}^{\infty} |x(t)|dt < \infty \quad (3.64)$$

2. $x(t)$ has finite number of maxima and minima within any finite interval;
3. $x(t)$ has finite number of discontinuities within any finite interval.

Alternatively, a more strict condition for the convergence of the integral is that $x(t)$ is an energy signal $x(t) \in L^2(\mathbb{R})$, i.e., it is square-integrable (Eq. 2.29). As some obvious examples, signals such as $x(t) = t$ and $x(t) = t^2$ grow without bound as $|t| \rightarrow \infty$ and therefore their Fourier spectra do not exist. However, we note that the Dirichlet conditions are sufficient but not necessary, as there also exist some signals that do not satisfy such conditions but their Fourier spectra may still exist. For example, some important and commonly used signals such $x(t) = 1$ and $x(t) = u(t)$ are neither square integrable nor absolutely integrable,

but their Fourier spectra can still be obtained, due to the introduction of the Dirac delta, a non-conventional function containing a value of infinity. The integrals of these functions can be considered to be marginally convergent.

Similar to the Fourier series expansion, the Fourier transform is also a unitary transformation $\mathcal{F}x(t) = X(f)$ that conserves inner product (Theorem 2.6):

$$\begin{aligned}
 < x(t), y(t) > &= \int_{-\infty}^{\infty} x(t)\bar{y}(t)dt \\
 &= \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df \right] \left[\int_{-\infty}^{\infty} \bar{Y}(f')e^{-j2\pi f't}df' \right] dt \\
 &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} X(f)\bar{Y}(f') \left[\int_{-\infty}^{\infty} e^{j2\pi(f-f')t}dt \right] df df' \\
 &= \int_{-\infty}^{\infty} X(f) \int_{-\infty}^{\infty} \bar{Y}(f')\delta(f-f')df' df \\
 &= \int_{-\infty}^{\infty} X(f)\bar{Y}(f)df = < X(f), Y(f) >
 \end{aligned} \tag{3.65}$$

Replacing $y(t)$ by $x(t)$ in Eq.3.65 above, we get Parseval's identity:

$$\|x(t)\|^2 = < x(t), x(t) > = < X(f), X(f) > = \|X(f)\|^2 \tag{3.66}$$

As a unitary transformation, the Fourier transform can be considered as a rotation of the basis functions of the function space. Before the transform, the function is represented as a linear combination of a uncountable set of standard basis functions $\delta(t - \tau)$ each for a particular time moment $t = \tau$, weighted by the coefficient function $x(\tau)$ for the signal amplitude at the time moment:

$$x(t) = \int_{-\infty}^{\infty} x(\tau)\delta(t - \tau)d\tau \tag{3.67}$$

After the transformation, the function is represented as a linear combination of a different set of orthonormal basis functions, a rotated version of the standard basis $\mathcal{F}^{-1}[\delta(t - \tau)] = e^{j2\pi f\tau}$, weighted by the spectrum $X(f)$ for each frequency component:

$$x(t) = \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df \tag{3.68}$$

The representations of the signal as a function $x(t)$ in time domain and a spectrum $X(f)$ in frequency domain are equivalent, in the sense that the total amount of energy or information is conserved due to the Parseval's identity. However, how the total energy is distributed through time t or frequency f can be very different, which may be one of the reasons why the Fourier transform is carried out to start with.

Example 3.2: Here we consider the Fourier transform of a few special signals:

- The unit impulse or Dirac delta:

$$\mathcal{F}[\delta(t)] = \int_{-\infty}^{\infty} \delta(t) e^{-j2\pi f t} dt = e^{-j2\pi 0 f} = 1 \quad (3.69)$$

- The constant function:

$$\mathcal{F}[1] = \int_{-\infty}^{\infty} e^{-j2\pi f t} dt = \delta(f) \quad (3.70)$$

The second equal sign is due to Eq.1.28.

- The unit step defined as:

$$u(t) = \begin{cases} 0 & t < 0 \\ 1/2 & t = 0 \\ 1 & t > 0 \end{cases} \quad (3.71)$$

Its Fourier transform is (Eq.1.30):

$$\mathcal{F}[u(t)] = \int_{-\infty}^{\infty} u(t) e^{-j2\pi f t} dt = \int_0^{\infty} e^{-j2\pi f t} dt = \frac{1}{2} \delta(f) + \frac{1}{j2\pi f} \quad (3.72)$$

Similarly, we also have (Eq.1.31):

$$\mathcal{F}[u(-t)] = \int_{-\infty}^0 e^{-j2\pi f t} dt = \frac{1}{2} \delta(f) - \frac{1}{j2\pi f} \quad (3.73)$$

Note that the term $\delta(f)/2$ is for the DC component of the unit step. These results can be verified based on the fact that $u(-t) = 1 - u(t)$:

$$\mathcal{F}[u(-t)] = \mathcal{F}[1] - \mathcal{F}[u(t)] = \delta(f) - \frac{1}{2} \delta(f) - \frac{1}{j2\pi f} = \frac{1}{2} \delta(f) - \frac{1}{j2\pi f} \quad (3.74)$$

- The sign function $x(t) = sgn(t)$ defined as:

$$sgn(t) = 2u(t) - 1 = \begin{cases} -1 & t < 0 \\ 0 & t = 0 \\ 1 & t > 0 \end{cases} \quad (3.75)$$

Due to linearity of the Fourier transform, its spectrum can be found to be:

$$\mathcal{F}[sgn(t)] = 2\mathcal{F}[u(t)] - \mathcal{F}[1] = \delta(f) + \frac{1}{j\pi f} - \delta(f) = \frac{1}{j\pi f} \quad (3.76)$$

Note that the term $\delta(f)/2$ disappears as the sign function has zero DC component.

3.2.2 Relation to The Fourier Expansion

Now let us consider how the Fourier spectrum of a periodic function is related to its Fourier expansion coefficients. The Fourier expansion of a periodic function

$x_T(t)$ is:

$$x_T(t) = \sum_{k=-\infty}^{\infty} X[k] e^{j2k\pi t/T} = \sum_{k=-\infty}^{\infty} X[k] e^{j2k\pi f_0 t} \quad (3.77)$$

where $f_0 = 1/T$ is the fundamental frequency and $X[k]$ the expansion coefficient. The Fourier transform of this periodic function $x_T(t)$ can be found to be:

$$\begin{aligned} X(f) &= \int_{-\infty}^{\infty} x_T(t) e^{-j2\pi f t} dt = \int_{-\infty}^{\infty} \left[\sum_{k=-\infty}^{\infty} X[k] e^{j2k\pi f_0 t} \right] e^{-j2\pi f t} dt \\ &= \sum_{k=-\infty}^{\infty} X[k] \int_{-\infty}^{\infty} e^{-j2\pi(f-kf_0)t} dt = \sum_{k=-\infty}^{\infty} X[k] \delta(f - kf_0) \end{aligned} \quad (3.78)$$

Here we have used the result of Eq.1.29. It is clear that the spectrum of a periodic function is discrete, in the sense that it is non-zero only at a set of discrete frequencies $f = kf_0$ where $X(f) = X[k]\delta(f - kf_0)$. This result also illustrates an important point: while the dimension of the Fourier coefficient $X[k]$ is the same as that of the signal $x_T(t)$, i.e., $[X[k]] = [x_T(t)]$, the dimension of the spectrum is

$$[X(f)] = [X[k]][t] = \frac{[X[k]]}{[f]} \quad (3.79)$$

As the dimension of $X(f)$ is that of the signal $x(t)$ multiplied by time, or divided by frequency, $X(f)$ is actually a *frequency density* function.

In the future we will loosely use the term “spectrum” for not only a continuous function $X(f)$ of frequency f , but also the discrete transform coefficients $X[k]$ as they can always be associated with a continuous function as in Eq.3.78

Next, we consider how the Fourier spectrum $X(t)$ of a signal $x(t)$ can be related to the Fourier series coefficients of its periodic extension defined as:

$$x'(t) = \sum_{n=-\infty}^{\infty} x(t + nT) = x'(t + T) \quad (3.80)$$

As $x'(t + T) = x'(t)$ is periodic, it can be Fourier expanded and the k th Fourier coefficient is:

$$\begin{aligned} X'[k] &= \frac{1}{T} \int_0^T x'(t) e^{-j2\pi kt/T} dt = \frac{1}{T} \int_0^T \left[\sum_{n=-\infty}^{\infty} x(t + nT) \right] e^{-j2\pi kt/T} dt \\ &= \frac{1}{T} \sum_{n=-\infty}^{\infty} \int_0^T x(t + nT) e^{-j2\pi kt/T} dt \end{aligned} \quad (3.81)$$

If we define $\tau = t + nT$, i.e., $t = \tau - nT$, the above becomes:

$$\begin{aligned} X'[k] &= \frac{1}{T} \sum_{n=-\infty}^{\infty} \int_{nT}^{(n+1)T} x(\tau) e^{-j2\pi k\tau/T} d\tau e^{-j2\pi nk} \\ &= \frac{1}{T} \int_{-\infty}^{\infty} x(\tau) e^{-j2\pi k\tau/T} d\tau = \frac{1}{T} X\left(\frac{k}{T}\right) \end{aligned} \quad (3.82)$$

(Note that $e^{-j2\pi nk} = 1$ as k and n are both integer.) This equation relates the Fourier transform $X(f)$ of a signal $x(t)$ to the Fourier series coefficient $X'[k]$ of the periodic extension $x'(t)$ of the signal. Now the Fourier expansion of $x'(t)$ can be written as:

$$x'(t) = \sum_{k=-\infty}^{\infty} X'[k] e^{j2\pi kt/T} = \frac{1}{T} \sum_{k=-\infty}^{\infty} X\left(\frac{k}{T}\right) e^{j2\pi kt/T} \quad (3.83)$$

This equation is actually a special case of the *Poisson summation formula* given in Eq.3.163 when $X(f) = \mathcal{F}[\delta(t)] = 1$.

3.2.3 Properties of The Fourier Transform

Here we consider a set of properties of the Fourier transform, many of which should look similar to those of the Fourier series expansion discussed before, simply because the Fourier expansion is just a special case (for periodic signals) of the Fourier transform, it naturally shares all of the properties. In the following, we always assume $x(t)$ and $y(t)$ are two complex functions (real as a special case) and $\mathcal{F}[x(t)] = X(f)$ and $\mathcal{F}[y(t)] = Y(f)$.

- **Linearity:**

$$\mathcal{F}[ax(t) + by(t)] = a\mathcal{F}[x(t)] + b\mathcal{F}[y(t)] \quad (3.84)$$

The Fourier transform of a function $x(t)$ is simply an inner product of the function with a kernel function $\phi_f(t) = e^{j2\pi ft}$ (Eq.3.62). Due to the linearity of the inner product in the first variable, the Fourier transform is also linear.

- **Time-frequency duality:**

$$\text{if } \mathcal{F}[x(t)] = X(f), \quad \text{then} \quad \mathcal{F}[X(t)] = x(-f) \quad (3.85)$$

Proof:

$$x(t) = \mathcal{F}^{-1}[X(f)] = \int_{-\infty}^{\infty} X(f) e^{j2\pi ft} df \quad (3.86)$$

Defining $t' = -t$, we have

$$x(-t') = \int_{-\infty}^{\infty} X(f) e^{-j2\pi ft'} df \quad (3.87)$$

Interchanging variables t' and f , we get

$$x(-f) = \int_{-\infty}^{\infty} X(t') e^{-j2\pi ft'} dt' = \mathcal{F}[X(t)] \quad (3.88)$$

In particular, if $x(t) = x(-t)$ is even, we have

$$\text{if } \mathcal{F}[x(t)] = X(f), \quad \text{then} \quad \mathcal{F}[X(t)] = x(f) \quad (3.89)$$

This duality is simply the result of the definition of the forward and inverse transforms in Eq. 3.58, which are highly symmetric between time and fre-

quency. Consequently, many of the properties and transforms of typical functions have strong duality between the time and frequency domains.

- **Even and odd signals:**

- If the signal is even, then its spectrum is also even:

$$\text{if } x(t) = x(-t), \quad \text{then} \quad X(f) = X(-f) \quad (3.90)$$

proof:

$$\begin{aligned} X(f) &= \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft} dt = \int_{-\infty}^{\infty} x(-t)e^{-j2\pi ft} dt \\ &= \int_{-\infty}^{\infty} x(t')e^{j2\pi ft'} dt' = X(-f) \end{aligned} \quad (3.91)$$

where we have assumed $t' = -t$.

- If the signal is odd, then its spectrum is also odd:

$$\text{if } x(t) = -x(-t), \quad \text{then} \quad X(f) = -X(-f) \quad (3.92)$$

The proof is similar to the above.

- **Time reversal:**

$$\mathcal{F}[x(-t)] = X(-f) \quad (3.93)$$

i.e., if the signal $x(t)$ is flipped in time with respect to the origin $t = 0$, its spectrum $X(f)$ is also flipped in frequency with respect to the origin $f = 0$,

Proof:

$$\mathcal{F}[x(-t)] = \int_{-\infty}^{\infty} x(-t)e^{-j2\pi ft} dt = \int_{-\infty}^{\infty} x(t')e^{j2\pi ft'} dt' = X(-f) \quad (3.94)$$

where we have assumed $-t' = t$. In particular, when $x(t) = \bar{x}(t)$ is real,

$$\mathcal{F}[x(-t)] = X(-f) = \int_{-\infty}^{\infty} x(t)e^{j2\pi ft} = \overline{\int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}} = \overline{X(f)} \quad (3.95)$$

- **Multiplication (Plancherel) theorem:**

$$\langle x(t), y(t) \rangle = \int_{-\infty}^{\infty} x(t)\bar{y}(t) dt = \int_{-\infty}^{\infty} X(f)\bar{Y}(f) df = \langle X(f), Y(f) \rangle \quad (3.96)$$

This is Eq. 3.65, indicating that the Fourier transform is a unitary transformation that conserves inner product. In particular, letting $y(t) = x(t)$, we get Parseval's identity representing signal energy conservation by the Fourier transform:

$$\|x(t)\|^2 = \int_{-\infty}^{\infty} |x(t)|^2 dt = \int_{-\infty}^{\infty} |X(f)|^2 df = \int_{-\infty}^{\infty} S_x(f) df = \|X(f)\|^2 \quad (3.97)$$

where $|x(t)|^2$ and $S_x(f) = |X(f)|^2$ are respectively the signal energy distributions over time and frequency, and $S_x(f)$ is defined as the *power density spectrum (PDS)* of the signal.

- Time and frequency scaling:

$$\mathcal{F}[x(at)] = \frac{1}{|a|} X\left(\frac{f}{a}\right) \quad (3.98)$$

Proof: First we assume a positive scaling factor $a > 0$ and get:

$$\mathcal{F}[x(at)] = \int_{-\infty}^{\infty} x(at)e^{-j2\pi ft} dt = \int_{-\infty}^{\infty} x(u)e^{-j2\pi fu/a} d\left(\frac{u}{a}\right) = \frac{1}{a} X\left(\frac{f}{a}\right) \quad (3.99)$$

where we have assumed $u = at$. Applying the time-reversal property to this result we get:

$$\mathcal{F}[x(-at)] = \frac{1}{a} X\left(-\frac{f}{a}\right) \quad (3.100)$$

Letting $a' = -a < 0$, we get the following for a negative scaling factor:

$$\mathcal{F}[x(a't)] = \frac{1}{-a'} X\left(\frac{f}{a'}\right) \quad (3.101)$$

Combining the above results for both positive and negative scaling factors, we get Eq.3.98.

If $|a| < 1$, the signal is stretched and its spectrum is compressed and scaled up. When $|a| \rightarrow 0$, $x(at)$ is so stretched that it approaches a constant, and its spectrum is compressed and scaled up to the extent that it approaches an impulse. On the other hand, if $|a| > 1$, then the signal is compressed and its spectrum is stretched and scaled down. When $|a| \rightarrow \infty$, we redefine the signal as $a x(at)$ with spectrum $X(f/a)$, the signal becomes an impulse and its spectrum $X(f/a)$ becomes a constant.

- Time and frequency shift:

$$\mathcal{F}[x(t \pm t_0)] = e^{\pm j2\pi f t_0} X(f) \quad (3.102)$$

$$\mathcal{F}^{-1}[X(f \pm f_0)] = e^{\mp j2\pi f_0 t} x(t) \quad (3.103)$$

Proof: We first prove Eq.3.102:

$$\mathcal{F}[x(t \pm t_0)] = \int_{-\infty}^{\infty} x(t \pm t_0) e^{-j2\pi f t} dt \quad (3.104)$$

Let $t' = t \pm t_0$, then $t = t' \mp t_0$, $dt' = dt$, the above becomes

$$\mathcal{F}[x(t \pm t_0)] = \int_{-\infty}^{\infty} x(t') e^{-j2\pi f(t' \mp t_0)} dt' = e^{\pm j2\pi f t_0} X(f) \quad (3.105)$$

We see that a time shift t_0 of the signal corresponds to a phase shift $2\pi f t_0$ for every frequency component $e^{j2\pi f t}$. This result can be intuitively understood. As the phase shift is proportional to the frequency, a higher frequency component will have a greater phase shift while a lower frequency component will have a smaller phase shift, so that the relative positions of all harmonics remain the same, and the shape of the signal as a superposition of these harmonics remains the same when shifted.

Applying the time-frequency duality to the time shift property in Eq.3.102, we get the frequency shift property in Eq.3.103.

- **Correlation:**

The *cross-correlation* between two functions $x(t)$ and $y(t)$ is defined as

$$r_{xy}(t) = x(t) \star y(t) = \int_{-\infty}^{\infty} x(\tau) \bar{y}(\tau - t) d\tau \quad (3.106)$$

Its Fourier transform is:

$$\mathcal{F}[r_{xy}(t)] = \mathcal{F}[x(t) \star y(t)] = X(f) \bar{Y}(f) \quad (3.107)$$

Proof:

As $\mathcal{F}[x(\tau)] = X(f)$ and $\mathcal{F}[y(\tau - t)] = Y(f)e^{-j2\pi ft}$, we can easily prove the property by applying the multiplication theorem:

$$\begin{aligned} r_{xy}(t) &= \int_{-\infty}^{\infty} x(\tau) \bar{y}(\tau - t) d\tau = \int_{-\infty}^{\infty} X(f) \bar{Y}(f) e^{j2\pi ft} df \\ &= \int_{-\infty}^{\infty} S_{xy}(f) e^{j2\pi f t} df = \mathcal{F}^{-1}[S_{xy}(f)] \end{aligned} \quad (3.108)$$

where $S_{xy}(f)$ is the *cross power density spectrum* $S_{xy}(f)$ of the two signals defined as:

$$S_{xy}(f) = X(f) \bar{Y}(f) = \mathcal{F}[r_{xy}(t)] \quad (3.109)$$

If both signals $\bar{x}(t) = x(t)$ and $\bar{y}(t) = y(t)$ are real, i.e., $\bar{X}(f) = X(-f)$ and $\bar{Y}(f) = Y(-f)$, then we have $S_{xy}(f) = X(f)Y(-f)$. In particular, when $x(t) = y(t)$, we have:

$$r_x(t) = \int_{-\infty}^{\infty} x(\tau) \bar{x}(\tau - t) d\tau = \int_{-\infty}^{\infty} S_x e^{j2\pi f \tau} df = \mathcal{F}^{-1}[S_x(f)] \quad (3.110)$$

where $r_x(t)$ is the *auto-correlation* and $S_x(f) = X(f) \bar{X}(f) = |X(f)|^2$ is the *power density spectrum* of $x(t)$.

- **Convolution theorem:**

As first defined by Eq.1.86 in Chapter 1, the convolution of two functions $x(t)$ and $y(t)$ is:

$$z(t) = x(t) * y(t) = \int_{-\infty}^{\infty} x(\tau) y(t - \tau) d\tau = \int_{-\infty}^{\infty} y(\tau) x(t - \tau) d\tau = y(t) * x(t) \quad (3.111)$$

If $y(t) = y(-t)$ is even, then $x(t) * y(t) = x(t) \star y(t)$ is the same as the correlation. The convolution theorem states:

$$\mathcal{F}[x(t) * y(t)] = X(f) Y(f) \quad (3.112)$$

$$\mathcal{F}[x(t)y(t)] = X(f) * Y(f) \quad (3.113)$$

Proof:

$$\begin{aligned}
 \mathcal{F}[x(t) * y(t)] &= \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} x(\tau)y(t - \tau)d\tau \right] e^{-j2\pi f t} dt \\
 &= \int_{-\infty}^{\infty} x(\tau)e^{-j2\pi f \tau} \int_{-\infty}^{\infty} y(t - \tau)e^{-j2\pi f(t-\tau)} dt d\tau \\
 &= \int_{-\infty}^{\infty} x(\tau)e^{-j2\pi f \tau} Y(f) d\tau = X(f)Y(f)
 \end{aligned} \tag{3.114}$$

Similarly, we can also prove:

$$\mathcal{F}[x(t)y(t)] = X(f) * Y(f) \tag{3.115}$$

In particular, as shown in Eq.1.85 in Chapter 1, the output $y(t)$ of an LTI system can be found as the convolution $y(t) = h(t) * x(t)$ of its impulse response $h(t)$ and the input $x(t)$. Now according to the convolution theorem, the output of the system can be more conveniently obtained in frequency domain by a multiplication:

$$Y(f) = H(f)X(f) \tag{3.116}$$

where $X(f)$ and $Y(f)$ are respectively the spectra of the input $x(t)$ and the output $y(t)$, and $H(f) = \mathcal{F}[h(t)]$, the Fourier transform of the impulse response function $h(t)$, is the *frequency response function (FRF)* of the system, first defined by Eq.1.91 in Chapter 1.

- **Time derivative:**

$$\mathcal{F} \left[\frac{d}{dt}x(t) \right] = j2\pi f X(f) = j\omega X(\omega) \tag{3.117}$$

Proof:

$$\begin{aligned}
 \frac{d}{dt}x(t) &= \frac{d}{dt} \int_{-\infty}^{\infty} X(f)e^{j2\pi f t} df = \int_{-\infty}^{\infty} X(f) \frac{d}{dt}e^{j2\pi f t} df \\
 &= \int_{-\infty}^{\infty} j2\pi f X(f)e^{j2\pi f t} df = \mathcal{F}^{-1}[j2\pi f X(f)]
 \end{aligned} \tag{3.118}$$

Repeating this process we get:

$$\mathcal{F} \left[\frac{d^n}{dt^n}x(t) \right] = (j2\pi f)^n X(f) \tag{3.119}$$

- **Frequency derivative:**

$$\begin{aligned}
 \mathcal{F}[t x(t)] &= j \frac{d}{df} X(f) \\
 \mathcal{F}[t^n x(t)] &= j^n \frac{1}{(2\pi)^n} \frac{d^n}{df^n} X(f)
 \end{aligned} \tag{3.120}$$

The proof is very similar to the above.

- **Time integration:**

The Fourier transform of a time integration is:

$$\mathcal{F}\left[\int_{-\infty}^t x(\tau)d\tau\right] = \frac{1}{j2\pi f} X(f) + \frac{1}{2}X(0)\delta(f) \quad (3.121)$$

Proof:

The integral of a signal $x(t)$ can be considered as its convolution with $u(t)$:

$$x(t) * u(t) = \int_{-\infty}^{\infty} x(\tau)u(t - \tau)d\tau = \int_{-\infty}^t x(\tau)d\tau \quad (3.122)$$

Due to the convolution theorem, we have:

$$\begin{aligned} \mathcal{F}\left[\int_{-\infty}^t x(\tau)d\tau\right] &= \mathcal{F}[x(t) * u(t)] = X(f)\left[\frac{1}{j2\pi f} + \frac{1}{2}\delta(f)\right] \\ &= \frac{1}{j2\pi f} X(f) + \frac{X(0)}{2}\delta(f) \end{aligned} \quad (3.123)$$

Comparing Eqs.3.117 and 3.121, we see that the time derivative and integral are the inverse operations of each other in frequency domain as well as in time domain. However, the second term in Eq.3.121 is necessary for representing the DC component $X(0)$ in signal $x(t)$, while Eq.3.117 does not have a corresponding term as derivative operation is insensitive to DC component in the signal.

- **Complex conjugate:**

$$\mathcal{F}[\bar{x}(t)] = \overline{X}(-f) \quad (3.124)$$

Proof: Taking the complex conjugate of the inverse Fourier transform, we get:

$$\begin{aligned} \bar{x}(t) &= \overline{\int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df} = \int_{-\infty}^{\infty} \overline{X(f)}e^{-j2\pi ft}df \\ &= \int_{-\infty}^{\infty} \overline{X}(-f')e^{j2\pi f't}df' = \mathcal{F}^{-1}[\overline{X}(-f)] \end{aligned} \quad (3.125)$$

$$(3.126)$$

where we have defined $f' = -f$.

- **Real and imaginary signals:**

- If $x(t)$ is real, then the real part $X_r(f)$ of its spectrum is even and the imaginary part $X_j(f)$ is odd.

$$X_r(f) = X_r(-f), \quad \text{and} \quad X_j(f) = -X_j(-f) \quad (3.127)$$

Proof: As $\bar{x}(t) = x(t)$ is real, i.e., $\mathcal{F}[\bar{x}(t)] = \mathcal{F}[x(t)]$, from Eq.3.124 we get:

$$X(f) = \overline{X}(-f), \quad \text{i.e.,} \quad X_r(f) + jX_j(f) = X_r(-f) - jX_j(-f) \quad (3.128)$$

Equating the real and imaginary parts on both sides we get Eq.3.127.

Moreover, when the real signal is either even or odd, we have the following results based on Eqs.3.90 and 3.92:

Table 3.1. Symmetry Properties of Fourier Transform

$x(t) = x_r(t) + jx_i(t)$	$X(f) = X_r(f) + jX_j(f)$
$x(t) = x_r(t)$ real	$X_r(f) = X_r(-f)$ even, $X_j(f) = -X_j(-f)$ odd
$x_r(t) = x_r(-t)$ real, even	$X_r(f) = X_r(-f)$ real, even $X_j(f) = 0$
$x_r(t) = -x_r(-t)$ real, odd	$X_j(f) = -X_j(f)$ imaginary, odd $X_r(f) = 0$
$x(t) = x_j(t)$ imaginary	$X_r(f) = -X_r(-f)$ odd, $X_j(f) = X_j(-f)$ even
$x_j(t) = x_j(-t)$ imaginary, even	$X_j(f) = X_j(-f)$ imaginary, even $X_r(f) = 0$
$x_j(t) = -x_j(-t)$ imaginary, odd	$X_r(f) = -X_r(-f)$ real, odd $X_j(f) = 0$

- * If $x(t) = x(-t)$ is even, then $X(f)$ is also even, i.e. $X_j(f) = 0$ and $X(f) = X_r(f) = X_r(-f)$ is real and even.
- * If $x(t) = -x(-t)$ is odd, then $X(f)$ is also odd, i.e., $X_r(f) = 0$ and $X(f) = X_j(f) = -X_j(-f)$ is imaginary and odd.
- If $x(t)$ is imaginary, then the real part $X_r(f)$ of its spectrum is odd and the imaginary part $X_j(f)$ is even:

$$X_r(f) = -X_r(-f), \quad \text{and} \quad X_j(f) = X_j(-f) \quad (3.129)$$

Proof: As $\bar{x}(t) = -x(t)$ is imaginary, i.e., $\mathcal{F}[\bar{x}(t)] = -\mathcal{F}[x(t)]$, from Eq.3.124 we get:

$$-X(f) = \overline{X}(-f), \quad \text{i.e.,} \quad X_r(f) + jX_j(f) = -X_r(-f) + jX_j(-f) \quad (3.130)$$

Equating the real and imaginary parts on both sides we get Eq.3.129.

Moreover, when the imaginary signal is either even or odd, we have the following results based on Eqs.3.90 and 3.92:

- * If $x(t) = x(-t)$ is even, then $X(f)$ is also even, i.e., $X_r(f) = 0$ and $X(f) = jX_j(f) = jX_j(-f)$ is imaginary and even.
- * If $x(t) = -x(-t)$ is odd, then $X(f)$ is also odd, i.e., $X_j(f) = 0$ and $X(f) = X_r(f) = -X_r(-f)$ is real and odd.

These results are summarized in Table 3.1.

The complex spectrum $X(f)$ of a time signal $x(t)$ can be expressed in either Cartesian form in terms of the real and imaginary parts $X_r(f)$ and $X_j(f)$, or in polar form in terms of the magnitude $|X(f)|$ and phase $\angle X(f)$:

$$X(f) = X_r(f) + jX_j(f) = |X(f)|e^{j\angle X(f)} \quad (3.131)$$

where

$$\begin{cases} |X(f)| = \sqrt{X_r^2(f) + X_j^2(f)} \\ \angle X(f) = \tan^{-1}[X_j(f)/X_r(f)] \end{cases}, \quad \begin{cases} X_r(f) = |X(f)| \cos \angle X(f) \\ X_j(f) = |X(f)| \sin \angle X(f) \end{cases} \quad (3.132)$$

We see that when the signal is either real or imaginary, $|X(f)|$ is always even and $\angle X(f)$ is always odd.

- **Physical interpretation:**

The spectrum of a signal $x(t)$ can be expressed as:

$$\begin{aligned} x(t) &= \int_{-\infty}^{\infty} X(f) e^{j2\pi f t} df = \int_{-\infty}^{\infty} |X(f)| e^{j2\pi f t + \angle X(f)} df \\ &= \int_{-\infty}^{\infty} |X(f)| \cos(2\pi f t + \angle X(f)) df + j \int_{-\infty}^{\infty} |X(f)| \sin(2\pi f t + \angle X(f)) df \end{aligned} \quad (3.133)$$

If $x(t)$ is real (as most signals in practice), the second term is zero while the first term (an integral of an even function of f) remains, and we have:

$$x(t) = 2 \int_0^{\infty} |X(f)| \cos(2\pi f t + \angle X(f)) df \quad (3.134)$$

We see that the Fourier transform expresses a real time signal as a superposition of infinitely many uncountable frequency components each with a different frequency f , magnitude $|X(f)|$, and phase $\angle X(f)$. Note that Eq.3.15 for periodical signals is just the discrete version of the equation above.

3.2.4 Fourier Spectra of Typical Functions

- **Unit impulse:**

The Fourier transform of the unit impulse function is given in Eq.3.69 according to the definition of the Fourier transform:

$$\mathcal{F}[\delta(t)] = \int_{-\infty}^{\infty} \delta(t) e^{-j2\pi f t} dt = 1 \quad (3.135)$$

- **Sign function:**

The Fourier transform of the sign function $sgn(t)$ is given in Eq.3.76:

$$\mathcal{F}[sgn(t)] = \frac{1}{j\pi f} \quad (3.136)$$

Note that $sign(t)$ is real and odd, and its spectrum is imaginary and odd. Moreover, based on the time-frequency duality property, we also get:

$$\mathcal{F}\left[\frac{1}{t}\right] = -j\pi sgn(f) \quad (3.137)$$

- **Unit step functions:**

As the unit step is the time integral of the unit impulse:

$$u(t) = \int_{-\infty}^t \delta(t)dt \quad (3.138)$$

and $\mathcal{F}[\delta(t)] = 1$, $\mathcal{F}[u(t)]$ can be found according the time integration property (Eq.3.121) to be:

$$\mathcal{F}[u(t)] = \frac{1}{j2\pi f} + \frac{1}{2}\delta(f) \quad (3.139)$$

which is the same as in Eq.3.72.

Moreover, due to the time reversal property $\mathcal{F}[x(-t)] = X(-f)$, we can also get the Fourier transform of a left-sided unit step:

$$\mathcal{F}[u(-t)] = \frac{1}{2}\delta(-f) + \frac{1}{-j2\pi f} = \frac{1}{2}\delta(f) - \frac{1}{j2\pi f} \quad (3.140)$$

(as $\delta(-f) = \delta(f)$.)

- **Constant:**

As a constant time function $x(t) = 1$ is not square-integrable, the integral of its Fourier transform does not converge:

$$\mathcal{F}[1] = \int_{-\infty}^{\infty} e^{-j2\pi ft} dt \quad (3.141)$$

However, we realize that the constant time function is simply the sum of a right-sided unit step and a left-sided unit step: $x(t) = 1 = u(t) + u(-t)$, and according to the linearity of the Fourier transform we have:

$$\mathcal{F}[1] = \mathcal{F}[u(t)] + \mathcal{F}[u(-t)] = \frac{1}{j2\pi f} + \frac{1}{2}\delta(f) - \frac{1}{j2\pi f} + \frac{1}{2}\delta(f) = \delta(f) \quad (3.142)$$

Alternatively, the Fourier transform of constant 1 can also be obtained according to the property of time-frequency duality, based on the Fourier transform of the unit impulse:

$$\mathcal{F}[1] = \int_{-\infty}^{\infty} e^{-j2\pi ft} dt = \delta(f) \quad (3.143)$$

Due to the property of time-frequency scaling, if the time function $x(t)$ is scaled by a factor of $1/2\pi$ to become $x(t/2\pi)$, its spectrum $X(f)$ will become $2\pi X(2\pi f) = 2\pi X(\omega)$. Specifically in this case, if we scale the constant 1 as a time function by $1/2\pi$ (still the same), its spectrum $X(f) = \delta(f)$ can be expressed as a function of angular frequency $X(\omega) = 2\pi\delta(\omega)$.

- **Complex exponentials and sinusoids:**

The Fourier transform of a complex exponential $x(t) = e^{j\omega_0 t} = e^{j2\pi f_0 t}$ of frequency f_0 is:

$$\mathcal{F}[e^{j2\pi f_0 t}] = \int_{-\infty}^{\infty} e^{-j2\pi(f-f_0)t} dt = \delta(f - f_0) \quad (3.144)$$

and according to Euler's formula, the Fourier transform of cosine function $x(t) = \cos(2\pi f_0 t)$ is:

$$\mathcal{F}[\cos(2\pi f_0 t)] = \frac{1}{2}[\delta(f - f_0) + \delta(f + f_0)] \quad (3.145)$$

and similarly the Fourier transform of $x(t) = \sin(2\pi f_0 t)$ is:

$$\mathcal{F}[\sin(2\pi f_0 t)] = \frac{1}{2j}[\delta(f - f_0) - \delta(f + f_0)] \quad (3.146)$$

Note that the sine and cosine functions are respectively odd and even, and so are their Fourier spectra. Also note that none of the step, constant, complex exponential and sinusoidal functions considered above is square-integrable, and correspondingly their Fourier transform integrals are only marginally convergent, in the sense that their transform functions $X(f)$ all contain a delta function ($\delta(f)$, $\delta(f - f_0)$, etc.) with an infinite value at certain frequency.

- **Exponential functions:**

A right-sided exponential decay function is defined as $e^{-at}u(t)$ ($a > 0$), and its Fourier transform can be found to be:

$$\begin{aligned} \mathcal{F}[e^{-at}u(t)] &= \int_0^\infty e^{-at}e^{-j2\pi ft}dt = \frac{1}{-(a + j2\pi f)}e^{-(a+j2\pi f)t}\Big|_0^\infty \\ &= \frac{1}{a + j2\pi f} = \frac{1}{a + j\omega} = \frac{a - j\omega}{a^2 + \omega^2} \end{aligned} \quad (3.147)$$

As $\lim_{a \rightarrow 0} e^{-at}u(t) \rightarrow u(t)$, we have

$$\mathcal{F}[u(t)] = \lim_{a \rightarrow 0} \mathcal{F}[e^{-at}u(t)] = \lim_{a \rightarrow 0} \left[\frac{1}{a + j2\pi f} \right] = \frac{1}{2}\delta(f) + \frac{1}{j2\pi f} \quad (3.148)$$

which is the same as in Eq.3.72. Note that it is tempting to assume at the limit $a = 0$, the second term alone will result, while in fact the first term $\delta(f)/2$ is also necessary. The proof of this result is left to the reader as a homework problem.

Next consider a left-sided exponential decay function $e^{at}u(-t)$, the time-reversal version of the right-sided decay function. According time reversal property $\mathcal{F}[x(-t)] = X(-f)$, we get:

$$\mathcal{F}[e^{at}u(-t)] = \frac{1}{a - j2\pi f} = \frac{1}{a - j\omega} \quad (3.149)$$

Finally, a two-sided exponential decay $e^{-a|t|}$ is the sum of the right-sided and left-sided decay functions and according to the linearity property, its Fourier transform can be obtained as:

$$\begin{aligned} \mathcal{F}[e^{-a|t|}] &= \mathcal{F}[e^{-at}u(t)] + \mathcal{F}[e^{at}u(-t)] = \frac{1}{a + j2\pi f} + \frac{1}{a - j2\pi f} \\ &= \frac{2a}{a^2 + (2\pi f)^2} = \frac{2a}{a^2 + \omega^2} \end{aligned} \quad (3.150)$$

- **Rectangular function and sinc function:**

A rectangular function, also called a square impulse, of width τ is defined as

$$\text{rect}_\tau(t) = \begin{cases} 1 & 0 < |t| < \tau/2 \\ 0 & \text{else} \end{cases} \quad (3.151)$$

which can be considered as the difference between two unit step functions:

$$\text{rect}(t) = u(t + \tau/2) - u(t - \tau/2) \quad (3.152)$$

Due to the properties of linearity and time shift, the spectrum of $\text{rect}_\tau(t)$ can be found to be

$$\begin{aligned} \mathcal{F}[\text{rect}(t)] &= \mathcal{F}[u(t + \tau/2)] - \mathcal{F}[u(t - \tau/2)] = \frac{e^{j\pi f \tau}}{j2\pi f} - \frac{e^{-j\pi f \tau}}{j2\pi f} \\ &= \frac{\tau}{\pi f \tau} \sin(\pi f \tau) = \tau \text{sinc}(f\tau) \end{aligned} \quad (3.153)$$

This spectrum is zero at $f = k/\tau$ for any integer k . If we let the width $\tau \rightarrow \infty$, the rectangular function becomes a constant 1 and its spectrum an impulse function. If we divide both sides of the equation above by τ and let $\tau \rightarrow 0$, the time function becomes an impulse and its spectrum a constant.

As both the rectangular function and sinc function are symmetric, the time-frequency duality property applies, i.e., the Fourier spectrum of a sinc function in time domain is a rectangular function in frequency domain, called an ideal low-pass filter:

$$H_{lp}(f) = \begin{cases} 1 & |f| < f_c \\ 0 & |f| > f_c \end{cases} \quad (3.154)$$

where f_c is called the *cutoff frequency*, then according to time-frequency duality, its time impulse response is:

$$h_{lp}(t) = \frac{\sin(2\pi f_c t)}{\pi t} = 2f_c \text{sinc}(2f_c t) \quad (3.155)$$

Note that the impulse response $h_{lp}(t)$ is nonzero for $t < 0$, indicating that the ideal low-pass filter is not causal (response before the input $\delta(0)$ at $t = 0$). In other words, an ideal low-pass filter is impossible to implement in real-time, but it can be trivially realized off-line in frequency domain.

- **Triangle function:**

$$\text{triangle}(t) = \begin{cases} 1 - |t|/\tau & |t| < \tau \\ 0 & |t| \geq \tau \end{cases} \quad (3.156)$$

Following the definition, the spectrum of the triangle function, as an even function, can be obtained as:

$$\begin{aligned}
 \mathcal{F}[triangle(t)] &= 2 \int_0^\tau (1 - t/\tau) \cos(2\pi f t) dt \\
 &= 2 \left[\int_0^\tau \cos(2\pi f t) dt - \frac{1}{\tau} \int_0^\tau t \cos(2\pi f t) dt \right] \\
 &= \frac{1}{\pi f} \left[\sin(2\pi f \tau) - \frac{t}{\tau} \sin(2\pi f t) \Big|_0^\tau + \frac{1}{\tau} \int_0^\tau \sin(2\pi f t) dt \right] \\
 &= \frac{-1}{2\tau(\pi f)^2} \cos(2\pi f t) \Big|_0^\tau = \frac{1}{2\tau(\pi f)^2} (1 - \cos(2\pi f \tau)) \\
 &= \tau \frac{\sin^2(\pi f \tau)}{(\pi f \tau)^2} = \tau \operatorname{sinc}^2(f \tau)
 \end{aligned} \tag{3.157}$$

Alternatively, the triangle function (with width 2τ) can be obtained more easily as the convolution of two square functions (with width τ) scaled by $1/\tau$:

$$triangle(t) = \frac{1}{\tau} rect(t) * rect(t) \tag{3.158}$$

its Fourier transform can be conveniently obtained based on the convolution theorem:

$$\mathcal{F}[triangle(t)] = \frac{1}{\tau} \mathcal{F}[rect(t) * rect(t)] = \frac{1}{\tau} \tau \operatorname{sinc}(f) \tau \operatorname{sinc}(f) = \tau \operatorname{sinc}^2(f \tau) \tag{3.159}$$

- **Gaussian function:**

Consider the Gaussian function $x(t) = e^{-\pi(t/a)^2}/a$. Note that in particular when $a = \sqrt{2\pi\sigma^2}$, $x(t)$ becomes the normal distribution with variance σ^2 and mean $\mu = 0$. The spectrum of $x(t)$ is:

$$\begin{aligned}
 X(f) &= \mathcal{F}\left[\frac{1}{a} e^{-\pi(t/a)^2}\right] = \frac{1}{a} \int_{-\infty}^{\infty} e^{-\pi(t/a)^2} e^{-j2\pi f t} dt \\
 &= \frac{1}{a} \int_{-\infty}^{\infty} e^{-\pi((t/a)^2 + j2ft)} dt = \frac{1}{a} e^{\pi(jaf)^2} \int_{-\infty}^{\infty} e^{-\pi[(t/a)^2 + j2ft + (jaf)^2]} dt \\
 &= e^{-\pi(af)^2} \int_{-\infty}^{\infty} e^{-\pi(t/a + jaf)^2} dt = e^{-\pi(af)^2}
 \end{aligned} \tag{3.160}$$

The last equation is due to the identity $\int_{-\infty}^{\infty} e^{-\pi x^2} dx = 1$. We see that the Fourier transform of a Gaussian function is another Gaussian function, and the area underneath either $x(t)$ or $X(f)$ is unity. Moreover, If we let $a \rightarrow 0$, $x(t)$ will approach $\delta(t)$, while its spectrum $e^{-\pi(af)^2}$ approaches 1. On the other hand, if we rewrite the above as

$$X(f) = \mathcal{F}[x(t)] = \mathcal{F}[e^{-\pi(t/a)^2}] = ae^{-\pi(af)^2} \tag{3.161}$$

and let $a \rightarrow \infty$, $x(t)$ approaches 1 and $X(f)$ approaches $\delta(f)$.

- **Impulse train:**

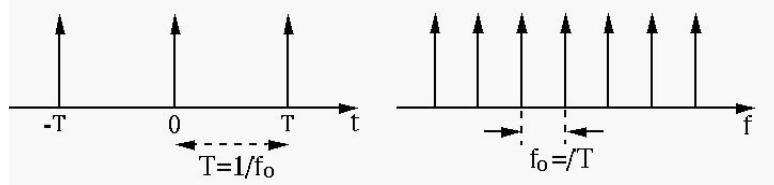


Figure 3.6 Impulse train and its spectrum

As discussed before the impulse train is a sequence of infinite unit impulses separated by a constant time interval T :

$$\text{comb}(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT) \quad (3.162)$$

The Fourier transform of this function is:

$$\begin{aligned} \mathcal{F}[\text{comb}(t)] &= \int_{-\infty}^{\infty} \text{comb}(t) e^{-j2\pi ft} dt = \int_{-\infty}^{\infty} \left[\sum_{n=-\infty}^{\infty} \delta(t - nT) \right] e^{-j2\pi ft} dt \\ &= \sum_{n=-\infty}^{\infty} \int_{-\infty}^{\infty} \delta(t - nT) e^{-j2\pi ft} dt = \sum_{n=-\infty}^{\infty} e^{-j2\pi nfT} \\ &= f_0 \sum_{n=-\infty}^{\infty} \delta(f - nf_0) = \frac{1}{T} \sum_{n=-\infty}^{\infty} \delta(f - n/T) \end{aligned} \quad (3.163)$$

where we have used Eq.1.35 with F replaced by f_0 . This equation, also called *Poisson formula*, is very useful in the discussion of impulse trains.

- **Periodic signals:**

As discussed before, a periodic signal $x_T(t + T) = x_T(t)$ can be expanded into a Fourier series with coefficients $X[k]$, as shown in Eq.3.6. We can also consider this periodic signal as the convolution of a finite signal $x(t)$ which is zero outside the interval $0 < t < T$ and an impulse train with the same interval:

$$x_T(t) = x(t) * \sum_{n=-\infty}^{\infty} \delta(t - nT) \quad (3.164)$$

This is illustrated in Fig.3.7. According to the convolution theorem, the Fourier transform of this periodic signal can be found to be:

$$\mathcal{F}[x_T(t)] = \mathcal{F}[x(t)] * \sum_{n=-\infty}^{\infty} \delta(t - nT) = \mathcal{F}[x(t)] \mathcal{F}\left[\sum_{n=-\infty}^{\infty} \delta(t - nT)\right] \quad (3.165)$$

Here the two Fourier transforms on the right-hand side above are, respectively:

$$\mathcal{F}[x(t)] = \int_0^T x(t) e^{-j2\pi ft} dt \quad (3.166)$$

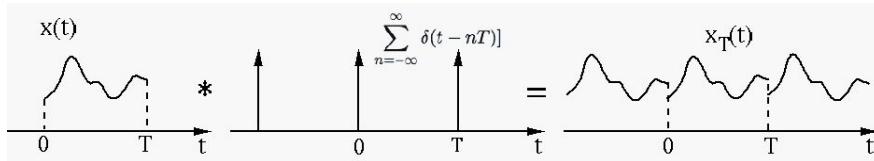


Figure 3.7 Generation of a periodic signal

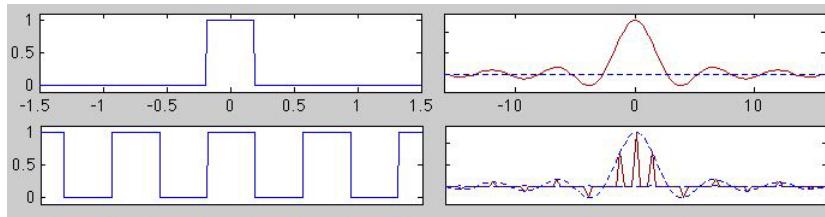


Figure 3.8 A periodic signal and its spectrum

and (Eq.3.163)

$$\mathcal{F}\left[\sum_{n=-\infty}^{\infty} \delta(t - nT)\right] = \frac{1}{T} \sum_{k=-\infty}^{\infty} \delta(f - kf_0) \quad (3.167)$$

Substituting these into Eq.3.165 we get:

$$\begin{aligned} \mathcal{F}[x_T(t)] &= \left[\int_0^T x(t) e^{-j2\pi f t} dt \right] \left[\frac{1}{T} \sum_{k=-\infty}^{\infty} \delta(f - kf_0) \right] \\ &= \sum_{k=-\infty}^{\infty} \frac{1}{T} \int_0^T x(t) e^{-j2\pi kf_0 t} dt \delta(f - kf_0) = \sum_{k=-\infty}^{\infty} X[k] \delta(f - kf_0) \end{aligned} \quad (3.168)$$

where $f_0 = 1/T$ is the fundamental frequency. This result indicates that the periodic signal has a discrete spectrum, which can be represented as an impulse train weighted by the Fourier coefficients $X[k]$. As an example, a square wave and its periodic version are shown respectively on the left of Fig.3.8, and their corresponding spectra are shown on the right. We see that the spectrum of the periodic version is composed of a set of impulses, weighted by the spectrum $X(f) = \mathcal{F}[x(t)]$.

Fig.3.9 shows a set of typical signals on the left and their Fourier spectra on the right.

3.2.5 The Uncertainty Principle

According to the property of time and frequency scaling (Eq.3.98), if a time function $x(t)$ is expanded ($a < 1$), its spectrum $X(f)$ will be compressed, and,

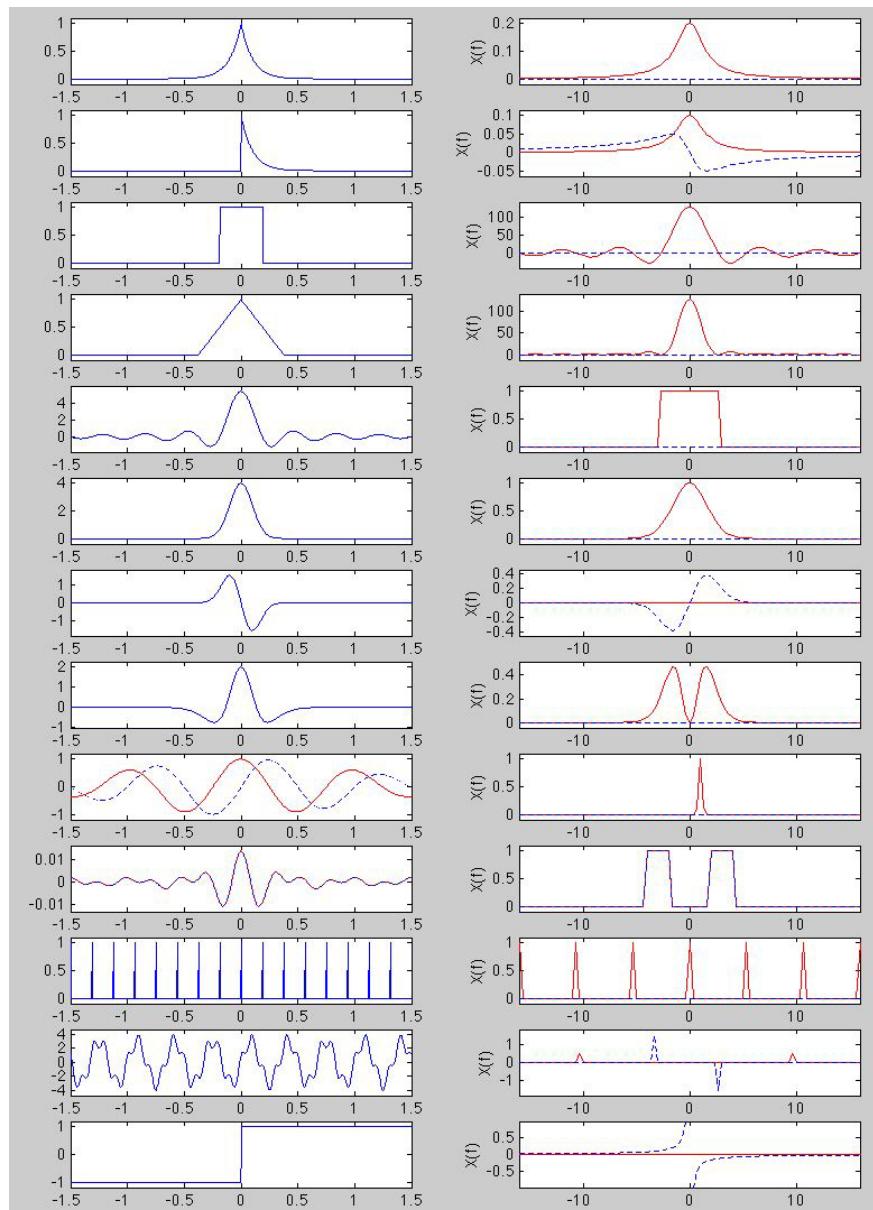


Figure 3.9 Examples of continuous-time Fourier transforms

A set of signals are shown on the left and their Fourier spectra are shown on the right (real and imaginary parts are shown in solid and dashed lines, respectively).

conversely, if $x(t)$ is compressed ($a > 1$), $X(f)$ will be expanded. This property indicates that if the energy of a signal is mostly concentrated within a short time range, then the energy in its spectrum is spread in a wide frequency range,

and vice versa. In particular, as two extreme examples, the Fourier transform of an impulse $\mathcal{F}[\delta(t)] = 1$ is a constant, while the Fourier transform of a constant $\mathcal{F}[1] = \delta(f)$ is an impulse.

This general phenomenon can be further quantitatively stated by the *uncertainty principle*. To do so, we need to borrow some concepts from probability theory. First, for a given function $x(t)$, we build another function:

$$p_x(t) = \frac{|x(t)|^2}{\|x(t)\|^2} = \frac{|x(t)|^2}{\langle x(t), x(t) \rangle} = \frac{|x(t)|^2}{\int_{-\infty}^{\infty} |x(t)|^2 dt} \quad (3.169)$$

where the denominator is the total energy of the signal $x(t)$ assumed to be finite, i.e., $x(t)$ is an energy signal. As $p_x(t)$ satisfies these conditions

$$p_x(t) > 0 \quad \text{and} \quad \int_{-\infty}^{\infty} p_x(t) dt = 1 \quad (3.170)$$

it can be considered as a probability density function over variable t , and how the function $x(t)$ spreads over time, i.e., the locality or the dispersion of $x(t)$, can be measured as the variance of this probability density $p_x(t)$:

$$\sigma_t^2 = \int_{-\infty}^{\infty} (t - \mu_t)^2 p_x(t) dt = \frac{1}{\|x(t)\|^2} \int_{-\infty}^{\infty} (t - \mu_t)^2 |x(t)|^2 dt \quad (3.171)$$

where μ_t is the mean of $p_x(t)$:

$$\mu_t = \int_{-\infty}^{\infty} t p_x(t) dt = \frac{1}{\|x(t)\|^2} \int_{-\infty}^{\infty} t |x(t)|^2 dt \quad (3.172)$$

The locality or dispersion of the spectrum of the signal can also be similarly measured as:

$$\sigma_f^2 = \frac{1}{\|X(f)\|^2} \int_{-\infty}^{\infty} (f - \mu_f)^2 |X(f)|^2 df \quad (3.173)$$

with μ_f defined as:

$$\mu_f = \frac{1}{\|X(f)\|^2} \int_{-\infty}^{\infty} f |X(f)|^2 df \quad (3.174)$$

Note that $\|x(t)\|^2 = \|X(f)\|^2$ due to Parseval's identity. Now the uncertainty principle can be stated as the following theorem:

Theorem 3.1. *Let $X(f) = \mathcal{F}[x(t)]$ be the spectrum of a given function $x(t)$ and σ_t^2 and σ_f^2 be defined as above. Then*

$$\sigma_t^2 \sigma_f^2 \geq \frac{1}{16\pi^2}, \quad \text{or} \quad \sigma_t \sigma_f \geq \frac{1}{4\pi} \quad (3.175)$$

Proof:

Without loss of generality, we assume $\mu_t = \mu_f = 0$, and consider

$$\sigma_t^2 \sigma_f^2 = \frac{1}{\|x(t)\|^4} \int_{-\infty}^{\infty} |tx(t)|^2 dt \int_{-\infty}^{\infty} |f X(f)|^2 df \quad (3.176)$$

Due to the time derivative property (Eq.3.117), we have:

$$\frac{1}{j2\pi} \mathcal{F} \left[\frac{d}{dt} x(t) \right] = f X(f) \quad (3.177)$$

also due to Parseval's identity we have:

$$\int_{-\infty}^{\infty} |f X(f)|^2 df = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \left| \frac{d}{dt} x(t) \right|^2 dt \quad (3.178)$$

Now the above becomes:

$$\sigma_t^2 \sigma_f^2 = \frac{1}{4\pi^2 \|x(t)\|^4} \int_{-\infty}^{\infty} |tx(t)|^2 dt \int_{-\infty}^{\infty} \left| \frac{d}{dt} x(t) \right|^2 dt \quad (3.179)$$

Applying the Cauchy-Schwarz inequality (Eq.2.30), we get:

$$\sigma_t^2 \sigma_f^2 \geq \frac{1}{4\pi^2 \|x(t)\|^4} \left[\int_{-\infty}^{\infty} t \bar{x}(t) \frac{d}{dt} x(t) dt \right]^2 \quad (3.180)$$

But as

$$\begin{aligned} \frac{d}{dt} [|x(t)|^2] &= \frac{d}{dt} [x(t) \bar{x}(t)] = \bar{x}(t) \frac{d}{dt} x(t) + x(t) \frac{d}{dt} \bar{x}(t) \\ &= 2 \operatorname{Re} \left[\frac{d}{dt} x(t) \bar{x}(t) \right] \leq 2 \frac{d}{dt} x(t) \bar{x}(t) \end{aligned} \quad (3.181)$$

replacing $\bar{x}(t) \frac{d}{dt} x(t)$ in the integrand by $\frac{d}{dt} [|x(t)|^2]/2$ we get:

$$\sigma_t^2 \sigma_f^2 \geq \frac{1}{4 \cdot 4\pi^2 \|x(t)\|^4} \left[\int_{-\infty}^{\infty} t \frac{d}{dt} [|x(t)|^2] dt \right]^2 \quad (3.182)$$

By integration by parts, the integral becomes:

$$\int_{-\infty}^{\infty} t \frac{d}{dt} [|x(t)|^2] dt = t |x(t)|^2 \Big|_{-\infty}^{\infty} - \int_{-\infty}^{\infty} |x(t)|^2 dt = - \int_{-\infty}^{\infty} |x(t)|^2 dt \quad (3.183)$$

Here we have assumed $\lim_{|t| \rightarrow \infty} tx^2(t) = 0$ for the reason that $x(t)$ contains finite amount of energy. Substituting this back into the inequality, we finally get:

$$\sigma_t^2 \sigma_f^2 \geq \frac{1}{4 \cdot 4\pi^2 \|x(t)\|^4} \left[\int_{-\infty}^{\infty} |x(t)|^2 dt \right]^2 = \frac{1}{16\pi^2} \quad (3.184)$$

This result is also referred to as the *Heisenberg uncertainty*, as it is analogous to the fact in quantum physics that the position and momentum of a particle cannot be accurately measured simultaneously, higher precision in one quantity implies lower precision in the other. Similarly here the uncertainty principle indicates an important fact: in the Fourier transform, the temporal and frequency localities of a signal cannot be achieved simultaneously.

3.3 Homework Problems

1. Show that the Fourier coefficients given in Eq.3.40 for the even triangle function are real and even ($X[k] = X[-k]$), and the Fourier coefficients given in Eq.3.41 for the odd triangle function are imaginary and odd ($X[k] = -X[-k]$).
2. If the square wave in Eq.3.31 is shifted to the left by $T/4$, it becomes an even function:

$$x_T(t) = \begin{cases} 1 & |t| < T/4 \\ 0 & T/4 < |t| < T/2 \end{cases} \quad (3.185)$$

Show that its Fourier series expansion becomes

$$\begin{aligned} x(t) &= \sum_{k=-\infty}^{\infty} X[k] e^{jk\omega_0 t} \\ &= \frac{1}{2} + \frac{2}{\pi} \left[\frac{\cos(\omega_0 t)}{1} - \frac{\cos(3\omega_0 t)}{3} + \frac{\cos(5\omega_0 t)}{5} + \dots \right] \end{aligned} \quad (3.186)$$

composed of odd harmonics of even cosine functions.

3. Find the Fourier series coefficients of an even triangle wave

$$x(t) = 2|t|/T \quad (3.187)$$

Express this even triangle wave $x(t)$ in terms of even cosine functions of different frequencies.

4. Given the signal below

$$x(t) = 3 \cos\left(\frac{\pi(10t-1)}{3}\right) - 2 \sin\left(\frac{\pi(5t+2)}{4}\right) \quad (3.188)$$

Find its fundamental frequency and period (if it is periodic) and then the Fourier series coefficients.

5. Find the Fourier series coefficients of the following signal:

$$x(t) = 2 \cos(12\pi t - \pi/2) - 3 \sin(20\pi t + \pi/3) \quad (3.189)$$

6. Find the Fourier spectrum of a truncated sinusoid

$$x(t) = \begin{cases} \cos(2\pi f_0 t) & |t| < \tau/2 \\ 0 & \text{else} \end{cases} \quad (3.190)$$

Sketch the spectrum.

7. Find the Fourier spectrum of the following signal:

$$x(t) = \begin{cases} -t & -\tau/2 < t < \tau/2 \\ 0 & \text{else} \end{cases} \quad (3.191)$$

Hint: $x(t)$ can be written as $x(t) = u(t+1) + u(t-1) - s(t)$ where

$$s(t) = \frac{2}{\tau} \int_{-\infty}^t r(t) dt = \begin{cases} 0 & t < -\tau/2 \\ 2t/\tau + 1 & -\tau/2 < t < 1 \\ 2 & t > 1 \end{cases}$$

is the integral of a square impulse with width τ :

$$r(t) = \begin{cases} 1 & |t| < \tau/2 \\ 0 & \text{else} \end{cases}$$

Find the spectrum of each of the three components and then sum them up.

8. Find the Fourier spectrum of the following signal:

$$x(t) = \begin{cases} 1 - t/\tau & 0 < t < \tau \\ 0 & \text{else} \end{cases} \quad (3.192)$$

9. Show that the Fourier transform of the step function $u(t)$ given in Eq.3.72 can also be obtained by:

$$\mathcal{F}[u(t)] = \lim_{a \rightarrow 0} \mathcal{F}[e^{-at}u(t)] = \lim_{a \rightarrow 0} \frac{a}{a^2 + \omega^2} + \lim_{a \rightarrow 0} \frac{-j\omega}{a^2 + \omega^2} \quad (3.193)$$

Hint: The first term approaches $\delta(f)/2$, i.e.,

$$\lim_{a \rightarrow 0} \frac{a}{a^2 + \omega^2} = \begin{cases} \infty & f = 0 \\ 0 & f \neq 0 \end{cases} \quad \text{and} \quad \int_{-\infty}^{\infty} \frac{a}{a^2 + \omega^2} df = \frac{1}{2} \quad (3.194)$$

You may need to use this integral:

$$\int \frac{dx}{a^2 + x^2} = \frac{1}{a} \tan^{-1} \left(\frac{x}{a} \right) \quad (3.195)$$

10. Find the Fourier spectra of the following functions:

- a. $e^{-at}u(t)$, ($a > 0$)
- b. $-e^{-at}u(-t)$, ($a < 0$)
- c. $e^{-a|t|}$, ($a > 0$)
- d. $\cos(\omega_0 t)e^{-at}u(t)$, ($a > 0$)
- e. $\sin(\omega_0 t)e^{-at}u(t)$, ($a > 0$)

11. Find the Fourier spectra of the following functions, and plot the magnitude and phase of each spectrum using any software tool of your choice (e.g., Matlab). (These functions are used as some “mother wavelet functions” in wavelet transforms.)

- a. Shannon wavelet:

$$\psi_1(t) = \frac{1}{\pi t} [\sin(2\pi f_2 t) - \sin(2\pi f_1 t)] \quad (3.196)$$

- b. Morlet wavelet:

$$\psi_2(t) = \frac{1}{\sqrt{2\pi}} e^{j\omega_0 t} e^{-t^2/2} \quad (3.197)$$

- c. Marr (Mexican hat) wavelet:

$$\psi_3(t) = \frac{1}{\sqrt{2\pi}\sigma^3} \left(1 - \frac{t^2}{\sigma^2} \right) e^{-t^2/2\sigma^2} \quad (3.198)$$

12. Find the Fourier spectrum of the following Gaussian modulated sinusoid:

$$x(t) = \cos(2\pi f_0 t) e^{-\pi(t/a)^2} \quad (3.199)$$

13. The result of the previous problem can be generalized to a sinusoid $\cos(2\pi f_0 t)$ modulated by any signal $s(t)$, the *amplitude modulation (AM)* in radio broadcasting. Assume $S(f) = \mathcal{F}[s(t)]$ is a triangle function

$$S(f) = 1 - \frac{|f|}{f_{max}} \quad (3.200)$$

where f_{max} is the highest frequency component contained in the signal $s(t)$. Obtain the spectrum $X(f)$ of the AM signal $x(t) = s(t) \cos(2\pi f_0 t)$ and plot $X(f)$ in frequency domain.

Another signal $y(t) = x(t) \cos(2\pi f_0 t)$ can be generated as the AM version of $x(t)$. Find and plot $Y(f) = \mathcal{F}[y(t)]$.

14. In Eq.3.164 we considered a convolution of an impulse train and another signal $x(t)$ of finite duration:

$$y(t) = x(t) * \sum_{n=-\infty}^{\infty} \delta(t - nT) \quad (3.201)$$

Here we assume $x(t)$ is the triangle function given in Eq.3.156 and its spectrum $X(f) = \tau \text{sinc}^2(f\tau)$ given in Eq.3.157.

Plot (sketch) both the convolution $y(t)$ in time domain and its spectrum $Y(f) = \mathcal{F}[y(t)]$ in frequency domain (identify all points at which $X(f) = 0$) in these two situations: (a) $T > 2\tau$ and (b) $T < 2\tau$. What is the essential difference between these two cases in both time and frequency domains?

(Note that as both the triangle function and the impulse train are even functions, time-frequency duality applies, i.e., the time and frequency domains can be interchanged. Then what we observe here is the basis for the sampling theorem to be considered in the next chapter.)

4 Discrete-Time Fourier Transform

4.1 Discrete-Time Fourier Transform

4.1.1 Fourier Transform of Discrete Signals

To use the digital technology to process a continuous time signal $x(t)$, an analog-to-digital converter (ADC, A/D) is needed to discretize the signal so that it becomes a sequence of time samples $x[n] = x(nt_0) = x(n/F)$ ($n = 0, \pm 1, \pm 2, \dots$), where t_0 is the *sampling period*, the time interval between two consecutive samples, and $F = 1/t_0$ is the *sampling rate* or *sampling frequency*, the number of samples per unit time. The sampled signal $x_s(t)$ can be represented as the product of the signal and the sampling function, an impulse train (also called a Dirac comb or Shah function):

$$x_s(t) = x(t) \text{comb}(t) = x(t) \sum_{n=-\infty}^{\infty} \delta(t - nt_0) = \sum_{n=-\infty}^{\infty} x[n] \delta(t - nt_0) \quad (4.1)$$

where $x[n] = x(nt_0) = x(n/F)$ is the n th sample of the signal $x(t)$ evaluated at $t = nt_0 = n/F$. The Fourier transform of this sampled signal is:

$$\begin{aligned} X(f) &= \mathcal{F}[x_s(t)] = \int_{-\infty}^{\infty} \left[\sum_{n=-\infty}^{\infty} x[n] \delta(t - nt_0) \right] e^{-j2\pi ft} dt \\ &= \sum_{n=-\infty}^{\infty} x[n] \int_{-\infty}^{\infty} \delta(t - nt_0) e^{-j2\pi ft} dt = \sum_{n=-\infty}^{\infty} x[n] e^{-j2n\pi f t_0} \end{aligned} \quad (4.2)$$

This is the spectrum of the discrete signal $x[n]$, which is periodic with the sampling frequency F as the period:

$$X(f + F) = \sum_{n=-\infty}^{\infty} x[n] e^{-j2n\pi(f+F)t_0} dt = \sum_{n=-\infty}^{\infty} x[n] e^{-j2n\pi f t_0} dt = X(f) \quad (4.3)$$

where we have used the fact that $e^{-j2n\pi F t_0} = e^{-j2n\pi} = 1$. Here we could use $X_F(f)$ to denote this periodic spectrum, to distinguish it from the non-periodic spectrum of the continuous time signal before Discretization, just as we used $x_T(t)$ to denote a periodic time signal with period T , to distinguish it from a non-periodic signal $x(t)$. However, such subscripts may be dropped for simplicity when no confusion will be caused.

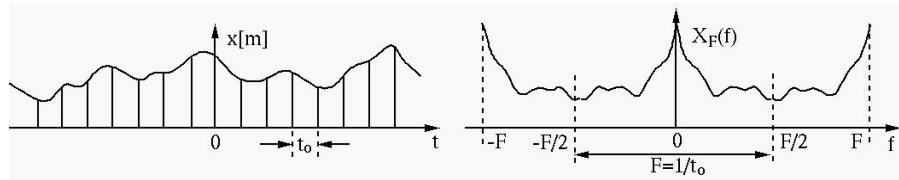


Figure 4.1 Fourier transform of discrete signals

To get the time samples of the discrete signal back from its spectrum $X_F(f)$, we multiply $e^{j2m\pi ft_0}/F = e^{j2m\pi f/F}/F$ on both sides of Eq. 4.2 and integrate with respect to f over a period F :

$$\begin{aligned} \frac{1}{F} \int_0^F X(f) e^{j2m\pi ft_0} df &= \frac{1}{F} \sum_{n=-\infty}^{\infty} x[n] \int_0^F e^{-j2(m-n)\pi f t_0} df \\ &= \sum_{n=-\infty}^{\infty} x[n] \delta[n - m] = x[n], \quad (m = 0, \pm 1, \pm 2, \dots) \end{aligned} \quad (4.4)$$

where we have used Eq.1.33 (with different variables names). This is the inverse discrete-time Fourier transform. With a minor modification of the scaling factor $1/F$ of Eqs.4.2 and 4.4, they can be written as a pair of the discrete-time Fourier transform (DTFT):

$$\begin{aligned} X_F(f) &= \mathcal{F}[x[n]] = \frac{1}{\sqrt{F}} \sum_{n=-\infty}^{\infty} x[n] e^{-j2n\pi f t_0} = \frac{1}{\sqrt{F}} \sum_{n=-\infty}^{\infty} x[n] e^{-j2n\pi f/F} \\ x[n] &= \mathcal{F}^{-1}[X_F(f)] = \frac{1}{\sqrt{F}} \int_0^F X_F(f) e^{j2n\pi f t_0} df = \frac{1}{\sqrt{F}} \int_0^F X_F(f) e^{j2n\pi f/F} df \\ &\quad (n = 0, \pm 1, \pm 2, \dots) \end{aligned} \quad (4.5)$$

where the first and second equations are respectively the forward and inverse DTFT. Comparing these equations with Eqs.2.128 and 2.129, we see that the DTFT is actually the representation of a signal vector $\mathbf{x} = [\dots, x[n], \dots]^T$ by a set of uncountably infinite orthonormal basis vectors:

$$\phi(f) = [\dots, e^{j2\pi nf/F}, \dots]^T / \sqrt{F}, \quad (0 < f < F) \quad (4.6)$$

satisfying Eq.2.126:

$$\langle \phi(f), \phi(f') \rangle = \frac{1}{F} \sum_{n=-\infty}^{\infty} e^{j2\pi n(f-f')/F} = \delta(f - f') \quad (4.7)$$

Now the vector $\mathbf{x} = [\dots, x[n], \dots]^T$ in the vector space spanned by the basis can be expressed as a linear combination, an integral, of the basis vectors:

$$\mathbf{x} = \int_0^F X_F(f) \phi(f) df \quad (4.8)$$

the element form of which is the inverse DTFT in Eq.4.5, and the coefficient function $X_F(f)$ can be found as the projection of the vector \mathbf{x} onto the basis

vector $\phi(f)$:

$$X_F(f) = \langle \mathbf{x}, \phi(f) \rangle = \mathbf{x}^T \bar{\phi}(f) = \frac{1}{\sqrt{F}} \sum_{n=-\infty}^{\infty} x[n] e^{-j2\pi n f / F} \quad (4.9)$$

which is the forward DTFT in Eq.4.5.

As a unitary transform, the DTFT also conserves inner product:

$$\begin{aligned} \langle \mathbf{x}, \mathbf{y} \rangle &= \mathbf{x}^T \bar{\mathbf{y}} = \sum_{n=-\infty}^{\infty} x[n] \bar{y}[n] \\ &= \sum_{n=-\infty}^{\infty} \left[\frac{1}{\sqrt{F}} \int_F X(f) e^{j2\pi n t_0 f} df \right] \left[\frac{1}{\sqrt{F}} \int_F \bar{Y}(f') e^{-j2\pi n t_0 f'} df' \right] \\ &= \int_F X(f) \int_F \bar{Y}(f') \left[\frac{1}{F} \sum_{m=-\infty}^{\infty} e^{-j2\pi m t_0 (f-f')} \right] df' df \\ &= \int_F X(f) \int_F \bar{Y}(f') \delta(f-f') df' df = \int_F X(f) \bar{Y}(f) df \\ &= \langle X(f), Y(f) \rangle \end{aligned} \quad (4.10)$$

In particular, when $\mathbf{x} = \mathbf{y}$, we get Parseval's identity:

$$\langle \mathbf{x}, \mathbf{x} \rangle = \sum_{n=-\infty}^{\infty} |x[n]|^2 = \int_F |X(f)|^2 df = \langle X(f), X(f) \rangle \quad (4.11)$$

indicating that the energy or information contained in the signal is preserved by the DTFT.

Comparing the pair of equations in Eq.4.5 with the Fourier series expansion of a periodic signal $x_T(t)$ in Eq. 3.5 we see a duality between time and frequency domains:

- A continuous and periodic time signal $x_T(t)$ (with period $T = 1/f_0$) is a function in the space spanned by a set of countably infinite periodic functions $\phi_k(t) = e^{j2\pi k f_0 t} / \sqrt{T}$ ($k = 0, \pm 1, \pm 2, \dots$). Its spectrum is non-periodic and discrete (with a frequency interval $f_0 = 1/T$ between two consecutive frequency components).
- A non-periodic and discrete time signal $x[n]$ (with a time interval $t_0 = 1/F$ between two consecutive samples) is a vector in the space spanned by a set of uncountably infinite (a continuum of) vectors $\phi(f) = [\dots, e^{j2\pi n t_0 f}, \dots] / \sqrt{F}$ ($0 \leq f < F$). Its spectrum is continuous and periodic (with period $F = 1/t_0$).

This duality between time and frequency is obviously due to the symmetry in the most generic definition of the forward and inverse Fourier transforms in Eq. 3.58.

Eqs.4.2 and 4.4 can also be expressed in terms of angular frequency $\omega = 2\pi f$ as:

$$\begin{aligned} X_\Omega(\omega) &= \sum_{n=-\infty}^{\infty} x[n]e^{-jn\omega t_0} \\ x[n] &= \frac{1}{\Omega} \int_0^\Omega X_\Omega(\omega)e^{jn\omega t_0} d\omega, \quad (n = 0, \pm 1, \pm 2, \dots) \end{aligned} \quad (4.12)$$

where $X_\Omega(\omega + \Omega)$ is the spectrum with period $\Omega = 2\pi F$. Moreover, once a continuous signal is sampled to become a sequence of discrete values, the sampling period t_0 may not be of interest anymore during the subsequent digital signal processing, and can be assumed to be $t_0 = 1$, then the sampling frequency also becomes unit $F = 1/t_0 = 1$, and the Fourier transform pair in Eq.4.5 of the discrete signal can be simply expressed as:

$$\begin{aligned} X(f) &= \sum_{n=-\infty}^{\infty} x[n]e^{-j2n\pi f}, \quad \text{or} \quad X(\omega) = \sum_{n=-\infty}^{\infty} x[n]e^{-jn\omega} \\ x[n] &= \int_0^1 X(f)e^{j2n\pi f} df = \frac{1}{2\pi} \int_0^{2\pi} X(\omega)e^{jn\omega} d\omega \end{aligned} \quad (4.13)$$

Now the periodicity of the spectrum becomes $X(f + 1) = X(f)$ or $X(\omega + 2\pi) = X(\omega)$.

In some literatures, the DTFT spectrum $X(f)$ or $X(\omega)$ is also denoted by $X(e^{j\omega})$, because it takes this form when treated as a special case of the *z-transform*, to be discussed in Chapter 6. However, all these different forms are just some notational variations of the same spectrum, a function of frequency f or angular frequency $\omega = 2\pi f$. We will use these notations interchangeably, whichever most convenient and suitable in the specific discussion, as no confusion should be caused given the context.

Example 4.1: Here we consider the Fourier transform of a few special signals:

- The Kronecker delta or a discrete unit impulse $x[n] = \delta[n]$:

$$\mathcal{F}[\delta[n]] = \sum_{n=-\infty}^{\infty} \delta[n]e^{-j2n\pi f} = e^{-j2\pi 0f} = 1 \quad (4.14)$$

- The constant function, a train of unit impulses, $x[n] = 1$:

$$\mathcal{F}[1] = \sum_{n=-\infty}^{\infty} e^{j2n\pi f} = \sum_{k=-\infty}^{\infty} \delta(f - k) = 2\pi \sum_{k=-\infty}^{\infty} \delta(\omega - 2k\pi) \quad (4.15)$$

Here we have used Eq.1.35. The spectrum is also an impulse train in frequency domain.

- The discrete sign function is defined as:

$$sgn[n] = \begin{cases} -1 & n < 0 \\ 0 & n = 0 \\ 1 & n > 0 \end{cases} \quad (4.16)$$

Its DTFT spectrum is:

$$\mathcal{F}[sgn[n]] = -\sum_{n=-\infty}^{-1} e^{-jn\omega} + \sum_{n=1}^{\infty} e^{-jn\omega} = -\sum_{m=1}^{\infty} e^{jm\omega} + \sum_{n=1}^{\infty} e^{-jn\omega} \quad (4.17)$$

Consider the first summation as the following limit when the real parameter $0 < a < 1$ approaches zero:

$$\lim_{a \rightarrow 1} \left[-\sum_{n=1}^{\infty} (ae^{j\omega})^n \right] = \lim_{a \rightarrow 1} \left[a - \sum_{n=0}^{\infty} (ae^{j\omega})^n \right] = \lim_{a \rightarrow 1} \left[a - \frac{1}{1 - ae^{j\omega}} \right] \quad (4.18)$$

Similarly the second summation can be written as:

$$\lim_{a \rightarrow 1} \left[\sum_{n=1}^{\infty} (ae^{-j\omega})^n \right] = \lim_{a \rightarrow 1} \left[\sum_{n=0}^{\infty} (ae^{-j\omega})^n - a \right] = \lim_{a \rightarrow 1} \left[\frac{1}{1 - ae^{-j\omega}} - a \right] \quad (4.19)$$

Note that in these limits we cannot simply replace a by 1 due to the singularity at $\omega = 2k\pi$ for any integer k . However, we can do so to the sum of the two terms, which is an odd function and is zero at $\omega = 2k\pi$:

$$\mathcal{F}[sgn[n]] = \lim_{a \rightarrow 1} \left[\frac{1}{1 - ae^{-j\omega}} - \frac{1}{1 - ae^{j\omega}} \right] = \frac{1 + e^{-j\omega}}{1 - e^{-j\omega}} = \frac{j \sin \omega}{\cos \omega - 1} \quad (4.20)$$

- The unit step function is defined as:

$$u[n] = \begin{cases} 0 & n < 0 \\ 1 & n \geq 0 \end{cases} \quad (4.21)$$

Note that $u[0] = 1$, unlike $u(0) = 1/2$ in the continuous case. Following the DTFT definition above its spectrum can be directly obtained from Eq.1.37.

$$\mathcal{F}[u[n]] = \sum_{n=0}^{\infty} e^{-j2\pi nf} = \frac{1}{1 - e^{-j2\pi f}} + \frac{1}{2} \sum_{k=-\infty}^{\infty} \delta(f - k) \quad (4.22)$$

Alternatively, we can write $u[n]$ as:

$$u[n] = \frac{1}{2} [1 + \delta[n] + sgn[n]] = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases} \quad (4.23)$$

and carry out the Fourier transform to each of the three terms to get:

$$\begin{aligned} \mathcal{F}[u[n]] &= \frac{1}{2} [\mathcal{F}[1] + \mathcal{F}[\delta[n]] + \mathcal{F}[sgn[n]]] \\ &= \frac{1}{2} \left[\sum_{k=-\infty}^{\infty} \delta(f - k) + 1 + \frac{1 + e^{-j\omega}}{1 - e^{-j\omega}} \right] = \frac{1}{1 - e^{-j2\pi f}} + \frac{1}{2} \sum_{k=-\infty}^{\infty} \delta(f - k) \end{aligned} \quad (4.24)$$

4.1.2 Properties of DTFT

As one of variations of the generic Fourier transform in Eq.3.58, the DTFT shares all of the properties discussed considered in previous chapter, but in different forms. Here we assume $X(f) = \mathcal{F}[x[n]]$ and $Y(f) = \mathcal{F}[y[n]]$. Proofs of many of these properties are not given as they can be easily derived from the definition. The reader are encouraged to prove them as homework problems.

- **Linearity**

$$\mathcal{F}[ax[n] + by[n]] = aX(f) + bY(f) \quad (4.25)$$

- **Periodicity**

$$X(f + k) = X(f) \quad (4.26)$$

where k is any integer.

- **Parseval's identity**

$$\begin{aligned} \langle x, x \rangle &= \sum_{n=-\infty}^{\infty} |x[n]|^2 = \int_0^1 |X(f)|^2 df \\ &= \frac{1}{2\pi} \int_0^{2\pi} |X(\omega)|^2 d\omega = \langle X(f), X(f) \rangle \end{aligned} \quad (4.27)$$

This was given in Eq.4.11.

- **Complex conjugate**

$$\mathcal{F}[\bar{x}[n]] = \overline{X}(-f) \quad (4.28)$$

- **Time reversal**

$$\mathcal{F}[x[-n]] = X(-f) \quad (4.29)$$

Combining the above with the previous property, we also have:

$$\mathcal{F}[\bar{x}[-n]] = \overline{X}(f) \quad (4.30)$$

In particular if $\bar{x}[n] = x[n]$ is real, then

$$\mathcal{F}[x[-n]] = X(-f) = \overline{X}(f) \quad (4.31)$$

- **Time and frequency shift**

$$\mathcal{F}[x[n \pm n_0]] = e^{\pm j 2n_0 f} X(f) = e^{\pm j n_0 \omega} X(\omega) \quad (4.32)$$

$$\mathcal{F}[e^{\mp j 2n\pi f_0} x[n]] = X(f \pm f_0) = X(\omega \pm 2\pi f_0) \quad (4.33)$$

- **Correlation**

$$\mathcal{F}[x[n] \star y[n]] = X(f)\bar{Y}(f) = S_{xy}(f) \quad (4.34)$$

where $S_{xy}(f) = X(f)\bar{Y}(f)$ is the *cross power spectral density* of the two signals, and $x[n] \star y[n]$ is the cross-correlation of the two sequences $x[n]$ and $y[n]$ defined as:

$$r_{xy}[n] = x[n] \star y[n] = \sum_m x[m]\bar{y}[m-n] \quad (4.35)$$

In particular, if both signals $\bar{x}[n] = x[n]$ and $\bar{y}[n] = y[n]$ are real, we have

$$\mathcal{F}[x[n] \star y[n]] = X(f)Y(-f) \quad (4.36)$$

- **Time and frequency convolutions**

$$\mathcal{F}[x[n] * y[n]] = X(f)Y(f) \quad (4.37)$$

$$\mathcal{F}[x[n]y[n]] = X(f) * Y(f) \quad (4.38)$$

Note that both $X(f+1) = X(f)$ and $Y(f+1) = Y(f)$ are periodic, and their convolution is called periodic convolution.

- **Time differencing**

Corresponding to the first order derivative of a continuous signal $dx(t)/dt = \lim_{\Delta \rightarrow 0} [x(t + \Delta) - x(t)]/\Delta$, the first order difference of a discrete signal is simply defined as $x[n] - x[n - 1]$. Based on the time shift property, we have:

$$\mathcal{F}[x[n] - x[n - 1]] = (1 - e^{-j2\pi f})X(f) \quad (4.39)$$

- **Time accumulation**

Corresponding to the integral of a continuous signal, the accumulation of a discrete signal is a summation of all its samples $x[n]$ from $n = -\infty$ up to $n = m$, and its Fourier transform is:

$$\mathcal{F}\left[\sum_{m=-\infty}^n x[m]\right] = \frac{1}{1 - e^{-j2\pi f}}X(f) + \frac{X(0)}{2} \sum_{k=-\infty}^{\infty} \delta(f - k) \quad (4.40)$$

The accumulation can be expressed as the following convolution:

$$\sum_{m=-\infty}^n x[m] = \sum_{m=-\infty}^{\infty} u[n-m]x[m] = u[n] * x[n] \quad (4.41)$$

where $u[n-m] = 0$ if $m > n$.

The DTFT of this convolution can be easily found according to the time convolution property:

$$\mathcal{F}\left[\sum_{m=-\infty}^n x[m]\right] = \mathcal{F}[u[n] * x[n]] \quad (4.42)$$

Comparing Eqs.4.39 and 4.40, we see that differencing and accumulation are the inverse operations of each other, just like the continuous time derivative

and integral which are also the inverse operations of each other (Eqs.3.117 and 3.121). The second term of the right-hand side in Eq.4.40 represents the DC component in the signal $x[n]$, which is not needed in Eq.4.39 as differencing operation is insensitive to DC component.

- **Frequency differentiation**

$$\mathcal{F}[n x[n]] = \frac{j}{2\pi} \frac{d}{df} X(f) \quad (4.43)$$

- **Modulation**

Here modulation means every odd sample of the signal $x[n]$ is negated.

$$\mathcal{F}[(-1)^n x[n]] = X\left(f + \frac{1}{2}\right) = X\left(f - \frac{1}{2}\right) \quad (4.44)$$

Proof: If we let $f_0 = 1/2$ in Eq.4.33 for the frequency shift property, and note $e^{j2n\pi f_0} = e^{jn\pi} = (-1)^n$, we get Eq.4.44.

- **Down-sampling**

$$\mathcal{F}[x_{(2)}[n]] = \mathcal{F}[x[2n]] = \frac{1}{2} \left[X\left(\frac{f}{2}\right) + X\left(\frac{f+1}{2}\right) \right] \quad (4.45)$$

Here the down-sampled version $x_{(2)}[n]$ of a signal $x[n]$ is composed of all the even terms of the signal with all odd terms dropped, i.e., $x_{(2)}[n] = x[2n]$. Down-sampling of a discrete signal corresponds to the compression of a continuous signal (Eq.3.98 with $a = 2$):

$$\mathcal{F}[x(2t)] = \frac{1}{2} X\left(\frac{f}{2}\right) \quad (4.46)$$

Proof:

$$\begin{aligned} \mathcal{F}[x_{(2)}[n]] &= \sum_{n=-\infty}^{\infty} x[2n] e^{-j2\pi n f} = \sum_{m=-\infty}^{\infty} x[m] e^{-j\pi m f} \\ &= \frac{1}{2} \left[\sum_{m=-\infty}^{\infty} x[m] e^{-j\pi m f} + \sum_{m=-\infty}^{\infty} (-1)^m x[n] e^{-j\pi m f} \right] \\ &= \frac{1}{2} \left[\sum_{m=-\infty}^{\infty} x[m] e^{-j\pi m f} + \sum_{m=-\infty}^{\infty} x[m] e^{-j\pi m(f+1)} \right] \\ &= \frac{1}{2} \left[X\left(\frac{f}{2}\right) + X\left(\frac{f+1}{2}\right) \right] \end{aligned} \quad (4.47)$$

Conceptually, the down-sampling of a given discrete signal $x[n]$ can be realized in the following three steps:

- Obtain its modulation $x[n](-1)^n = x[n]e^{jn\pi}$. Due to the frequency shift property, this corresponds to the spectrum shifted by $1/2$:

$$\mathcal{F}[(-1)^n x[n]] = \mathcal{F}[e^{jn\pi} x[n]] = X(f + 1/2) \quad (4.48)$$

- Obtain the average of the signal and its modulation in both time and frequency domains:

$$\mathcal{F}\left[\frac{1}{2}[x[n] + x[n](-1)^n]\right] = \frac{1}{2} \left[X(f) + X(f + \frac{1}{2}) \right] \quad (4.49)$$

- Remove odd samples of the average to get $x_{(2)}[n]$. In frequency domain, this corresponds to replacing f by $f/2$:

$$\mathcal{F}[x_{(2)}[n]] = \frac{1}{2} \left[X(\frac{f}{2}) + X(\frac{f+1}{2}) \right] \quad (4.50)$$

- **Up-sampling (time expansion)**

$$\mathcal{F}[x^{(k)}[n]] = X(kf) \quad (4.51)$$

Here $x^{(k)}[n]$ is defined as:

$$x^{(k)}[n] = \begin{cases} x[n/k] & \text{if } n \text{ is a multiple of } k \\ 0 & \text{else} \end{cases} \quad (4.52)$$

i.e. $x^{(k)}[n]$ is obtained by inserting $k - 1$ zeros between every two consecutive samples of $x[n]$. Correspondingly its spectrum $X(kf)$ in frequency domain is compressed k times with the same magnitude. Note that up-sampling is quite different from the time scaling of a continuous signal in Eq.3.98 with $a = 1/k$: $\mathcal{F}[x(t/k)] = kX(kf)$ in which case the signal $x(t)$ is expanded by k , and consequently its Fourier spectrum $X(f)$ is compressed by k , while its magnitude is also scaled up by k .

Proof:

$$\mathcal{F}\left[x^{(k)}[n]\right] = \sum_{n=-\infty}^{\infty} x[n/k]e^{-j2n\pi f} = \sum_{m=-\infty}^{\infty} x[m]e^{-j2km\pi f/k} = X(kf) \quad (4.53)$$

Note that the change of the summation index from m to $m = n/k$ has no effect as the terms skipped are all zeros.

Combining the down and up-sampling above, we see that if a signal $x[n]$ with $X(f) = \mathcal{F}[x[n]]$ is first down-sampled and then up-sampled, its DTFT transform is:

$$\mathcal{F}[(x_{(2)})^{(2)}[n]] = \frac{1}{2} \left[X(f) + X(f + \frac{1}{2}) \right] \quad (4.54)$$

Example 4.2: Here we consider the up-sampling, modulation and down-sampling of a discrete signal of square wave $x[n]$ with seven non-zero samples, as shown in Fig.4.2.

- The square wave and its spectrum, a sinc function, are shown in the first row of the figure. Note that the DC component is 7, the number of non-zero samples in the signal $x[n]$.

- The up-sampled version $x^{(2)}[n]$ of the signal in both time and frequency domains are shown in the second row. Note that unlike time expansion of continuous signals, here the magnitude of the spectrum is not scaled by up-sampling.
- The up-sampled version $x^{(3)}[n]$ of the signal in both time and frequency domains are shown in the third row.
- The modulation of the signal is shown in the fourth row. Note that all odd-numbered samples are negated and the spectrum is shifted by 1/2, and its DC component is -1 (3 positive samples and 4 negative samples in time domain).
- The average of the signal and its modulation are shown in the fifth row. Note that the odd numbered sampled becomes zero. In frequency domain, the averaged spectrum is no longer a sinc function, but a sinusoid (due to the two time samples at $n = -2$ and $n = 2$ with a DC component 1 (due to the time sample at $n = 0$).
- Finally as shown in the last row, the time signal is compressed by a factor 2 with all odd numbered samples (all of value zero) dropped. Correspondingly, the spectrum is expanded by a factor of 2.

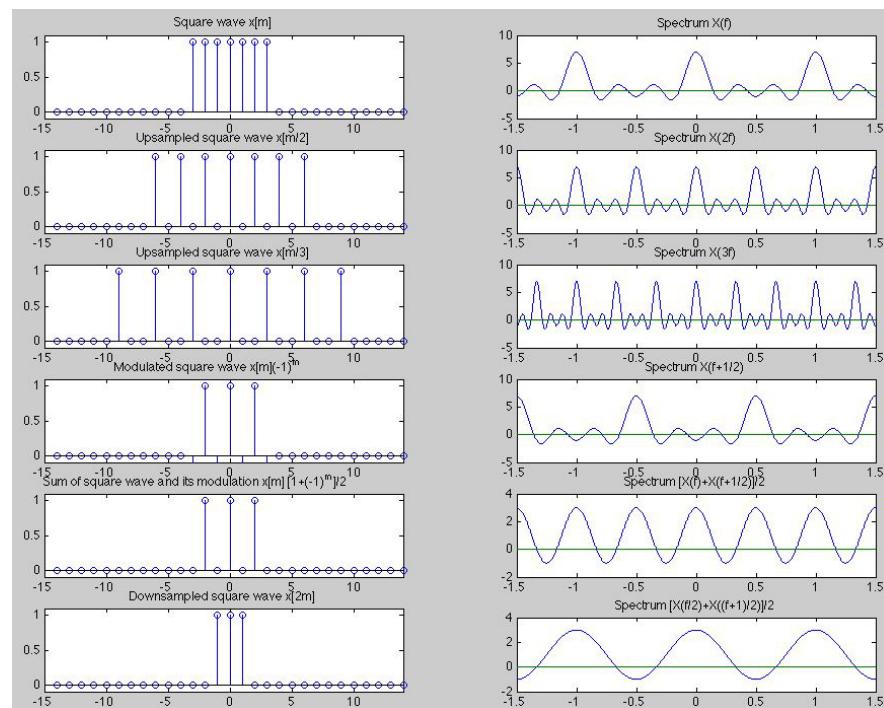


Figure 4.2 Modulation, up and down-sampling

The square wave and its modulation, up and down sampling versions on the left, and their spectra (showing three periods) on the right.

Example 4.3: Here we consider the convolution of two finite discrete signals $x[n]$ of length M and $h[n]$ of size N , i.e., $x[n]$ is zero outside the range $0 \leq n \leq M - 1$; and $h[n]$ is zero outside the range $0 \leq n \leq N - 1$. Their convolution is:

$$y[n] = x[n] * h[n] = \sum_{m=-\infty}^{\infty} x[m]h[n-m] \quad (4.55)$$

Note that the range for $h[n-m]$ is $0 \leq n-m < N$, i.e., $m \leq n < N+m$. But as $0 \leq m \leq M-1$, we get the range $0 \leq m \leq n \leq N+m-1 \leq N+M-1$ for $y[n]$ outside which $y[n] = 0$. Specifically, let:

$$\mathbf{x} = [1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8]^T, \quad \mathbf{h} = [1 \ 2 \ 3]^T \quad (4.56)$$

Here $M = 8$, $N = 3$, and the length of the result of this convolution is $M + N - 1 = 8 + 3 - 1 = 10$, any $y[n]$ outside the range of $0 < n < 9$ is zero. This convolution can be illustrated below:

$$\begin{array}{ccccccccccccccccccccc} \hline m & \cdots & -1 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & \cdots \\ \hline x[m] & \cdots & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 0 & 0 & 0 & \cdots \\ \hline h[0-m] & \cdots & 2 & 1 & & & & & & & & & & & \cdots \\ h[1-m] & \cdots & 3 & 2 & 1 & & & & & & & & & & \cdots \\ h[2-m] & \cdots & & 3 & 2 & 1 & & & & & & & & & \cdots \\ h[3-m] & \cdots & & & 3 & 2 & 1 & & & & & & & & \cdots \\ h[4-m] & \cdots & & & & 3 & 2 & 1 & & & & & & & \cdots \\ h[5-m] & \cdots & & & & & 3 & 2 & 1 & & & & & & \cdots \\ h[6-m] & \cdots & & & & & & 3 & 2 & 1 & & & & & \cdots \\ h[7-m] & \cdots & & & & & & & 3 & 2 & 1 & & & & \cdots \\ h[8-m] & \cdots & & & & & & & & 3 & 2 & 1 & & & \cdots \\ h[9-m] & \cdots & & & & & & & & & 3 & 2 & 1 & & \cdots \\ h[10-m] & \cdots & & & & & & & & & & 3 & 2 & 1 & \cdots \\ \hline y[n] & \cdots & 0 & 1 & 4 & 10 & 16 & 22 & 28 & 34 & 40 & 37 & 24 & 0 & \cdots \end{array} \quad (4.57)$$

4.1.3 Discrete Time Fourier Transform of Typical Functions

- **Constant:** If $x[n] = 1$ in Eq.4.1, we get an impulse train in time domain:

$$x_s(t) = \text{comb}(t) = \sum_{n=-\infty}^{\infty} \delta(t - nt_0) = \sum_{n=-\infty}^{\infty} \delta(t - n/F) \quad (4.58)$$

whose discrete-time Fourier transform is also an impulse train in frequency domain:

$$\mathcal{F}[x[n]] = \mathcal{F}[1] = \sum_{n=-\infty}^{\infty} e^{j2n\pi f} = \sum_{k=-\infty}^{\infty} \delta(f - k) \quad (4.59)$$

The last equal sign is due to Eq.1.35.

- **Complex exponential**

Applying frequency shift property to the previous result we get:

$$\mathcal{F}[e^{j2n\pi f_0}] = \sum_{k=-\infty}^{\infty} \delta(f - f_0 - k) \quad (4.60)$$

- **Sinusoids**

$$\begin{aligned} \mathcal{F}[\cos(2n\pi f_0)] &= \frac{1}{2} [\mathcal{F}[e^{j2n\pi f_0}] + \mathcal{F}[e^{-j2n\pi f_0}]] \\ &= \frac{1}{2} \left[\sum_{k=-\infty}^{\infty} \delta(f - f_0 - k) + \sum_{k=-\infty}^{\infty} \delta(f + f_0 - k) \right] \end{aligned} \quad (4.61)$$

Similarly we have:

$$\begin{aligned} \mathcal{F}[\sin(2n\pi f_0)] &= \frac{1}{2j} [\mathcal{F}[e^{j2n\pi f_0}] - \mathcal{F}[e^{-j2n\pi f_0}]] \\ &= \frac{1}{2j} \left[\sum_{k=-\infty}^{\infty} \delta(f - f_0 - k) - \sum_{k=-\infty}^{\infty} \delta(f + f_0 - k) \right] \end{aligned} \quad (4.62)$$

- **Kronecker delta**

$$\mathcal{F}[\delta[n]] = \sum_{n=-\infty}^{\infty} \delta[n] e^{j2n\pi f} = e^{j0} = 1 \quad (4.63)$$

- **Sign function**

$$\mathcal{F}[sgn[n]] = \frac{-e^{j2\pi f}}{1 - e^{j2\pi f}} + \frac{e^{-j2\pi f}}{1 - e^{-j2\pi f}} = \frac{1 + e^{-j2\pi f}}{1 - e^{-j2\pi f}} = \frac{j \sin \omega}{\cos \omega - 1} \quad (4.64)$$

This is given in Eq.4.20.

- **Unit step function**

$$\mathcal{F}[u[n]] = \frac{1}{1 - e^{-j2\pi f}} + \frac{1}{2} \sum_{k=-\infty}^{\infty} \delta(f - k) \quad (4.65)$$

This is given in Eq.4.24.

- **Exponential decay**

First consider a right-sided exponential function:

$$x[n] = a^n u[n], \quad (|a| < 1) \quad (4.66)$$

$$\mathcal{F}[a^n u[n]] = \sum_{n=0}^{\infty} (ae^{-j2\pi f})^n = \frac{1}{1 - ae^{-j2\pi f}} \quad (4.67)$$

Next consider the two-sided version:

$$x[n] = a^{|n|} = a^n u[n] + a^{-n} u[-n - 1], \quad (|a| < 1) \quad (4.68)$$

The transform of the first term is the same as before, while the transform of the second term is:

$$\begin{aligned}\mathcal{F}[a^{-n}u[-n-1]] &= \sum_{m=-\infty}^{-1} a^{-n}e^{-j2n\pi f} = \sum_{n=0}^{\infty} (ae^{j2\pi f})^n - 1 \\ &= \frac{ae^{j2\pi f}}{1 - ae^{j2\pi f}}\end{aligned}\quad (4.69)$$

The over all transform is:

$$\mathcal{F}[a^{|n|}] = \frac{1}{1 - ae^{-j2\pi f}} + \frac{ae^{j2\pi f}}{1 - ae^{j2\pi f}} = \frac{1 - a^2}{1 + a^2 - 2a \cos(2\pi f)} \quad (4.70)$$

- **Square wave**

$$x[n] = \begin{cases} 1 & |n| \leq N \\ 0 & |n| > N \end{cases} \quad (4.71)$$

The Fourier transform of this square wave of width $2N + 1$ can be found to be:

$$\begin{aligned}\mathcal{F}[x[n]] &= \sum_{n=-N}^N e^{-jn\omega} = \sum_{n=-N}^0 e^{-jn\omega} + \sum_{n=0}^N e^{-jn\omega} - 1 \\ &= \frac{1 - e^{j(N+1)\omega}}{1 - e^{j\omega}} + \frac{1 - e^{-j(N+1)\omega}}{1 - e^{-j\omega}} - 1 = \frac{e^{j(N+1)\omega} - e^{-jN\omega}}{e^{j\omega} - 1} \frac{e^{-j\omega/2}}{e^{-j\omega/2}} \\ &= \frac{e^{j(2N+1)\omega/2} - e^{-j(2N+1)\omega/2}}{e^{j\omega/2} - e^{-j\omega/2}} = \frac{\sin((2N+1)\omega/2)}{\sin(\omega/2)}\end{aligned}\quad (4.72)$$

- **Triangle wave**

$$x[n] = \begin{cases} 1 - |n|/N & |n| \leq N \\ 0 & |n| > N \end{cases} \quad (4.73)$$

This triangle wave function with width $2N + 1$ can be constructed as the convolution of two square wave functions of width N , scaled down by N , therefore its transform can be found by convolution property to be:

$$\mathcal{F}[x[n]] = \frac{1}{N} \left[\frac{\sin(N\omega/2)}{\sin(\omega/2)} \right]^2 \quad (4.74)$$

- **Sinc function**

$$x[n] = \frac{\sin(2n\pi f_0)}{n\pi} = \frac{\sin(n\omega_0)}{n\pi} \quad (4.75)$$

First consider a square function in frequency:

$$X(\omega) = \begin{cases} 1 & |\omega| \leq \omega_0 \\ 0 & |\omega| > \omega_0 \end{cases} \quad (4.76)$$

The inverse transform of $X(\omega)$ is:

$$\mathcal{F}^{-1}[X(\omega)] = \frac{1}{2\pi} \int_{-\omega_0}^{\omega_0} e^{jm\omega} d\omega = \frac{1}{2\pi} \frac{1}{jn} [e^{jn\omega_0} - e^{-jn\omega_0}] = \frac{\sin(n\omega_0)}{n\pi} \quad (4.77)$$

i.e.,

$$\mathcal{F} \left[\frac{\sin(n\omega_0)}{n\pi} \right] = \begin{cases} 1 & |\omega| < \omega_0 \\ 0 & |\omega| > \omega_0 \end{cases} \quad (4.78)$$

Fig.4.3 shows a set of typical discrete signals and their discrete-time Fourier transforms.

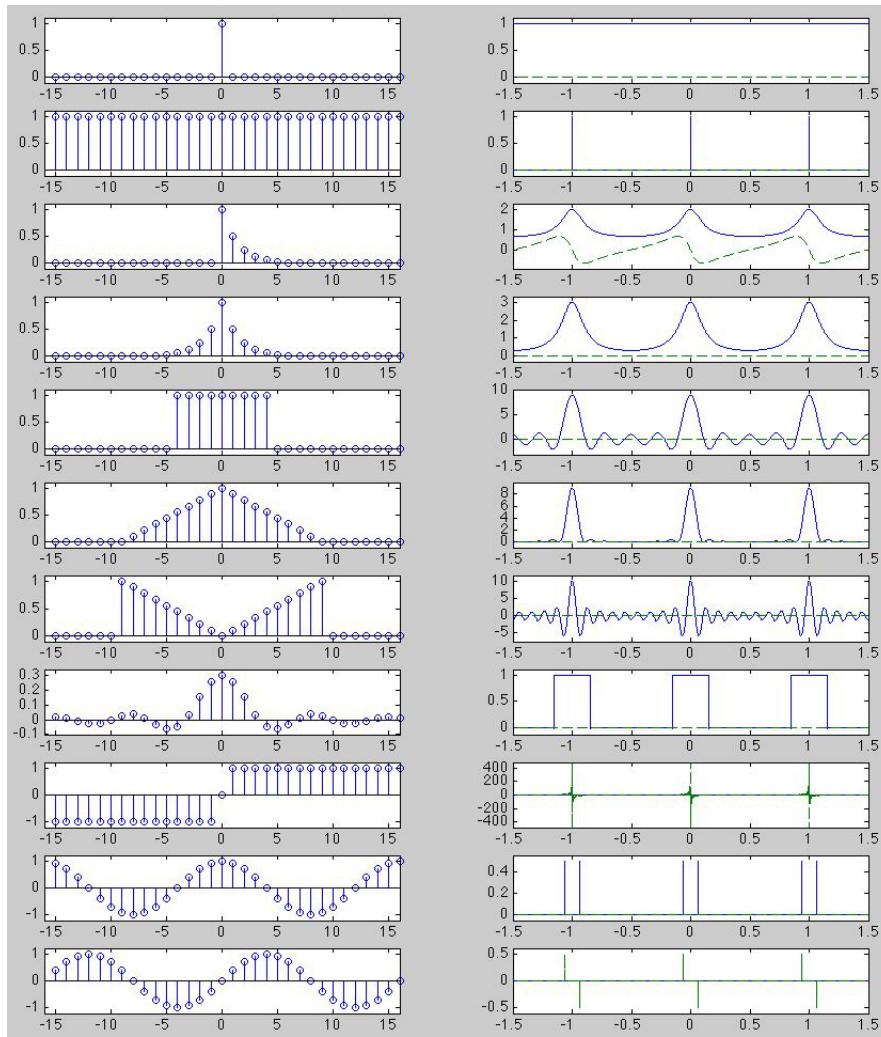


Figure 4.3 Examples of discrete-time Fourier transforms

A set of discrete signals are shown on the left and their DTFT spectra (showing three periods) are shown on the right (real and imaginary parts are shown in solid and dashed lines, respectively).

4.1.4 The Sampling Theorem

An important issue in sampling is the determination of the sampling frequency. On the one hand, it is desirable to minimize the sampling frequency to reduce the data size for lower computational complexity of the digital signal processing and less space and time needed for storage and transmission. On the other hand, the sampling frequency cannot be too low as this may cause certain information contained in the signal to be lost.

We know a time signal $x(t)$ can be perfectly reconstructed from its Fourier spectrum $X(f) = \mathcal{F}[x(t)]$ as its information is equivalently contained in either time or frequency domain (Parseval's identity). However, after sampling by which $x(t)$ is represented by a sequence of samples $x[n]$ ($n = 0, \pm 1, \pm 2, \dots$), can $x(t)$ still be perfectly reconstructed from the spectrum $X_F(f) = \mathcal{F}[x[n]]$?

To answer this question, we consider how the spectrum $X_F(f)$ of the sampled signal $x_s(t)$ is related to the spectrum $X(f)$ of the original signal $x(t)$. Due to the convolution theorem, the spectrum of the sampled signal $x_s(t) = x(t) \text{comb}(t)$ (Eq. 4.1) is the following convolution in frequency domain:

$$\begin{aligned} X_F(f) &= \mathcal{F}[x(t) \text{comb}(t)] = X(f) * \text{Comb}(f) = X(f) * F \sum_{k=-\infty}^{\infty} \delta(f - kF) \\ &= \int_{-\infty}^{\infty} X(f - f') F \sum_{k=-\infty}^{\infty} \delta(f' - kF) df' = F \sum_{k=-\infty}^{\infty} X(f - kF) \end{aligned} \quad (4.79)$$

where $\text{Comb}(f) = F \sum_{k=-\infty}^{\infty} \delta(f - kF)$ is the spectrum of the comb function (Eq. 3.163). We see that the spectrum $X_F(f)$ of the sampled signal is a superposition of infinitely many shifted and scaled (both by F) replicas of the spectrum $X(f)$ of $x(t)$. Obviously if $X(f)$ can be recovered from $X_F(f)$, then $x(t)$ can be reconstructed from $X(f)$.

Consider the following two cases, also illustrated in Fig.4.4, where the maximum frequency component contained in the signal $x(t)$ is f_{max} , i.e., $X(f) = 0$ for any $|f| > f_{max}$, such a signal is said to be *band-limited*.

- If $F/2 > f_{max}$, the neighboring replicas in $X_F(f)$ are separated (second plot) and the original spectrum $X(f)$ (first plot) can be perfectly recovered by a filtering process:

$$X(f) = H_{lp}X_F(f) \quad (4.80)$$

where $H_{lp}(f)$ is an ideal low-pass filter defined as

$$H_{lp}(f) = \begin{cases} 1/F & |f| < f_c = F/2 \\ 0 & \text{otherwise} \end{cases} \quad (4.81)$$

This filter scales all frequencies lower than the cut-off frequency $F/2$ by a factor $1/F = t_0$ but suppresses to zero any frequency higher than $f_c = F/2$.

- If $F/2 < f_{max}$, then *aliasing* or *folding* (to be discussed later) will occur as the replicas in $X_F(f)$ overlap with each other. As they are no longer separable,

it is impossible to recover $X(f)$, as the output of the ideal filter (last plot) is distorted due to the overlapping replicas in $X_F(f)$ (third plot). For example, the highest frequency f_{max} in the signal now appears as a lower frequency $F - f_{max}$, as if it is folded around $F/2$.

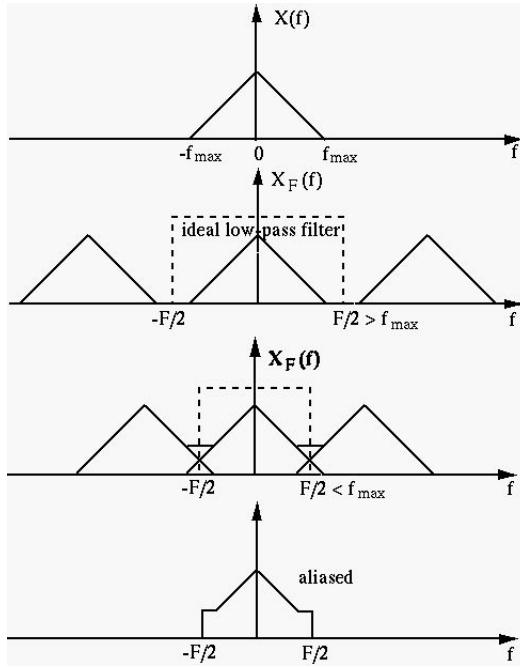


Figure 4.4 Reconstruction of time signal in frequency domain

The above result leads to the well known *sampling theorem*, also called the *Nyquist-Shannon theorem*:

Theorem 4.1. *A signal can be completely reconstructed from its samples taken at a sampling frequency F , if it contains no frequencies higher than $F/2$, referred to as the Nyquist frequency:*

$$f_{max} < f_{Nyquist} = F/2, \quad i.e. \quad F > 2f_{max} \quad (4.82)$$

This equation is referred to as the Nyquist condition for perfect signal reconstruction.

Now we can answer the original question regarding the proper sampling frequency. The lowest sampling frequency F at which the signal can be sampled without losing any information must be higher than twice the maximum frequency of the signal, i.e., $F > 2f_{max}$, otherwise aliasing (or folding) occurs and the original signal cannot be perfectly reconstructed. In practice, however, it is often the case that the signal to be sampled contains frequency components

higher than the Nyquist frequency. To avoid aliasing in such cases, an anti-aliasing low-pass filtering can be carried out to remove all frequencies higher than the Nyquist frequency before sampling. However, certain signal information contained in the filtered out frequency components is lost in this process.

To fully understand the sampling theorem we consider the following examples which serve to illustrate the various effects of the sampling process, when the Nyquist condition is either satisfied or dissatisfied.

Example 4.4: The sampling of a sinusoidal signal $x(t) = \sin(2\pi f_0 t)$ with a sampling rate of $F = 4$ samples per second (sampling period $t_0 = 1/F = 1/4$). This process can also be modeled by the observation of an object rotating counter clock-wise at f_0 cycles per second when illuminated by a strobe light at a fixed rate of $F = 4$ flashes per second, or a wagon wheel in a movie with F frames per second (e.g., $F = 24$ frames per second), as illustrated in Fig.4.5.

We consider the following five cases in which the signal frequency f_0 takes a set of different values. The sampling process can be represented in time domain by the samples of the signal:

$$x[n] = x(t)|_{t=n t_0} = x(nt_0) = x(n/F) = x(n/4) \quad (4.83)$$

- $f_0 = 1 < F/2 = 2$ Hz:

$$x[n] = x(n/4) = \frac{1}{2j}[e^{j2n\pi/4} - e^{-j2n\pi/4}] = \sin(2n\pi/4) \quad (4.84)$$

The two frequency components $f = \pm 1$ Hz are both inside the period $-2 < f < 2$. However, note that as $X(f \pm 4) = X(f)$ is periodic, these two frequency components also appear at $f = \pm 1 + 4k$ for any integer k . In our model the object is rotating at a rate of $f_0 = 1$ cycles per second or 90° per flash counter clockwise, as shown in the first row of Fig.4.5.

- $f_0 = 2 = F/2 = 2$ Hz:

$$x[n] = x(n/4) = \frac{1}{2j}[e^{j2n\pi2/4} - e^{-j2n\pi2/4}] = \frac{1}{2j}[e^{jn\pi} - e^{-jn\pi}] = 0 \quad (4.85)$$

The signal is sampled two times per period and both samples happen to be zero in this case, as if the samples were taken from a zero signal $x(t) = 0$. In our model, the object is rotating at a rate of 180° per flash, when the vertical displacement of the object happen to be zero, as if it is not rotating, as shown in the second row of Fig.4.5.

- $f_0 = 3 > F/2 = 2$ Hz:

$$\begin{aligned} x[n] &= x(n/4) = \frac{1}{2j}[e^{j2n\pi3/4} - e^{-j2n\pi3/4}] \\ &= \frac{1}{2j}[e^{-j2m\pi/4} - e^{j2m\pi1/4}] = -\sin(2m\pi/4) \end{aligned} \quad (4.86)$$

As the signal is under-sampled, its samples are identical to those obtained from a different signal $-\sin(2\pi t) = \sin(-2\pi t)$ at a frequency $f_0 = -1$ Hz. In frequency domain, the two frequency components at $f = \pm 3$ Hz are both outside the central period $-2 < f < 2$, but their replicas $f = 3 - 4 = -1$ and $f = -3 + 4 = 1$ appear inside the central period with opposite polarity. This effect is called *folding*. In the model, the object is rotating at a rate of 270° per flash but it appears to be rotating at a lower rate of 90° per flash in the opposite (clockwise) direction, as shown in the third row of Fig.4.5.

- $f_0 = 4 = F$ Hz:

$$x[n] = x(n/4) = \frac{1}{2j}[e^{j2n\pi 4/4} - e^{-j2n\pi 4/4}] = \frac{1}{2j}[e^{j2n\pi} - e^{-j2n\pi}] = 0 \quad (4.87)$$

The signal is sampled once per period, the samples are necessarily constant, which are all zero in this case. In frequency domain, the replicas of $f_0 = \pm 4$ both appear at the origin at $f = 0$ Hz. In the model, the rotating object stays in the same position when illuminated, and its vertical displacement is always zero, i.e., it appears to be standing still, as shown in the 4th row of Fig.4.5.

- $f_0 = 5 > F/2 = 2$ Hz,

$$\begin{aligned} x[n] = x(n/4) &= \frac{1}{2j}[e^{j2n\pi 5/4} - e^{-j2n\pi 5/4}] \\ &= \frac{1}{2j}[e^{j2n\pi/4} - e^{-j2n\pi/4}] = \sin(2n\pi/4) \end{aligned} \quad (4.88)$$

The samples are identical to those taken from a different signal $\sin(2\pi t)$ with frequency $f_0 = 1$ Hz. In frequency domain, the two frequency components $f = \pm 5$ Hz are both outside the central period $-2 < f < 2$, but their replicas of appear inside the central period at $f = 5 - 4 = 1$ and $f = -5 + 4 = -1$ Hz with the same polarity. This effect is called *aliasing*. In the model, the object rotating 450° per flash appears to rotate 90° per flash in the same counter clockwise direction, as shown in the last row of Fig.4.5.

Note that in all these cases, the observed frequency f is always the replicas of the lowest frequency inside the central period $-F/2 < f < F/2$ in the spectrum, which is the same as the true signal frequency $f = f_0$ only when $f_0 < F/2$. Otherwise, aliasing or folding occurs and the apparent frequency is always lower than the true frequency. In the model of a rotating object, even if we know the object could have rotated an angle of $\phi \pm 2k\pi$ per flash, the perceived frequency by our visual system is always either ϕ or $\phi - 2\pi = -(2\pi - \phi)$ per flash, depending on which has a lower absolute value. In the latter case, as the polarity is changed, not only is the frequency appears to be lower, but also the direction is reversed.

In the marginal case where the signal frequency $f_0 = F/2$ is equal to the Nyquist frequency, the sampled signal may appear zero, as shown above, but this is not necessarily the case in general. Consider the same signal as above

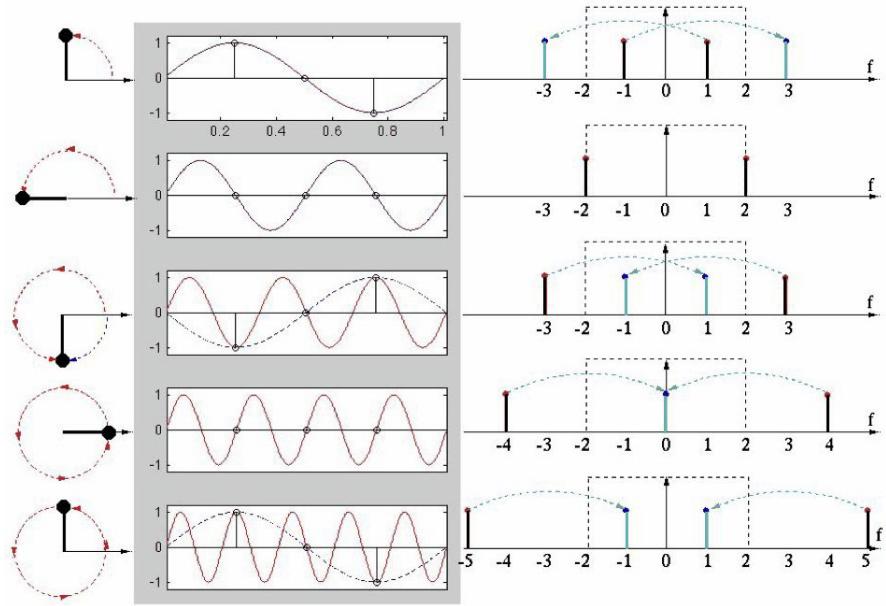


Figure 4.5 Aliasing and folding in time and frequency domains

Model of rotating object illuminated by a strobe light (left, only the first flash is shown), sampling of the vertical displacement (middle), and the aliased frequency (perceived rotation) (right)

with a phase shift $x(t) = \sin(2\pi f_0 t + \phi)$. When it is sampled at exactly the rate of $F = 2f_0$, the values of its samples depend on the phase ϕ :

$$x[n] = x(n/F) = \sin(2n\pi f_0/F + \phi) = \sin(n\pi + \phi) \quad (4.89)$$

This is indeed zero when $\phi = 0$ as shown before. However, when $\phi \neq 0$, we have:

$$x[n] = \sin(n\pi + \phi) = \begin{cases} \sin \phi & n \text{ is even} \\ -\sin \phi & n \text{ is odd} \end{cases} \quad (4.90)$$

In other words, in the marginal case when $f_0 = F/2$, so long as $\phi \neq 0$ and $\phi \neq \pi$, the sign of $x[n]$ alternates and the frequency f_0 of $x(t)$ can be accurately represented, but its amplitude is scaled by $\sin \phi$, and its phase ϕ is not reflected, as shown in Fig.4.6. In particular when $\phi = \pi/2$, $x[n] = 1$ if n is even and $x[n] = -1$ if n is odd, i.e., the amplitude of the signal is accurately represented by its samples.

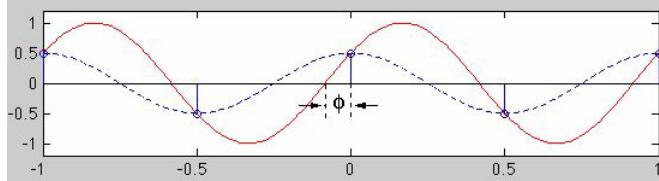


Figure 4.6 Marginal sampling: signal frequency equals Nyquist frequency $f_0 = F/2$

Example 4.5: This example further illustrates the effect of sampling and aliasing/folding. Consider a continuous signal

$$x(t) = \cos(2\pi ft + \phi) = \frac{1}{2}[e^{j(2\pi ft+\phi)} + e^{-j(2\pi ft+\phi)}] = c_1 e^{j2\pi ft} + c_{-1} e^{-j2\pi ft} \quad (4.91)$$

where $c_1 = e^{j\phi}/2$ and $c_{-1} = e^{-j\phi}/2$ are respectively the two non-zero coefficients for the frequency components $e^{j2\pi ft}$ and $e^{-j2\pi ft}$. When this signal is sampled at a rate of $F = 1/t_0$, it becomes a discrete signal:

$$\begin{aligned} x[n] &= \cos(2\pi fnt_0 + \phi) = \cos(2\pi fn/F + \phi) = \frac{e^{j\phi}}{2} e^{j2\pi fn/F} + \frac{e^{-j\phi}}{2} e^{-j2\pi fn/F} \\ &= c_1 e^{-j2\pi fn/F} + c_{-1} e^{-j2\pi fn/F} \end{aligned}$$

Fig.4.7 shows the signal being sampled at $F = 6$ samples per second, while its frequency f increases from 1 to 12 Hz with 1 Hz increment. In time domain (left), the original signal (solid line) and the reconstructed one (dashed line) are both plotted. In frequency domain (right), the spectrum of the sampled version of the signal is periodic with period $F = 6$, and three periods are shown including two neighboring periods on both the positive and negative sides as well as the middle one. However, note that the signal reconstruction by inverse Fourier transform, and also by human eye, is only based on the information in the middle period.

- $f = 1 < F/2 = 3$, the two non-zero frequency components $e^{\pm j2\pi ft}$ are both inside the middle period $-3 < f < 3$ Hz of the spectrum, based on which the signal can be perfectly reconstructed.
- $f = 2 < F/2 = 3$, frequency components $e^{\pm j2\pi ft}$ move outward to a higher frequency of ± 2 Hz, which are still inside the middle period, no aliasing or folding occurs.
- $f = 3 = F/3$, the signal is marginally aliased. Depending on the relative phase difference between the signal and the sampling function, the signal may be distorted to different extent. In the worst case, when the two samples happen to be taken at the zero-crossings of the signal ($\phi = 0$ or $\phi = \pi$), they are both zero and the signal $x(t) = \cos(2\pi 4f + \phi)$ is aliased to a zero signal $x(t) = 0$.
- $f = 4 > F/2 = 3$, the two coefficients $e^{\pm j2\pi ft}$ are outside the middle period, but the replica at $f = 4$ Hz moves from the right into the middle period to appear as $4 - 6 = -2$ Hz, and the replica at $f = -4$ Hz moves from left into the middle period to appear as $-4 + 6 = 2$ Hz. The reconstructed signal

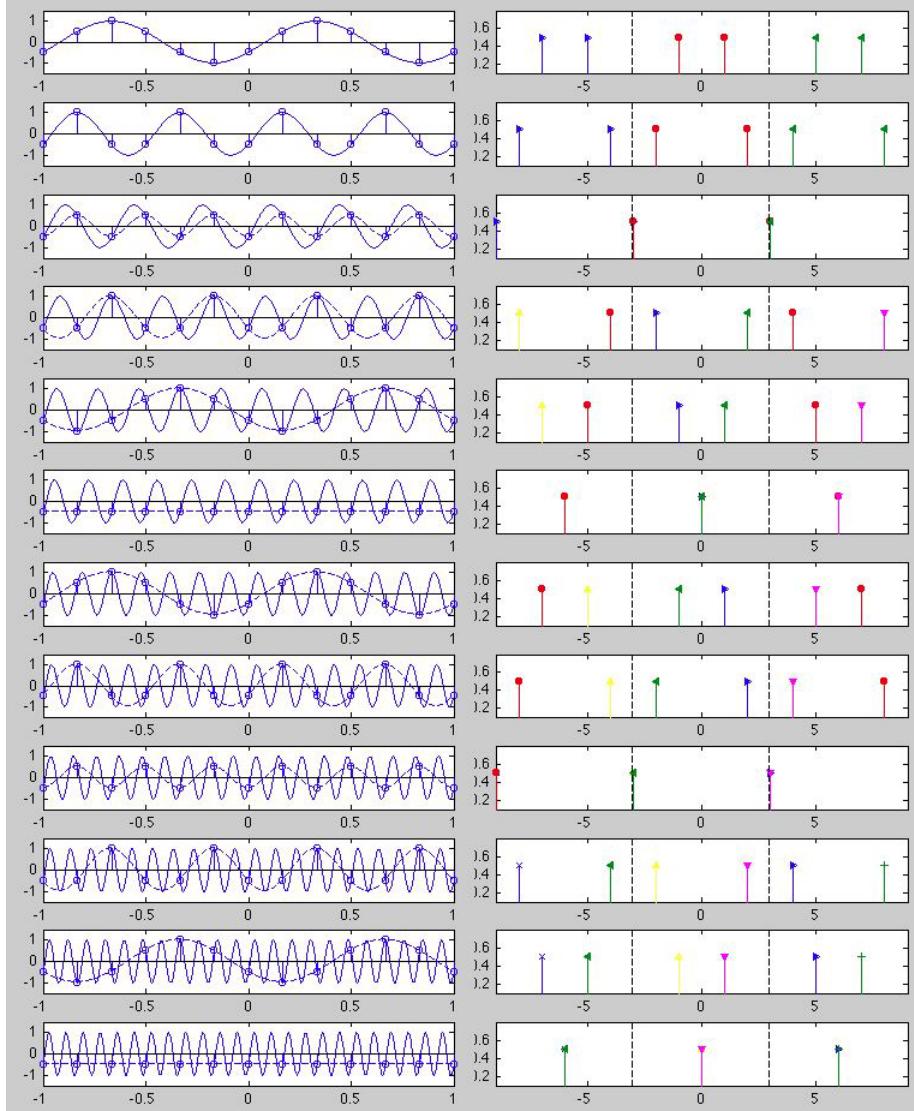


Figure 4.7 Aliasing in time and frequency domains

based on these folded frequency components is $\cos(2\pi 2t - \phi)$, different from the original signal $x(t) = \cos(2\pi 4t + \phi)$.

- $f = 5 >= F/2 = 3$, similar folding occurs and the reconstructed signal based on the folded frequency at $f = \pm 1 \text{ Hz}$ is $\cos(2\pi t - \phi)$.
- $f = 6 = F$, one sample is taken per period, the aliased frequency is zero, and the reconstructed signal is $\cos(\phi)$
- $f = 7 = F + 1$, the two coefficients for $f = \pm 7$ are out of the middle period, but the replica at $f = -7 \text{ Hz}$ is aliased to appear inside the middle period as $-7 + 6 = -1 \text{ Hz}$, and the replica of $f = 7$ is aliased to appear inside the

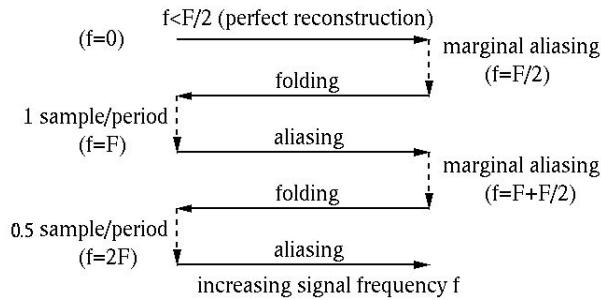


Figure 4.8 Aliasing-folding cycle as signal frequency increases

middle period as $7 - 6 = 1$ Hz. Based on these aliased frequency components, the reconstructed signal is $\cos(2\pi ft + \phi)$, which appears to be the same as the non-aliased cases when $f = 1$.

- $f = 8 = F + 2$, similar aliasing occurs and the reconstructed signal is $\cos(2\pi 2t + \phi)$, which appears the same as the non-aliased case of $f = 2$.
- $f = 9 = F + F/2$, marginal aliasing occurs same as the case of $f = 3$.
- When $f = 10 = F + 4$ and $f = 11 = F + 5$, folding occurs similar to the cases when $f = 4$ and $f = 5$, respectively.
- $f = 12 = 2F$, same as in the case of $f = 6 = F$, one sample is taken per period and the aliased frequency is zero.

We see that only when $f < F/2$ (the first two cases) can the signal be perfectly reconstructed. After that the cycle of folding and aliasing will repeat as the signal frequency f increases continuously. This pattern is illustrated in Fig.4.8.

Example 4.6: Consider the following three continuous signals that are first sampled at a sampling rate of $F = 10$ samples/second, and then reconstructed based on the resulting samples:

1. $x_1(t) = 2 \cos(2\pi 7t) + \cos(2\pi 2t)$
2. $x_2(t) = 2 \cos(2\pi 8t) + \cos(2\pi 2t)$
3. $x_3(t) = 2 \cos(2\pi 8t) - 2 \cos(2\pi 2t)$

As the sampling rate is not higher than twice of the highest frequency component in the signal, aliasing/folding happens in all three cases, causing various forms of signal distortion, as shown in Fig.4.9 that compares the original signals (solid curves) with the reconstructions (dashed curves). It can be seen that the reconstructed signal is distorted in the first case, it becomes a single sinusoid in the second case, and it becomes zero due to the fact that the original signal happens to be sampled at zero-crossings.

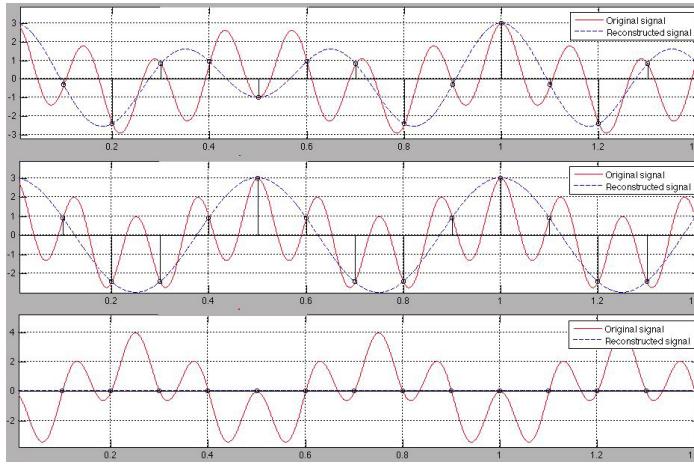


Figure 4.9 Different cases of aliasing/folding

The mathematical derivation of these results is left for the reader as a homework problem. The Matlab function `guidemo_sampling` used for creating these plots is provided.

The sampling theorem is derived based on the assumption that the signal spectrum occupies the entire frequency range $|f| < f_{max}$, and the signal can be perfectly reconstructed by an ideal low-pass filter if $F > 2f_{max}$. However, if the energy of the signal is concentrated within a certain frequency band $f_{min} < |f| < f_{max}$, it is possible to reconstruct the signal by a band-pass filter if $F < 2f_{max}$. As shown in Fig.4.10, the signal within the frequency range $f_{min} < |f| < f_{max} = 3$ (Hz or kHz, for example) can be perfectly reconstructed if $F > 2f_{max} = 6$ (top) according to the sampling theorem, but it can still be reconstructed if $F = 3.5 < 2f_{max} = 6$ (bottom), so long as in the periodic spectrum after sampling the original spectrum (dark gray) is not distorted by its replica on either right or left, it is therefore possible to use an ideal band-pass filter to recover the original spectrum by suppressing all of its replicas.

4.1.5 Reconstruction by Interpolation

Once a continuous signal is sampled in the process of A/D conversion, it becomes a discrete signal that can be digitally processed/filtered by some digital signal processing (DSP) system (or just a computer). Often the processed signal needs to be converted back into analog form by a digital-to-analog converter (DAC, D/A), which reconstructs the signal from its samples.

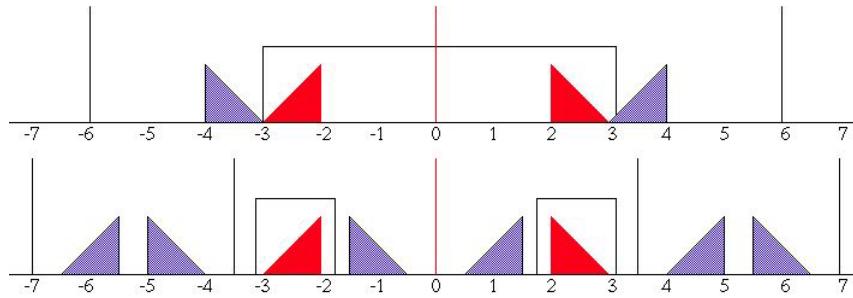


Figure 4.10 Low-pass and band-pass filtering for signal reconstruction

When the signal frequency is limited to ($f_{min} = 2 < |f| < f_{max} = 3$), it can be sampled with $F \geq 2f_{max} = 6$ and reconstructed by an ideal low-pass filter (top), or sampled with $F = 3.5 < 6$ and reconstructed by an ideal band-pass filter (bottom).

As shown above, the reconstruction of a continuous signal $x(t)$ from its sampled version $x_s(t)$ is a low-pass (LP) filtering process in frequency domain:

$$\hat{X}(f) = H_{lp}(f)X_F(f) \quad (4.92)$$

where $H_{lp}(f)$ is an ideal low-pass (LP) filter defined in Eq.4.81. If the Nyquist condition is satisfied, the output of the filter is $\hat{X}(f) = X(f)$, from which the signal $x(t) = \mathcal{F}^{-1}[X(f)]$ can be perfectly reconstructed. In practice, as the ideal LP filter is hard to implement, sometimes a non-ideal LP filter could also be used to approximately reconstruct the signal. On the other hand, if the Nyquist condition is not satisfied, any signal component with frequency $f > F/2$ is outside the central period but one of its aliased or folded version will appear inside the central period, based on which the reconstructed signal is a distorted version of the original signal.

In time domain, the reconstruction of a signal $x(t)$ from its sampled version $x_s(t)$ is an interpolation process by which the gaps between two consecutive samples is filled. The interpolation can be considered as a convolution of the impulses in $x_s(t)$ with a certain function $h(t)$:

$$\begin{aligned} \hat{x}(t) &= h(t) * x_s(t) = h(t) * \sum_{n=-\infty}^{\infty} x[n]\delta(t - nt_0) \\ &= \sum_{n=-\infty}^{\infty} x[n]h(t) * \delta(t - nt_0) = \sum_{n=-\infty}^{\infty} x[n]h(t - nt_0) \end{aligned} \quad (4.93)$$

We consider the following reconstructions based on three different interpretation functions $h_0(t)$, $h_1(t)$ and $h_{lp}(t)$. The time domain interpolation based on the these functions and the corresponding low-pass filtering are illustrated in Fig.4.11.

- **Zero-order hold**

The impulse response of a *zero-order hold* filter is:

$$h_0(t) = \begin{cases} 1 & 0 \leq t < t_0 = 1/F \\ 0 & \text{else} \end{cases} \quad (4.94)$$

This is the rectangular function discussed before (Eq. 3.151) with $\tau = 2t_0$, but shifted by $t_0/2$. Based on $h_0(t)$, a continuous signal $\hat{x}_0(t)$ can be generated by

$$\hat{x}_0(t) = h_0(t) * x_s(t) = \sum_{n=-\infty}^{\infty} x[n]h_0(t - nt_0) \quad (4.95)$$

This is a series of square impulses with their heights modulated by $x[n]$. The interpolation corresponds a low-pass filtering in frequency domain:

$$H_0(f) = \mathcal{F}[h_0(t)] = \frac{1}{\pi f} \sin(\pi f t_0) e^{-j2\pi f t_0/2} \quad (4.96)$$

(Eq. 3.153 with an exponential factor corresponding to the time shift of $t_0/2$)

- **First-order hold**

The impulse response of a *first-order hold* filter is:

$$h_1(t) = \begin{cases} 1 - |t|/t_0 & |t| < t_0 \\ 0 & \text{otherwise} \end{cases} \quad (4.97)$$

which is the triangle function previously discussed (Eq.3.156) with $\tau = t_0$. A continuous signal $\hat{x}_1(t)$ can be generated by:

$$\hat{x}_1(t) = h_1(t) * x_s(t) = \sum_{n=-\infty}^{\infty} x[n]h_1(t - nt_0) \quad (4.98)$$

which is the linear interpolation of the sample train $x[n]$ (a straight line segment connecting every two consecutive samples). This interpolation corresponds a low-pass filtering in frequency domain by the following (Eq. 3.157)

$$H_1(f) = \mathcal{F}[h_1(t)] = \frac{1}{(\pi f)^2 t_0} \sin^2(\pi f t_0) = t_0 \operatorname{sinc}^2(ft_0) \quad (4.99)$$

- **Ideal reconstruction**

The reconstructed signals $\hat{x}_0(t)$ and $\hat{x}_1(t)$ are only approximations of the actual signal $x(t)$, as these interpolations correspond to non-ideal low-pass filtering in frequency domain. The interpolation function for a perfect reconstruction is obviously associated with the ideal low-pass filtering method given in Eq.4.81. The impulse response of this filter is (Eq. 3.155):

$$h_2(t) = h_{lp}(t) = \mathcal{F}^{-1}[H_{lp}(f)] = t_0 \frac{\sin(2\pi f_c t)}{\pi t} \quad (4.100)$$

and the low-pass filtering corresponds to the following convolution in time domain:

$$\begin{aligned}\hat{x}_2(t) &= h_2(t) * x_s(t) = t_0 \frac{\sin(2\pi f_c t)}{\pi t} * \sum_{n=-\infty}^{\infty} x[n] \delta(t - nt_0) \\ &= \frac{t_0}{\pi} \sum_{n=-\infty}^{\infty} x[n] \frac{\sin(2\pi f_c(t - nt_0))}{t - nt_0}\end{aligned}\quad (4.101)$$

This signal generated by the ideal LP filter is the perfect reconstruction of the signal $\hat{x}_2(t) = x(t)$ without any distortion.

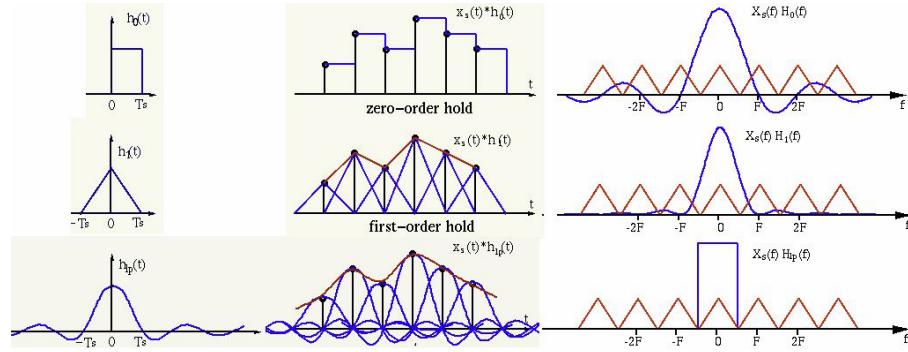


Figure 4.11 0th, 1st and 2nd-order hold reconstructions

The impulse response of the filter (left), the interpolation in time domain (middle), and the corresponding LP filtering in frequency domain (right)

Having considered both signal sampling (A/D conversion) and reconstruction (D/A conversion), we can now put them together with some digital signal processing system to form a pipeline as shown in Fig.4.12. The discrete signal $x[n]$ obtained by sampling is then processed/filtered to become $y[n] = h[n] * x[n]$, based on which a continuous version $y(t)$ can be reconstructed and used. For example, an analog audio signal can be sampled, digitally processed (e.g., low-pass filtered to remove some high frequency noise) and then converted back to analog form to drive the speaker.

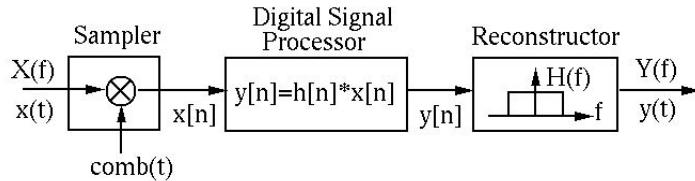


Figure 4.12 Signal sampling, digital processing and reconstruction

The sampling rate of the comb function $comb(t) = \sum_k \delta(t - k/F)$ is F , the cut-off frequency of the ideal low-pass filter $H(f)$ is $F/2$.

4.2 Discrete Fourier Transform

4.2.1 Formulation of DFT

In practice it is impossible to describe a physical signal, typically continuous and non-periodic, as a time function $x(t)$, as the analytical expression of the function is in general not available. In order to process and analyze such a signal in frequency domain as well as in time domain by a digital computer, the signal needs to be digitized in the following two steps:

- First, the signal needs to be truncated so that it has a finite duration from 0 to T , outside which the signal is not defined. However, for certain mathematical convenience we could further assume that the signal repeats itself outside the interval $0 < t < T$, i.e., it is a periodic signal with period T . Correspondingly in frequency domain, the Fourier spectrum of such a periodic signal becomes discrete, composed of a set of impulses weighted by the Fourier expansion coefficients.
- Second, the signal needs to be discretized by sampling with a sampling rate F so that it can be processed by a digital computer. Correspondingly in frequency domain, the spectrum of the signal becomes periodic.

Of course the order of these two steps can be reversed so that the continuous signal is first sampled and then truncated. In either case, when the signal is both finite (periodic) and discrete, its spectrum, also discrete and finite periodic), can be obtained by the *discrete Fourier transform (DFT)*.

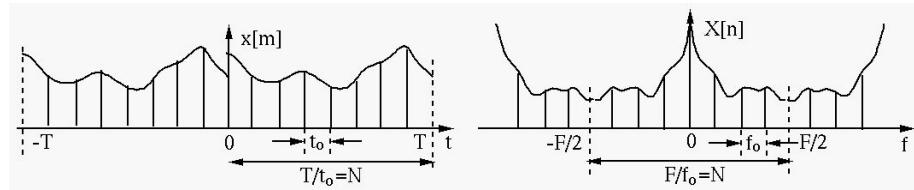


Figure 4.13 From continuous Fourier transform to discrete Fourier transform

To formulate the discrete Fourier transform, we recall the two different forms of the Fourier transform. First, a periodic signal $x_T(t+T) = x_T(t)$ has a discrete Fourier spectrum, an impulse train weighted by the coefficients $X[k]$ of its Fourier expansion (Eq. 3.78). The interval between two neighboring frequency components is the fundamental frequency $f_0 = 1/T$. Second, a discrete signal $x[n]$ obtained by sampling a continuous signal $x(t)$ at a sampling rate F (or a gap of $t_0 = 1/F$ between two consecutive samples) has a periodic spectrum $X_F(f+F) = X_F(f)$ (Eq. 4.3). It is therefore obvious that if a signal is both periodic with period T and discrete with interval t_0 between two consecutive samples, its spectrum will be both discrete with an interval $f_0 = 1/T$ between two frequency components, and periodic with a period of $F = 1/t_0$. In time domain, the number of samples in a period T is $N = T/t_0$, while in frequency

domain, the number of frequency components in a period F is:

$$\frac{F}{f_0} = \frac{1/t_0}{1/T} = \frac{T}{t_0} = N \quad (4.102)$$

In other words, the number of independent variables, or *degrees of freedom (DOF)*, in either time or frequency domain is conserved by the DFT. This fact is also expected from the view point of information conservation of the transform. We also have the following relations that are useful in the future discussion:

$$TF = \frac{T}{t_0} = N, \quad f_0 t_0 = \frac{t_0}{T} = \frac{1}{N} \quad (4.103)$$

Consider a continuous signal already truncated with duration T and assumed to be periodic $x_T(t+T) = x_T(t)$. This signal is further sampled when multiplied by the sampling function $\text{comb}(t)$:

$$x_T(t) \text{comb}(t) = x_T(t) \sum_{n=-\infty}^{\infty} \delta(t - nt_0) = \sum_{n=-\infty}^{\infty} x[n] \delta(t - nt_0) \quad (4.104)$$

where $x[n] = x_T(nt_0)$ is the nth sample of the signal. Note that $x[n]$ is periodic with period N :

$$x[n+N] = x_T((n+N)t_0) = x_T(nt_0 + T) = x_T(nt_0) = x[n] \quad (4.105)$$

The Fourier expansion coefficient of this sampled version of the periodic and sampled signal can be found as:

$$\begin{aligned} X[k] &= \frac{1}{T} \int_0^T \left[\sum_{n=-\infty}^{\infty} x[n] \delta(t - nt_0) \right] e^{-j2\pi k f_0 t} dt \\ &= \frac{1}{T} \sum_{n=0}^{N-1} x[n] \int_0^T \delta(t - nt_0) e^{-j2\pi k f_0 t} dt = \frac{1}{T} \sum_{n=0}^{N-1} x[n] e^{-j2\pi k f_0 n t_0} \\ &= \frac{1}{T} \sum_{n=0}^{N-1} x[n] e^{-j2\pi n k / N}, \quad (k = 0, 1, \dots, N-1) \end{aligned} \quad (4.106)$$

The number of terms in the summation is reduced from infinity to N for those inside the integral range from 0 to T , as all terms outside the range make no contribution to the integral. Note that $X[k+N] = X[k]$ is also periodic with period N :

$$X[k+N] = \frac{1}{T} \sum_{n=0}^{N-1} x[n] e^{-j2\pi(k+N)n/N} = \frac{1}{T} \sum_{k=0}^{N-1} x[k] e^{-j2\pi n k / N} e^{-j2n\pi} = X[k] \quad (4.107)$$

The inverse transform can be obtained by multiplying both sides of Eq.4.106 by $e^{j2\pi\nu k/N}/F$, and taking summation with respect to n from 0 to $N - 1$:

$$\begin{aligned} \frac{1}{F} \sum_{k=0}^{N-1} X[k] e^{j2\pi\nu k/N} &= \frac{1}{F} \sum_{k=0}^{N-1} \left[\frac{1}{T} \sum_{n=0}^{N-1} x[n] e^{-j2\pi n k/N} \right] e^{j2\pi\nu k/N} \\ &= \sum_{n=0}^{N-1} x[n] \frac{1}{N} \sum_{k=0}^{N-1} e^{j2\pi n [\nu - k]/N} = \sum_{n=0}^{N-1} x[n] \delta[\nu - n] = x[\nu] \end{aligned} \quad (4.108)$$

Here we have used Eq.1.40. Now we put Eqs. 4.106 and 4.108 together to form the DFT pair:

$$\begin{aligned} X[k] = \mathcal{F}[x[n]] &= \frac{1}{T} \sum_{n=0}^{N-1} x[n] e^{-j2\pi n k/N}, \quad (k = 0, 1, \dots, N - 1) \\ x[n] = \mathcal{F}^{-1}[X[k]] &= \frac{1}{F} \sum_{k=0}^{N-1} X[k] e^{j2\pi n k/N}, \quad (n = 0, 1, \dots, N - 1) \end{aligned} \quad (4.109)$$

The first equation is the forward DFT while the second one the inverse DFT. As both $x[n]$ and $X[k]$ are periodic with period N , the summation in either the forward or inverse transform can be over any consecutive N points, such as from $-N/2$ to $N/2 - 1$.

We can modify the scaling factors $1/T$ and $1/F$ for the forward and inverse transforms in Eq.4.109 by redistributing the total scaling factor of $1/FT = 1/N$ differently between the two transforms. For example, we can scale either of the two by $1/N$, or alternatively, we can also evenly distribute it on both sides:

$$\begin{aligned} x[n] = \mathcal{F}^{-1}[X[k]] &= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X[k] e^{j2\pi n k/N}, \quad (n = 0, 1, \dots, N - 1) \\ X[k] = \mathcal{F}[x[n]] &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x[n] e^{-j2\pi n k/N}, \quad (k = 0, 1, \dots, N - 1) \end{aligned} \quad (4.110)$$

The advantage of this notation is that now the signal can be represented as a vector $\mathbf{x} = [x[0], \dots, x[N - 1]]^T$ in an N -D vector space \mathbb{C}^N spanned by a set of N orthonormal basis vectors:

$$\mathbf{w}_k = \frac{1}{\sqrt{N}} [e^{j2\pi 0k/N}, \dots, e^{j2\pi (N-1)k/N}]^T, \quad (k = 0, \dots, N - 1) \quad (4.111)$$

satisfying (Eq.1.40)

$$\langle \mathbf{w}_k, \mathbf{w}_l \rangle = \frac{1}{N} \sum_{n=0}^{N-1} e^{j2\pi n(k-l)/N} = \delta[k - l] \quad (4.112)$$

Under this basis, the given signal vector \mathbf{x} can be represented as:

$$\mathbf{x} = \sum_{k=0}^{N-1} X[k] \mathbf{w}_k, \quad \text{where} \quad X[k] = \langle \mathbf{x}, \mathbf{w}_k \rangle = \mathbf{x}^T \overline{\mathbf{w}}_k \quad (4.113)$$

The first equation above is the vector form of first equation in Eq.4.110 for the inverse DFT, the second equation is the same as the second equation in Eq.4.110 for the forward DFT. In this case, Parseval's identity holds, i.e., $\|\mathbf{x}\|^2 = \|\mathbf{X}\|^2$.¹

The discrete spectrum $X[k]$ of the samples $x[n]$ of a signal $x(t)$ is obviously related to, but certainly not equal to, the spectrum $X(f) = \mathcal{F}[x(t)]$, as the signal has been significantly modified by the truncation and sampling process before the DFT can be carried out. First, due to the truncation and the assumed periodicity, the signal may no longer be continuous and smooth. Discontinuity will occur at the end point between two consecutive periods if $x(0) \neq x(T)$, as shown on the left of Fig.4.13. Second, due to the sampling process, aliasing or folding may occur if the Nyquist condition is not satisfied. Consequently The spectrum may be contaminated by various artifacts, most likely some faulty high frequency components corresponding to the discontinuities, together with some faulty low frequencies due to aliasing or folding. Therefore special attention needs to be paid to the truncation and sampling process in order to minimize such artifacts. For example, certain windowing method can be used to smooth the truncated signal, and some anti-aliasing low-pass filtering can be used to reduce the high frequency components before sampling to avoid aliasing. Only then can the DFT generate meaningful data representative of the actual signal of interest.

Example 4.7: Consider a discrete sinusoid of $N = 5$ samples with frequency $f = 1/N = 1/5$ (one cycle per $N = 5$ points):

$$x[n] = \cos\left(n \frac{2\pi}{5}\right) = \frac{1}{2}[e^{j2\pi n/5} + e^{-j2\pi n/5}], \quad (n = 0, \dots, N-1 = 4) \quad (4.114)$$

Comparing this expression with the DFT expansion:

$$x[n] = \sum_{k=0}^4 X[k] e^{j2\pi nk/5} \quad (4.115)$$

¹ In Matlab a scaling factor of $1/N$ is included in the inverse DFT function IFFT while the forward DFT function FFT has a scaling factor 1. For Parseval's identity to hold, these functions need to be rescaled: $X = fft(x)/sqrt(length(x))$ and $x = ifft(X) * sqrt(length(X))$.

we see that $X[1] = 1/2$ and $X[4] = X[-1] = 1/2$. Alternatively, following the DFT definition we can also get the nth Fourier coefficient as:

$$\begin{aligned} X[k] &= \frac{1}{N} \sum_{n=0}^{N-1} x[n] e^{-j2\pi nk/N} = \frac{1}{10} \sum_{n=0}^4 [e^{-j2\pi n(n-1)/5} + e^{-j2\pi n(n+1)/5}] \\ &= \frac{1}{2} [\delta[n+1] + \delta[n-1]] \end{aligned} \quad (4.116)$$

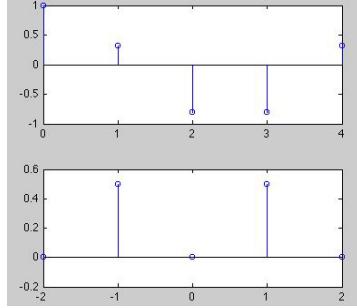


Figure 4.14 Discrete cosine and its DFT spectrum

Example 4.8: Consider a symmetric square wave with a period of N and width $2M < N$:

$$x[n] = \begin{cases} 1 & |n| \leq M \\ 0 & M < |n| \leq N/2 \end{cases} \quad (4.117)$$

For convenience, we choose the limits of the Fourier transform summation from $-N/2$ to $N/2 - 1$ (instead of from 0 to $N - 1$) and get:

$$X[k] = \sum_{n=-N/2}^{N/2-1} x[n] e^{-j2\pi nk/N} = \sum_{n=-M}^M e^{-j2\pi nk/N} \quad (4.118)$$

Let $n' = n + M$, we have $n = n' - M$ and

$$\begin{aligned} X[k] &= \sum_{n'=0}^{2M} e^{-j2\pi n'k/N} e^{j2\pi Mn/N} = e^{j2\pi Mk/N} \frac{1 - e^{-j2\pi(2M+1)k/N}}{1 - e^{-j2\pi k/N}} \\ &= e^{j2\pi Mk/N} \frac{e^{-j\pi(2M+1)k/N} (e^{j\pi(2M+1)k/N} + e^{-j\pi(2M+1)k/N})}{e^{-j\pi k/N} (e^{j\pi k/N} - e^{-j\pi k/N})} \\ &= \frac{\sin((2M+1)k\pi/N)}{\sin(k\pi/N)} \end{aligned} \quad (4.119)$$

The signal and its DFT spectrum are shown in Fig.4.15 ($N = 64$, $M = 8$).

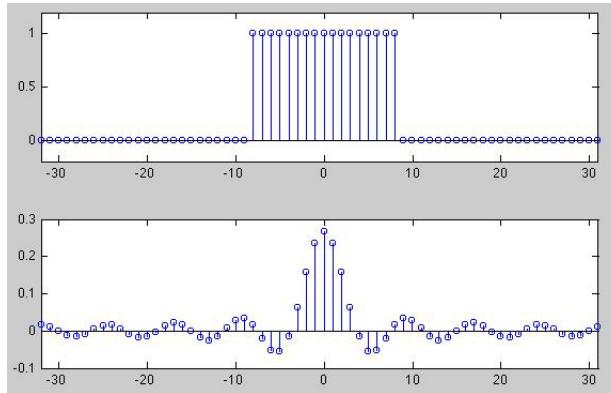


Figure 4.15 Square impulse and its DFT spectrum

4.2.2 Array Representation

Similar to the k th basis function for the Fourier series expansion $\phi_k(t) = e^{j2\pi f_k t} = \cos(2\pi f_k t) + j \sin(2\pi f_k t)$ representing a continuous sinusoid of frequency $f_k = kf_0 = k/T$ (k cycles per period of T), here the N samples $w_k[n] = e^{j2\pi nk/N} = \cos(2\pi nk/N) + j \sin(2\pi nk/N)$ ($n = 0, \dots, N-1$) of the k th basis vector \mathbf{w}_k in Eq.4.111 represent a sinusoid of frequency $f_k = k/N$ (k cycles per period of N samples). However, we also note that kf_0 is the frequency for a continuous sinusoid that grows without limit as k increases, k/N is for the samples of a continuous sinusoid that does not grow without limit. For example, when $k = N-1$, $k/N = (N-1)/N$ actually represents a frequency of 1 (instead of $N-1$) cycles per period of N samples, as any frequency higher than $N/2$ is under sampled and appears as a frequency lower than $N/2$ cycles per N samples caused by aliasing.

Shown in Fig.4.16 (1st and 2nd columns) are the first $N = 8$ basis functions $\phi_k(t) = e^{j2\pi kt/T}$ ($k = 0, \dots, 7$) for the Fourier series expansion (continuous curves), together with the discrete samples $w_k[n] = e^{j2\pi nk/N}$ for each of the basis vectors of the corresponding 8-point DFT (the circles). We see that while the frequency $kf_0 = k/T$ for the continuous sinusoid $e^{j2\pi kt/T}$ increases with k , the frequency of $e^{j2\pi nk/N}$ does not increase with k monotonically. Actually its frequency k/N is proportional to k only when $k < N/2 = 4$, but it becomes $(N-k)/N$ when $k > N/2 = 4$, due obviously to aliasing. We also note that the 0th basis vector \mathbf{w}_0 represents the DC component of the signal, and the 4th ($N/2$) basis vector $\mathbf{w}_{N/2}$ is the highest representable frequency of $N/2 = 4$ cycles per period T . The 3rd and 4th columns of Fig.4.16 are for an example to be considered later.

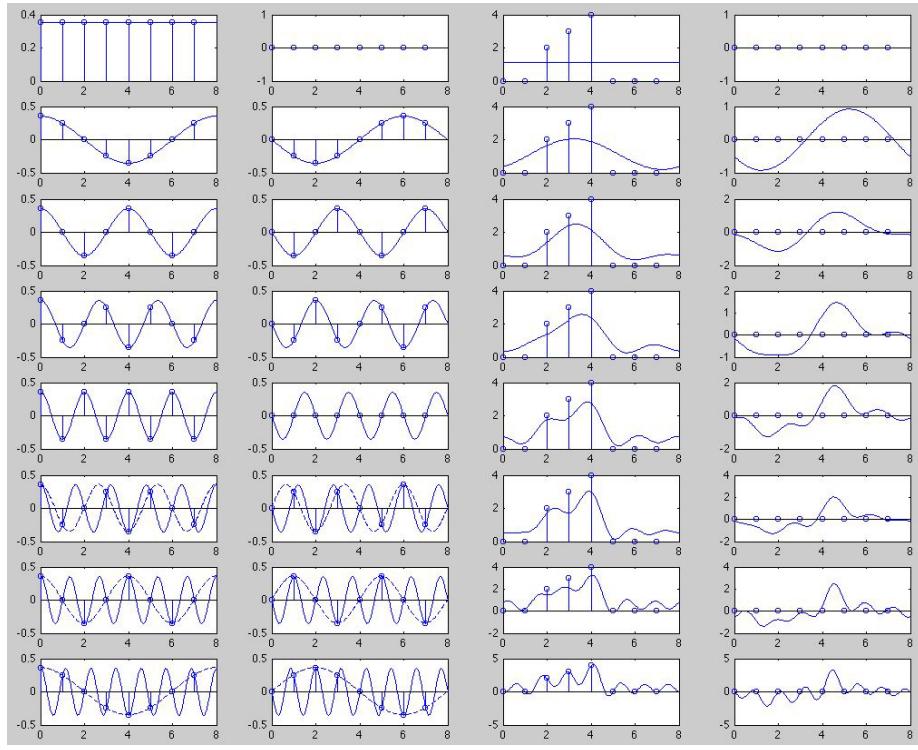


Figure 4.16 Basis functions and vectors of 8-point DFT

The $N=8$ real and imaginary parts of the DFT basis vectors and the associated basis functions are shown respectively in the 1st and 2nd columns; the real and imaginary parts of the reconstructions by inverse DFT of a discrete signal (Example 4.9) with progressively more components are shown respectively in the 3rd and 4th columns.

In general, the N by N matrix of an N -point DFT is composed of the N basis vectors \mathbf{w}_k ($k = 0, \dots, N - 1$) as the N column vectors:

$$\mathbf{W} = [\mathbf{w}_0, \dots, \mathbf{w}_{N-1}] = \frac{1}{\sqrt{N}} \begin{bmatrix} e^{j2\pi 00/N} & e^{j2\pi 01/N} & \dots & e^{j2\pi 0(N-1)/N} \\ e^{j2\pi 10/N} & e^{j2\pi 11/N} & \dots & e^{j2\pi 1(N-1)/N} \\ \vdots & \vdots & \ddots & \vdots \\ e^{j2\pi(N-1)0/N} & e^{j2\pi(N-1)1/N} & \dots & e^{j2\pi(N-1)(N-1)/N} \end{bmatrix} \quad (4.120)$$

As \mathbf{w}_k are orthogonal, \mathbf{W} is unitary $\mathbf{W}^* \mathbf{W} = \mathbf{I}$ or $\mathbf{W}^* = \mathbf{W}^{-1}$. Also, as $w[l, k] = e^{j2\pi kl/N} = w[k, l]$, $\mathbf{W} = \mathbf{W}^T$ is a symmetric. We therefore have:

$$\mathbf{W}^{-1} = \overline{\mathbf{W}}, \quad \text{i.e.} \quad \mathbf{W}\overline{\mathbf{W}} = \mathbf{I} \quad (4.121)$$

Now the DFT of a signal vector \mathbf{x} can be expressed in the following matrix forms:

$$\mathbf{x} = \mathbf{W}\mathbf{X} = [\mathbf{w}_0, \dots, \mathbf{w}_{N-1}] \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \sum_{k=0}^{N-1} X[k] \mathbf{w}_k \quad (4.122)$$

Left multiplying $\mathbf{W}^{-1} = \overline{\mathbf{W}}$ on both sides, we get

$$\overline{\mathbf{W}}\mathbf{x} = \overline{\mathbf{W}}\mathbf{W}\mathbf{X} = \mathbf{X} \quad (4.123)$$

i.e.,

$$\mathbf{X} = \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \overline{\mathbf{W}}\mathbf{x} = \begin{bmatrix} \overline{\mathbf{w}}_0^T \\ \vdots \\ \overline{\mathbf{w}}_{N-1}^T \end{bmatrix} \mathbf{x} \quad (4.124)$$

where the k th coefficient is the projection of the signal vector \mathbf{x} onto the k th basis vector \mathbf{w}_k :

$$X[k] = \langle \mathbf{x}, \mathbf{w}_k \rangle = \overline{\mathbf{w}}_k^T \mathbf{x} = \mathbf{x}^T \overline{\mathbf{w}}_k \quad (4.125)$$

Equations 4.122 and 4.123 form the DFT pair in matrix form (while Eq. 4.110 is the component form):

$$\begin{cases} \mathbf{X} = \overline{\mathbf{W}}\mathbf{x} & \text{(forward)} \\ \mathbf{x} = \mathbf{W}\mathbf{X} & \text{(inverse)} \end{cases} \quad (4.126)$$

As a unitary operation, the DFT is actually a rotation in \mathbb{C}^N , represented by the unitary matrix \mathbf{W} . Any signal vector $\mathbf{x} = [x[0], \dots, x[N-1]]^T$ given under the standard basis \mathbf{e}_n ($n = 0, \dots, N-1$) can also be expressed in terms of a different set of basis vectors \mathbf{w}_k ($k = 0, \dots, N-1$):

$$\begin{aligned} \mathbf{x} &= \mathbf{I}\mathbf{x} = [\mathbf{e}_0, \dots, \mathbf{e}_{N-1}]\mathbf{x} = \sum_{n=0}^{N-1} x[n] \mathbf{e}_n \\ &= \mathbf{W}\mathbf{X} = [\mathbf{w}_0, \dots, \mathbf{w}_{N-1}]\mathbf{X} = \sum_{k=0}^{N-1} X[k] \mathbf{w}_k \end{aligned} \quad (4.127)$$

where $\mathbf{w}_k = \mathbf{W}\mathbf{e}_k$ is obtained by rotating the standard basis vector \mathbf{e}_k ($k = 0, \dots, N-1$). Equivalently, the signal vector is rotated in the opposite direction to become $\mathbf{X} = \mathbf{W}^{-1}\mathbf{x} = \overline{\mathbf{W}}\mathbf{x}$. As rotation does not change vector norm (Parseval's identity), the signal energy is conserved $\|\mathbf{x}\| = \|\mathbf{X}\|$, i.e., either the original signal \mathbf{x} in time domain or its Fourier coefficients X in frequency domain contains the same amount of energy or information.

We now consider specifically the following three examples for $N=2$, 4 and 8.

- $N = 2$, the element of the l th row and k th column ($l, k = 0, 1$) of the 2-point DFT matrix is:

$$w[l, k] = \frac{1}{\sqrt{2}}(e^{j2\pi/N})^{kl} = \frac{1}{\sqrt{2}}(e^{j\pi})^{kl} = \frac{1}{\sqrt{2}}(-1)^{kl} \quad (4.128)$$

and the DFT matrix is:

$$\mathbf{W}_{2 \times 2} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (4.129)$$

The DFT of a 2-point signal $\mathbf{x} = [x[0], x[1]]^T$ can be trivially found as:

$$\mathbf{X} = \begin{bmatrix} X[0] \\ X[1] \end{bmatrix} = \overline{\mathbf{W}}\mathbf{x} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} x[0] + x[1] \\ x[0] - x[1] \end{bmatrix} \quad (4.130)$$

We see that the first component $X[0]$ is proportional to the sum of the two signal samples representing the average or DC component of the signal, and the second $X[1]$ is proportional to the difference between the two samples representing the variations (details) in the signal.

- $N = 4$, the element of the l th row and k th column ($l, k = 0, \dots, 3$) of the 4-point DFT matrix is:

$$w[l, k] = \frac{1}{\sqrt{N}} (e^{j2\pi/N})^{kl} = \frac{1}{2} (e^{j\pi/2})^{kl} = j^{kl} \quad (4.131)$$

The 4 by 4 DFT matrix is:

$$\mathbf{W}_{4 \times 4} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & -1 & 0 \\ 1 & -1 & 1 & -1 \\ 1 & 0 & -1 & 0 \end{bmatrix} + \frac{j}{2} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix} \quad (4.132)$$

- $N = 8$, we have:

$$w[l, k] = \frac{1}{\sqrt{N}} (e^{j2\pi/N})^{kl} = \frac{1}{\sqrt{8}} (e^{j\pi/4})^{kl} = \frac{1}{\sqrt{8}} (0.707 + j 0.707)^{kl} \quad (4.133)$$

The real and imaginary parts of the DFT matrix $\mathbf{W} = \mathbf{W}_r + j\mathbf{W}_j$ are respectively:

$$\mathbf{W}_r = \frac{1}{\sqrt{8}} \begin{bmatrix} 1.0 & 1.0 & 1.0 & 1.0 & 1.0 & 1.0 & 1.0 & 1.0 \\ 1.0 & 0.7 & 0.0 & -0.7 & -1.0 & -0.7 & 0.0 & 0.7 \\ 1.0 & 0.0 & -1.0 & 0.0 & 1.0 & 0.0 & -1.0 & -0.0 \\ 1.0 & -0.7 & 0.1 & 0.7 & -1.0 & 0.7 & 0.0 & -0.7 \\ 1.0 & -1.0 & 1.0 & -1.0 & 1.0 & -1.0 & 1.0 & -1.0 \\ 1.0 & -0.7 & 0.0 & 0.7 & -1.0 & 0.7 & 0.0 & -0.7 \\ 1.0 & 0.0 & -1.0 & 0.0 & 1.0 & 0.0 & -1.0 & -0.0 \\ 1.0 & 0.7 & 0.0 & -0.7 & -1.0 & -0.7 & 0.0 & 0.7 \end{bmatrix} \quad (4.134)$$

and

$$\mathbf{W}_j = \frac{1}{\sqrt{8}} \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & -0.7 & -1.0 & -0.7 & 0.0 & 0.7 & 1.0 & 0.7 \\ 0.0 & -1.0 & 0.0 & 1.0 & 0.0 & -1.0 & 0.0 & 1.0 \\ 0.0 & -0.7 & 1.0 & -0.7 & 0.0 & 0.7 & -1.0 & 0.7 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.7 & -1.0 & 0.7 & 0.0 & -0.7 & 1.0 & -0.7 \\ 0.0 & 1.0 & 0.0 & -1.0 & 0.0 & 1.0 & 0.0 & -1.0 \\ 0.0 & 0.7 & 1.0 & 0.7 & 0.0 & -0.7 & -1.0 & -0.7 \end{bmatrix} \quad (4.135)$$

The values of \mathbf{W}_r and \mathbf{W}_j are also plotted in the first two columns of Fig.4.16.

Example 4.9: The $N = 8$ samples of a real signal $x[n]$ are given as a complex vector with zero imaginary part:

$$\mathbf{x} = [(0, 0), (0, 0), (2, 0), (3, 0), (4, 0), (0, 0), (0, 0), (0, 0)]^T \quad (4.136)$$

The real and imaginary parts of the 8-point DFT matrix \mathbf{W} are given in Eqs.4.134 and 4.135 respectively. The DFT of the signal can be carried out by matrix multiplication:

$$\mathbf{X} = \overline{\mathbf{W}}\mathbf{x} \quad (4.137)$$

where $\mathbf{X} = \mathbf{X}_r + j\mathbf{X}_j$ are the $N = 8$ DFT coefficients:

$$\begin{aligned} \mathbf{X}_r &= [3.18, -2.16, 0.71, -0.66, 1.06, -0.66, 0.71, -2.16]^T \\ \mathbf{X}_j &= [0.0, -1.46, 1.06, -0.04, 0.0, 0.04, -1.06, 1.46]^T \end{aligned} \quad (4.138)$$

The signal $x[n]$ can be reconstructed by the inverse DFT from its DFT coefficients $X[k]$:

$$\mathbf{x} = \begin{bmatrix} x[0] \\ \vdots \\ x[7] \end{bmatrix} = \mathbf{W}\mathbf{X} = [\mathbf{w}_0, \dots, \mathbf{w}_7] \begin{bmatrix} X[0] \\ \vdots \\ X[7] \end{bmatrix} = \sum_{k=0}^7 X[k]\mathbf{w}_k \quad (4.139)$$

The reconstruction of this 8-point discrete signal as a linear combination of its frequency components, is illustrated in columns 3 (real) and 4 (imaginary) of Fig. 4.16, as the discrete version of the corresponding Fourier series expansion of a continuous signal. Here progressively more and higher frequency components are included in the reconstruction for better approximation of the signal, from the DC component alone (top row) until all N frequency components are used for a perfect reconstruction (last row).

4.2.3 Properties of DFT

As one of the variations of the generic continuous-time Fourier transform (CTFT), the DFT shares all the properties of the CTFT discussed previously, but in different forms. Here we consider only a set of selected properties and leave out some of the proofs which should be very similar to those for the corresponding CTFT properties.

- **Time and frequency shift**

$$\mathcal{F}[x[n \pm n_0]] = X[k]e^{\pm j2\pi n_0 k/N}, \quad \mathcal{F}[x[n]e^{\mp j2\pi n k_0/N}] = X[k \pm k_0] \quad (4.140)$$

- **DC and highest frequency representable**

$X_r[0]$ represents the DC offset of the signal (zero frequency):

$$X_r[0] = \sum_{n=0}^{N-1} x_r[n] \cos\left(\frac{2\pi n 0}{N}\right) = \sum_{n=0}^{N-1} x_r[n] \quad (4.141)$$

and $X_r[N/2]$ represents the highest frequency component:

$$X_r[N/2] = \sum_{n=0}^{N-1} x_r[n] \cos\left(\frac{2\pi n N/2}{N}\right) = \sum_{n=0}^{N-1} x_r[n](-1)^n \quad (4.142)$$

When $k = 0$ and $n = N/2$, the imaginary parts $X_j[0] = X_j[N/2] = 0$ are zero as $\sin(0) = \sin(n\pi) = 0$.

- **Symmetry**

The DFT is complex transform which can be separated into real and imaginary parts:

$$\begin{aligned} X[k] &= \sum_{n=0}^{N-1} x[n] e^{-j2\pi nk/N} \\ &= \sum_{n=0}^{N-1} [x_r[n] + jx_i[n]] \left[\cos\left(\frac{2\pi nk}{N}\right) - j \sin\left(\frac{2\pi nk}{N}\right) \right] = X_r[k] + jX_j[k] \end{aligned} \quad (4.143)$$

where

$$\begin{aligned} X_r[k] &= \sum_{n=0}^{N-1} x_r[n] \cos\left(\frac{2\pi nk}{N}\right) + \sum_{n=0}^{N-1} x_j[n] \sin\left(\frac{2\pi nk}{N}\right) \\ X_j[k] &= \sum_{n=0}^{N-1} x_j[n] \cos\left(\frac{2\pi nk}{N}\right) - \sum_{n=0}^{N-1} x_r[n] \sin\left(\frac{2\pi nk}{N}\right) \end{aligned} \quad (4.144)$$

In particular, if $x[n] = x_r[n]$ is real ($x_j[n] = 0$), then $X_r[k]$ is even

$$X_r[k] = \sum_{n=0}^{N-1} x_r[n] \cos\left(\frac{2\pi nk}{N}\right) = X_r[-k] \quad (4.145)$$

and $X_j[k]$ is odd

$$X_j[k] = - \sum_{n=0}^{N-1} x_r[n] \sin\left(\frac{2\pi nk}{N}\right) = -X_j[-k] \quad (4.146)$$

- **Convolution theorem**

The convolution of two finite and discrete $x[n]$ and $h[n]$ ($n = 0, \dots, N-1$) is defined as

$$y[n] = h[n] * x[n] = \sum_{m=0}^{N-1} x[m]h[n-m], \quad (n = 0, \dots, N-1) \quad (4.147)$$

As both $x[n+N] = x[n]$ and $h[n+N] = h[n]$ are assumed to be periodic with period N , it is obvious that the result $y[n]$ of the convolution is also periodic: $y[n+N] = y[n]$. The convolution is therefore also referred to as a *circular convolution*.

Let $X[k] = \mathcal{F}[x[n]]$ and $H[k] = \mathcal{F}[h[n]]$, then the convolution theorem states:

$$\mathcal{F}[h[n] * x[n]] = H[k]X[k], \quad \mathcal{F}[h[n]x[n]] = H[k] * X[k] \quad (4.148)$$

We now prove the first part of Eq.4.148:

$$\begin{aligned} \mathcal{F}[x[n] * h[n]] &= \sum_{n=0}^{N-1} \left[\sum_{m=0}^{N-1} x[m]h[n-m] \right] e^{-j2\pi nk/N} \\ &= \sum_{m=0}^{N-1} x[m] \left[\sum_{n=0}^{N-1} h[n-m]e^{-j2\pi(n-m)k/N} \right] e^{-j2\pi mk/N} \\ &= H[k] \sum_{m=0}^{N-1} x[m]e^{-j2\pi km/N} = H[k]X[k] \end{aligned} \quad (4.149)$$

Note that due to the assumed periodicity, the upper and lower limits for of the summation are not important so long as they cover all N terms in the period. The second part of Eq.4.148 can be similarly proved.

- **Diagonalization of circulant matrix**

An N by N matrix \mathbf{H} can be constructed based on $h[n]$ of the convolution above, with its element in the m th row and n th column defined as $h[m, n] = h[m-n]$, so that the circular convolution in Eq.4.147 can be expressed as a matrix multiplication $\mathbf{y} = \mathbf{Hx}$:

$$\begin{bmatrix} y[0] \\ y[1] \\ \vdots \\ y[N-2] \\ y[N-1] \end{bmatrix} = \begin{bmatrix} h[0] & h[N-1] & \cdots & h[2] & h[1] \\ h[1] & h[0] & \cdots & h[3] & h[2] \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ h[N-2] & h[N-3] & \cdots & h[0] & h[N-1] \\ h[N-1] & h[N-2] & \cdots & h[1] & h[0] \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ \vdots \\ x[N-2] \\ x[N-1] \end{bmatrix} \quad (4.150)$$

This matrix \mathbf{H} is a *circulant matrix* each row of which is a circularly right-rotated version of the row above. Let $H[k] = \mathcal{F}[h[n]]$ be the DFT of $h[n]$, and

$\mathbf{w}_k = [w^{j2\pi 0k/N}, \dots, w^{j2\pi(N-1)k/N}]^T$ be the kth column vector of the DFT matrix \mathbf{W} , then we can show that they are respectively the eigenvalue and eigenvector of the matrix \mathbf{H} :

$$\mathbf{H}\mathbf{w}_k = H[k]\mathbf{w}_k, \quad (k = 0, \dots, N-1) \quad (4.151)$$

To show this, we first consider the mth element of the left hand side:

$$\begin{aligned} & \sum_{n=0}^{N-1} h[m, n]w^{j2\pi nk/N} = \sum_{n=0}^{N-1} h[m-n]w^{j2\pi nk/N} \\ &= \sum_{l=0}^{N-1} h[l]w^{-j2\pi lk/N}w^{j2\pi mk/N} = H[k]w^{j2\pi mk/N} \\ & \quad (m = 0, \dots, N-1) \end{aligned} \quad (4.152)$$

where we have assumed $m - n = l$. This result happens be the mth element of the right hand side of Eq.4.151, i.e., Eq.4.151 holds. If we further define $\mathbf{D} = \text{diag}(H[0], \dots, H[N-1])$ as a diagonal matrix composed of all N DFT coefficients along the main diagonal, then Eq.4.151 can be written in matrix form as:

$$\mathbf{H}\mathbf{W} = \mathbf{W}\mathbf{D}, \quad \text{i.e.} \quad \mathbf{W}^{-1}\mathbf{H}\mathbf{W} = \overline{\mathbf{W}}\mathbf{H}\mathbf{W} = \mathbf{D} \quad (4.153)$$

We see that the circulant matrix \mathbf{H} is diagonalized by the DFT matrix $\mathbf{W} = [\mathbf{w}_0, \dots, \mathbf{w}_{N-1}]$. Now by taking the DFT on both sides of $\mathbf{y} = \mathbf{H}\mathbf{x}$ in Eq.4.150 (by pre-multiplying $\overline{\mathbf{W}}$), we get:

$$\mathbf{Y} = \mathcal{F}[\mathbf{y}] = \overline{\mathbf{W}}\mathbf{y} = \overline{\mathbf{W}}\mathbf{H}\mathbf{x} = \overline{\mathbf{W}}\mathbf{H}\mathbf{W}\overline{\mathbf{W}}\mathbf{x} = \mathbf{D}\mathbf{X} \quad (4.154)$$

or in component form:

$$\begin{bmatrix} Y[0] \\ Y[1] \\ \vdots \\ Y[N-1] \end{bmatrix} = \begin{bmatrix} H[0] & 0 & \cdots & 0 \\ 0 & H[1] & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & H[N-1] \end{bmatrix} \begin{bmatrix} X[0] \\ X[1] \\ \vdots \\ X[N-1] \end{bmatrix} \quad (4.155)$$

The kth element of this vector equation is:

$$Y[k] = H[k]X[k] \quad (4.156)$$

This is of course the matrix form of the discrete convolution theorem.

Example 4.10: Given a discrete LTI system with impulse response $h[n]$ and an input $x[n]$:

$$\mathbf{x} = [1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8]^T, \quad \mathbf{h} = [1 \ 2 \ 3 \ 0 \ 0 \ 0 \ 0 \ 0]^T \quad (4.157)$$

we want to find the output $y[n]$ as the convolution:

$$y[n] = h[n] * x[n] = \sum_m h[n-m]x[m], \quad (n = 0, \dots, N-1) \quad (4.158)$$

Different from the convolution in Example 4.3, here $y[n] = h[n] * x[n]$ is a *circular convolution* as both \mathbf{x} and \mathbf{h} are assumed to be periodic with period $N = 8$, and, consequently, their convolution is also periodic, as shown below:

m	$\dots -2 -1$	$0 1 2 3 4 5 6 7$	$8 9 10 \dots$
$x[m]$	$\dots 7 8$	$1 2 3 4 5 6 7 8$	$1 2 3 \dots$
$h[0-m]$	$\dots 3 2$	1	...
$h[1-m]$	$\dots 3$	2 1	...
$h[2-m]$	\dots	3 2 1	...
$h[3-m]$	\dots	3 2 1	...
$h[4-m]$	\dots	3 2 1	...
$h[5-m]$	\dots	3 2 1	...
$h[6-m]$	\dots	3 2 1	...
$h[7-m]$	\dots	3 2 1	...
$h[8-m]$	\dots	3 2 1	...
$h[9-m]$	\dots	3 2 1	...
$h[10-m]$	\dots	3 2 1	...
$y[n]$	$\dots 34 40$	38 28 10 16 22 28 34 40	38 28 10 ...

For example, when $n = 2$, we have:

$$\begin{aligned} y[2] &= \sum_{m=0}^7 h[2-m]x[m] = h[2]x[0] + h[1]x[1] + h[0]x[2] \\ &= 3 \times 1 + 2 \times 2 + 1 \times 3 = 10 \end{aligned} \quad (4.160)$$

We see that the resulting $y[n+8] = y[n]$ is indeed periodic.

Next, we show that this discrete convolution can also be carried out by DFT. We find the 8-point DFTs $\mathbf{X} = DFT[\mathbf{x}]$ and $\mathbf{H} = DFT[\mathbf{h}]$ and also their element-wise product $\mathbf{Y} = [Y[0], \dots, Y[7]]^T$, where $Y[k] = H[k]X[k]$ ($k = 0, \dots, 7$):

$$\mathbf{X} = \begin{bmatrix} 36 \\ -4 + 9.657j \\ -4 + 4j \\ -4 + 1.657j \\ -4 \\ -4 - 1.657j \\ -4 - 4j \\ -4 - 9.657j \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} 6 \\ 2.414 - 4.414j \\ -2 - 2j \\ -0.414 + 1.586j \\ 2 \\ -0.414 - 1.586j \\ -2 + 2j \\ 2.414 + 4.414j \end{bmatrix}, \quad \mathbf{Y} = \begin{bmatrix} 216 \\ 32.971 + 40.971j \\ 16 \\ -0.971 - 7.029j \\ -8 \\ -0.971 + 7.029j \\ 16 \\ 32.971 - 40.971j \end{bmatrix} \quad (4.161)$$

The convolution $y[n] = h[n] * x[n]$ can be obtained by inverse DFT to be:

$$\mathbf{y} = DFT^{-1}[\mathbf{Y}] = [38 28 10 16 22 28 34 40]^T \quad (4.162)$$

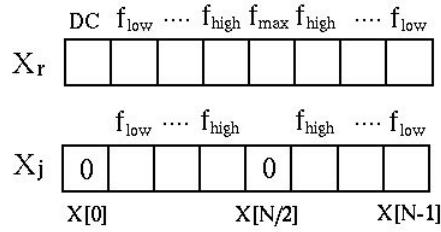


Figure 4.17 DFT coefficients for different frequency components

Below we further consider two issues regarding the N DFT coefficients. First, we consider the interpretation of the DFT coefficients $X[k]$ ($k = 0, \dots, N - 1$) in order to know how they can be properly modified for various desired data processing purposes such as filtering (e.g., low, band or high-pass/stop). Here we assume the time signal $x = [x[0], \dots, x[N - 1]]^T$ is real and N is even, therefore the real part of its spectrum $X_r[k] = X_r[-k]$ is even and the imaginary part $X_j[k] = -X_j[-k]$ is odd, and the inverse DFT can be written as:

$$x[n] = \operatorname{Re} \left[\sum_{k=0}^{N-1} X[k] e^{j2\pi nk/N} \right] \quad (4.163)$$

$$\begin{aligned} &= \sum_{k=0}^{N-1} [X_r[k] \cos(2\pi nk/N) - X_j[k] \sin(2\pi nk/N)] \\ &= \sum_{k=0}^{N-1} |X[k]| \cos(2\pi nk/N + \angle X[k]) \end{aligned} \quad (4.164)$$

where

$$\begin{cases} |X[k]| = \sqrt{X_r^2[k] + X_j^2[k]} \\ \angle X[k] = \tan^{-1}(X_j[k]/X_r[k]) \end{cases}, \quad \begin{cases} X_r[k] = |X[k]| \cos \angle X[k] \\ X_j[k] = |X[k]| \sin \angle X[k] \end{cases} \quad (4.165)$$

and the forward DFT is:

$$X[k] = \frac{1}{N} \sum_{n=0}^{N-1} x[n] e^{-j2\pi nk/N} = \frac{1}{N} \sum_{n=0}^{N-1} x[n] [\cos(2\pi nk/N) - j \sin 2\pi nk/N] \quad (4.166)$$

These N DFT coefficients and the frequency components they represent are illustrated in Fig.4.17.

Consider specifically the following terms in the summation in Eq.4.164:

- $k = 0$:

$$X[0] = \frac{1}{N} \sum_{n=0}^{N-1} x[n] \quad (4.167)$$

This is the DC component, which is real with zero phase;

- $k = N/2$:

$$\begin{aligned} X[N/2] &= \frac{1}{N} \sum_{n=0}^{N-1} x[n] \cos(n\pi) = \frac{1}{N} \sum_{n=0}^{N-1} x[n](-1)^n \\ &= \frac{1}{N} \sum_{n=0,2,\dots,N-2} [x[n] - x[n+1]] \end{aligned} \quad (4.168)$$

This is the coefficient for the highest frequency component $\cos(n\pi) = (-1)^n$ with frequency $f_{max} = 1/2$ with period $1/f_{max} = 2$. Same as $X[0]$, $X[N/2]$ is also real with zero phase shift;

- $k = 1, \dots, N/2 - 1$:

These terms represent $(N-2)/2$ sinusoids $|X[k]| \cos(2\pi nk/N + \angle X[k])$ with frequency k/N , amplitude $|X[k]|$ and phase shift $\angle X[k]$;

- $k = N/2 + 1, \dots, N - 1$:

Due to the periodicity $X[k-N] = X[k]$, these terms are the same as those in the range $k = -1, \dots, -(N/2-1)$ or $-k = 1, \dots, N/2-1$ and we have

$$|X[-k]| \cos(-2\pi nk/N + \angle X[-k]) = |X[k]| \cos(2\pi nk/N + \angle X[k]) \quad (4.169)$$

(Note that $\angle X[k]$ is odd and cos function is even.) These are the same sinusoids as those in the previous range.

Combining all the terms above together we can rewrite Eq.4.164 as:

$$x[n] = X[0] + X[N/2] \cos(n\pi) + 2 \sum_{k=1}^{N/2-1} |X[k]| \cos(2\pi nk/N + \angle X[k]) \quad (4.170)$$

This is the discrete version of Eq. 3.134 in the case of the continuous Fourier transform.

We consider as an example each of the $N = 8$ complex coefficients $X[k] = X_r[k] + jX_j[k]$ given in Eq.4.138 in Example 4.9:

- $X_r[0] = 3.18/\sqrt{8}$ is proportional to the sum of all signal samples $x[n]$, therefore it represents the average of the signal. As $X_j[0] = 0$, $\angle X[0] = 0$.
- $X_r[4] = 1.06/\sqrt{8}$ is the amplitude of the highest frequency component with $f_4 = 4/8$. As $X_j[4] = 0$, $\angle X[4] = 0$.
- The remaining $(N-2)/2 = 3$ pairs of terms corresponding to $k = 1, 7$, $k = 2, 6$ and $k = 3, 5$ represent 3 sinusoids with frequency $f_k = k/N$, amplitude $|X[k]| = \sqrt{X_r^2[k] + X_j^2[k]}$, and phase $\angle X[k] = \tan^{-1}(X_j[k]/X_r[k])$:
 - $k = 1, 7$:
 $f_1 = 1/8$, $\omega_1 = 0.79$, $|X[1]| = 2.61/\sqrt{8}$, $\angle X[1] = -2.55$ rad/sec.
 - $k = 2, 6$:
 $f_2 = 2/8$, $\omega_2 = 1.57$, $|X[2]| = 1.28/\sqrt{8}$, $\angle X[2] = 0.98$ rad/sec.
 - $k = 3, 5$:
 $f_3 = 3/8$, $\omega_3 = 2.36$, $|X[3]| = 0.67/\sqrt{8}$, $\angle X[3] = -3.08$ rad/sec.

Now the signal can be expanded as (Eq. 4.170):

$$\begin{aligned} x[n] &= \frac{1}{\sqrt{N}}[X[0] + 2 \sum_{n=1}^3 |X[k]| \cos(\frac{2\pi nk}{N} + \angle X[k]) + X[4] \cos(m\pi)] \\ &= \frac{1}{\sqrt{8}}[3.18 + 2(2.61 \cos(0.79n - 2.55) + 1.28 \cos(1.57n + 0.98) \\ &\quad + 0.67 \cos(2.36n - 3.08)) + 1.06 \cos(3.14n)], \quad (n = 0, \dots, 7) \end{aligned} \quad (4.171)$$

Next, we consider the centralization of the DFT spectrum. In all previous discussions regarding the Fourier spectrum, the DC component of zero frequency at the origin is always conceptually assumed to be in the middle of the frequency axis, while the higher frequencies (both positive and negative) are farther away from the middle point on both sides of the origin. However, on the other hand, the N DFT coefficients $X[k]$ in the vector $\mathbf{X} = \overline{\mathbf{W}}\mathbf{x}$ generated by the DFT algorithm are indexed in such a way that the DC component $X[0]$ of zero frequency is the first (leftmost) element of \mathbf{X} while the highest frequency component $X[N/2]$ is in the middle. It is therefore sometimes desirable to rearrange the DFT spectrum \mathbf{X} so that it is in consistent with the conceptual form of the spectrum. Specifically, this centralization process can be carried out by right shifting all components $X[k]$ in \mathbf{X} by $N/2$, so that in the resulting vector $X'[k + N/2] = X[k]$ the DC component ($k = 0$) appears in the middle at $N/2$, while the elements in the first half ($k < N/2$) for the positive frequencies are shifted to the second half to the right of the DC component, and those originally in the second half ($k > N/2$) for the negative frequencies are shifted to the first half to the left of the DC, due to the periodicity $X'[k + N/2] = X'[k + N/2 - N] = X'[k - N/2]$ (indicating this right shift by $N/2$ is equivalent to left shift by $N/2$).

Computationally, according to the frequency shift property of the DFT, this centralization process can also be realized in time domain before the DFT, by multiplying the time sample $x[n]$ by $e^{jn\pi} = (-1)^n$:

$$\mathcal{F}[x[n]e^{jn\pi}] = \mathcal{F}[x[n](-1)^n] = X[k - N/2] \quad (4.172)$$

In other words, if we negate all odd-indexed samples of the discrete signal so that it becomes: $x[0], -x[1], x[2], -x[3], \dots$, then its DFT spectrum becomes centralized with DC in the middle of the array. This process is illustrated in Fig.4.18.

As an example, the real signal $\mathbf{x} = [0, 0, 2, 3, 4, 0, 0]^T$ and its DFT coefficients in Example 4.9 are plotted in the top two panels of Fig.4.19, respectively. If we negate all odd-indexed elements of the signal, the spectrum becomes centralized, as plotted in the third panel of the figure. Note that as the time signal is real, the real part of spectrum is even $X_r[1] = X_r[7]$, $X_r[2] = X_r[6]$, $X_r[3] = X_r[5]$; and the imaginary part is odd: $X_j[1] = -X_j[7]$, $X_j[2] = -X_j[6]$, $X_j[3] = -X_j[5]$. Also note that $X_r[0] \neq X_r[4]$ and $X_j[0] = X_j[4] = 0$ are always zero.

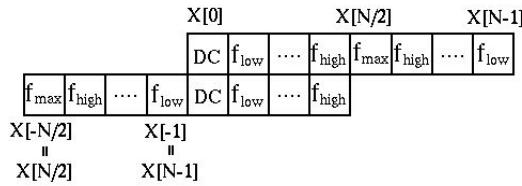


Figure 4.18 Centralization of the DFT spectrum

Coefficient Indexing of DFT algorithm (top) and conceptual DFT spectrum (bottom). Note that $X[-N/2] = X[N/2], \dots, X[-1] = X[N - 1]$.

$$\begin{aligned} \mathbf{X}_r &= [1.06, -0.66, 0.71, -2.16, 3.18, -2.16, 0.71, -0.66]^T \\ \mathbf{X}_j &= [0.0, 0.04, -1.06, 1.46, 0.0, -1.46, 1.06, -0.04]^T \end{aligned} \quad (4.173)$$

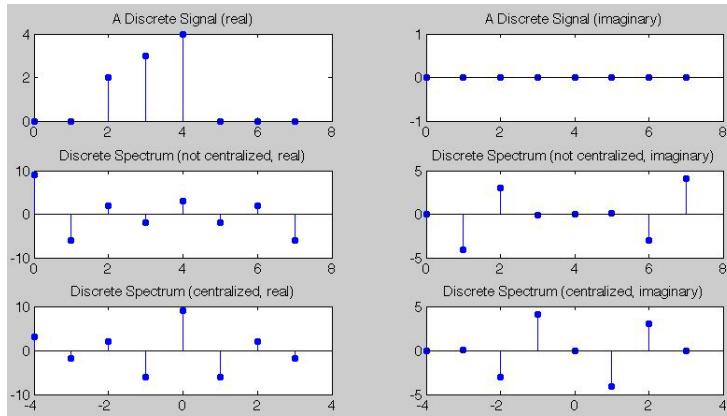


Figure 4.19 A discrete signal (top) and its DFT spectrum before (middle) and after (bottom) centralization

The real and imaginary parts of these complex coefficients are shown respectively in the middle and bottom panels.

4.2.4 DFT Computation and Fast Fourier Transform (FFT)

The Fourier transform of a signal can be carried out numerically by a computer only if the signal is (a) discrete and (b) finite, i.e., out of all four different forms of the Fourier transform discussed above, only DFT can actually be carried out.

Following the definition of the DFT in Eq.4.110, it can be easily implemented by the following Matlab code:

```
function X=dft(x)
```

```

N=length(x);
X=[exp(-j*2*pi*[0:N-1]'*[0:N-1]/N)/sqrt(N)]*x;
end

function x=idft(X)
N=length(X);
x=[exp(j*2*pi*[0:N-1]'*[0:N-1]/N)/sqrt(N)]*X;
end

```

Here the signal x and its DFT spectrum X are both assumed to be column vectors. Matlab has a built-in function for the fast Fourier transform FFT to be discussed below.

Here the signal x and its spectrum X are assumed to be column vectors (same as in the text). The C code for DFT and inverse DFT based on array multiplications in Eq.4.126 is listed below. The function dft takes a both the real and imaginary parts of a complex data vector for the time signal as the input and returns the complex DFT coefficients. This is an in-place algorithm, i.e., the input vector $xx[n] + j xi[n]$ ($n = 0, \dots, N - 1$) for the time signal will be overwritten by the output, its DFT coefficients. The same function is also used for the inverse DFT, in which case the input is the DFT coefficients while the output is the reconstructed signal vector in time domain. The function carries out forward DFT when the parameter inv=0, or inverse DFT when inv=1.

```

void dft(xx,xi,N,inv)
    float *xr, *xi;           // real and imaginary parts of data
    int N;                   // size of data
    int inverse;             // inv=0 forward DFT, inv=1 inverse DFT
{ int k,m,n;
    float arg,s,c,*yr,*yi;
    yr=(float *) malloc(N*sizeof(float));
    yi=(float *) malloc(N*sizeof(float));
    for (k=0; k<N; k++) {   // for all N frequency components
        yr[k]=yi[k]=0;
        for (n=0; n<N; n++) { // for all N data samples
            arg=-2*Pi*n*k/N;
            if (inv) arg=-arg; // minus sign not needed for inverse DFT
            c=cos(arg); s=sin(arg);
            yr[k]+=xr[n]*c-xi[n]*s;
            yi[k]+=xi[n]*c+xr[n]*s;
        }
    }
    arg=1.0/sqrt((float)N);
    for (k=0; k<N; k++)
        { xr[k]=arg*yr[k]; xi[k]=arg*yi[k]; }

```

```

    free(yr); free(yi);
}

```

The computational complexity of this algorithm is $O(N^2)$, due obviously to the two nested loops each of size N , i.e., it takes $O(N)$ operations to obtain each of the N coefficients $X[k]$. Due to such a high computational complexity, the actual application of the Fourier transform was quite limited in practice before the fast algorithm was available.

To speed up the computation, a revolutionary *fast Fourier transform (FFT)* algorithm was developed in 1960's by which the complexity of a DFT is reduced from $O(N^2)$ to $O(N \log_2 N)$. For example, if the signal size is $N = 10^3 \approx 2^{10}$, then $O(N^2) = 10^6$ but $O(N \log_2 N) \approx 10^4$, the complexity is reduced by 100 fold. Due to this significant improvement in computational efficiency, the Fourier transform became highly valuable not only theoretically but also practically.

The FFT algorithm is based on the following properties of the elements of the matrix \mathbf{W} . We first define $w_N = e^{-j2\pi/N}$ and note the following properties of w_N :

$$w_N^{kN} = e^{-j2k\pi N/N} = e^{-j2k\pi} = 1 \quad (4.174)$$

$$w_{2N}^{2k} = e^{-j2k2\pi/2N} = e^{-jk2\pi/N} = w_N^k \quad (4.175)$$

$$w_{2N}^N = e^{-j2N\pi/2N} = e^{-j\pi} = -1 \quad (4.176)$$

$$(4.177)$$

We let $N = 2M$ and write an N -point DFT as:

$$\begin{aligned}
X[k] &= \sum_{n=0}^{N-1} x[n]e^{j2\pi nk/N} = \sum_{n=0}^{N-1} x[n]w_N^{nk} \\
&= \sum_{n=0}^{M-1} x[2n]w_{2M}^{2nk} + \sum_{n=0}^{M-1} x[2n+1]w_{2M}^{(2n+1)k} \\
&= \sum_{n=0}^{M-1} x[2n]w_M^{nk} + \sum_{n=0}^{M-1} x[2n+1]w_M^{nk}w_{2M}^k \\
&= X_{even}[k] + X_{odd}[k]w_{2M}^k
\end{aligned} \quad (4.178)$$

where we have used Eq.4.175 and defined:

$$X_{even}[k] = \sum_{n=0}^{N-1} x[2n]w_M^{nk}, \quad \text{and} \quad X_{odd}[k] = \sum_{n=0}^{N-1} x[2n+1]w_M^{nk} \quad (4.179)$$

These are two $N/2$ -point DFTs for the even and odd indexed signal samples, respectively. In other words, an N -point DFT is now converted into two $N/2$ -point DFTs. Also note that this is only for the first half of the N coefficients $X[k]$ for $k = 0, \dots, M-1$. The coefficients in the second half can be obtained

by replacing k in Eq. 4.178 by $k + M$:

$$X[k + M] = X_{even}[k + M] + X_{odd}[k + M]w_{2M}^{k+M} \quad (4.180)$$

Due to Eq. 4.174, we have:

$$X_{even}[k + M] = \sum_{n=0}^{M-1} x[2n]w_M^{n(k+M)} = \sum_{n=0}^{M-1} x[2n]w_M^{nk} = X_{even}[k] \quad (4.181)$$

and similarly $X_{odd}[k + M] = X_{odd}[k]$. Also, due to Eq. 4.176, we have:

$$w_{2M}^{k+M} = w_{2M}^k w_{2M}^M = -w_{2M}^k \quad (4.182)$$

then Eq. 4.180 can be written as:

$$X[k + M] = X_{even}[k] - X_{odd}[k]w_{2M}^k \quad (4.183)$$

The N -point DFT can now be obtained from Eqs. 4.178 and 4.183 with complexity of $O(N)$, once $X_{even}[k]$ and $X_{odd}[k]$ are obtained by the two $N/2$ -point DFTs Eq. 4.179, each of which can be carried out in exactly the same way. In other words, this process of reducing the data size by half can be carried out recursively $\log_2 N$ times until eventually the size is 1 and DFT coefficient is simply the same as the signal sample. This recursion is illustrated in Fig. 4.20. We see that the an N -point DFT is carried out in $\log_2 N$ stages, each with $O(N)$ complexity, with total complexity of $O(N \log_2 N)$.

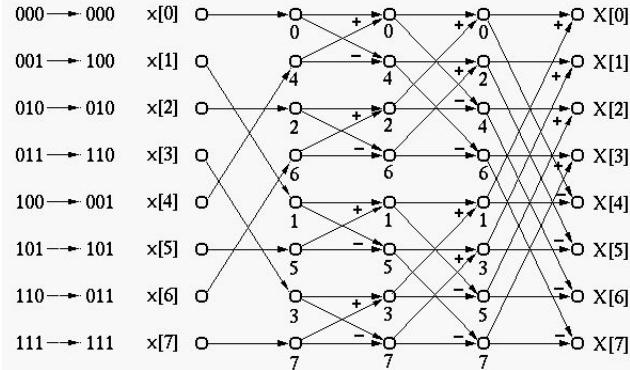


Figure 4.20 The fast Fourier transform algorithm

The C code for the FFT algorithm is given below. The function fft takes as input two vectors of N elements each for the real and imaginary parts of the complex time signal, and returns its complex DFT coefficients as the outputs. Here the total number of vector elements N is assumed to be a power of 2, so that the FFT algorithm can be conveniently implemented. This is an in-place algorithm, i.e., the input vector $xr[n] + j xi[n]$ ($n = 0, \dots, N - 1$) for the time signal will be overwritten by its DFT coefficients. The same function is also used for the inverse DFT, in which case the input will be the DFT coefficients while

the output is the reconstructed signal vector in time domain. The function carries out forward DFT when the argument $\text{inv}=0$, or inverse DFT when $\text{inv}=1$. The main body of the function is composed of an outer loop of size $\log_2 N$, the total number of stages, and an inner loop of size N for the computation for each stage. The computational complexity is therefore $O(N \log_2 N)$.

```

void fft(xr,xi,N,inv)
    float *xr,*xi;           // real and imaginary parts of data
    int N;                   // size of data
    int inv;                 // inv=0 for FFT, inv=1 for IFFT
{ int i,i1,j,k,l,ln,n,m;
    float arg,s,c,w,tmpc,tmpi;
    ln=log2f((float)N);
    for (i=0; i<N; ++i) {      // for all N elements of data
        j=0;
        for (k=0; k<ln; ++k)
            j=(j<<1) | (1&(i>>k)); // bit reversal
        if (j < i) {             // swap x[i] and x[j]
            w=xr[i]; xr[i]=xr[j]; xr[j]=w;
            w=xi[i]; xi[i]=xi[j]; xi[j]=w;
        }
    }
    for (i=0; i<ln; i++) {      // for log2(N) stages
        n=pow(2.0,(float)i);   // section size in current stage
        w=-Pi/n;
        if (inv) w=-w;          // no minus sign for inverse DFT
        k=0;
        while (k<N-1) {         // for N elements in a stage
            for (j=0; j<n; j++) { // for all points in each section
                l=k+j;
                c=cos(j*w); s=sin(j*w);
                tmpc=xr[l+n]*c-xi[l+n]*s;
                tmpi=xi[l+n]*c+xr[l+n]*s;
                xr[l+n]=xr[l]-tmpc;
                xi[l+n]=xi[l]-tmpi;
                xr[l]=xr[l]+tmpc;
                xi[l]=xi[l]+tmpi;
            }
            k=k+2*n;              // move on to next section
        }
    }
    arg=1.0/sqrt((float)N);
    for (i=0; i<N; i++)
        { xr[i]*=arg; xi[i]*=arg; }
}

```

}

The computational complexity for DFT can be further reduced if the signal is real, in which case the imaginary part of the signal is zero $x_r[n] = 0$ in time domain and the real part of the spectrum is even $X_r[-k] = X_r[k]$ while the imaginary part is odd $X_j[-k] = -X_j[k]$ in frequency domain. This 50% redundancy in either time or frequency domain can be avoided to reduce the complexity by half.

Also, if two real signal vectors $x_1[n]$ and $x_2[n]$ ($n = 0, \dots, N-1$) need to be transformed by one DFT:

1. Construct a complex vector composed of $x_1[n]$ as its real part and $x_2[n]$ as its imaginary part:

$$x[n] = x_1[n] + j x_2[n], \quad (n = 0, \dots, N-1) \quad (4.184)$$

2. Obtain the DFT of $x[n]$:

$$X[k] = \mathcal{F}[x[n]] = X_r[k] + j X_j[k], \quad (k = 0, \dots, N-1) \quad (4.185)$$

3. Obtain $\mathcal{F}[x_1[n]] = X_1[k] = X_{1r}[k] + j X_{1j}[k]$.

As $x_1[n]$ is real, the real part of its spectrum $X_{1r}[k]$ is even and the imaginary part $X_{1j}[k]$ is odd, i.e.,

$$X_1[k] = X_{1r}[k] + j X_{1j}[k] = \frac{X_r[k] + X_r[-k]}{2} + j \frac{X_j[k] - X_j[-k]}{2} \quad (4.186)$$

The two fractions extract respectively the even component of $X_r[k]$ and the odd component of $X_j[k]$.

4. Obtain $\mathcal{F}[x_2[n]] = X_2[k] = X_{2r}[k] + j X_{2j}[k]$.

As $j x_2[n]$ is imaginary, the real part of its spectrum $j X_{2r}[k]$ is odd and the imaginary part $j X_{2j}[k]$ is even, i.e.,

$$j X_2[k] = j X_{2r}[k] + j(j X_{2j}[k]) = \frac{X_r[k] - X_r[-k]}{2} + j \frac{X_j[k] + X_j[-k]}{2} \quad (4.187)$$

The two fractions extract respectively the odd component of $X_r[k]$ and the even component of $X_j[k]$. Dividing both sides by j , we get the spectrum $X_2[k]$ of real signal $x_2[n]$:

$$X_2[k] = X_{2r}[k] + j X_{2j}[k] = \frac{X_j[k] + X_j[-k]}{2} - j \frac{X_r[k] - X_r[-k]}{2} \quad (4.188)$$

As we can now obtain the spectra of two signal vectors with the computation of only one, the complexity can be reduced by half.

4.2.5 Four different forms of Fourier transform

The various forms of the Fourier transform for different types of signals (periodic or non-periodic, continuous or discrete) discussed in the current and previous

chapters can be considered as the following four different variations of the most generic Fourier transform as shown below.

- **I. Non-periodic continuous signal, continuous, non-periodic spectrum**

This is the most generic form of the Fourier transform for any continuous and non-periodic signal $x(t)$, considered as a function in a function space spanned by a set of uncountably infinite basis functions $\phi_f(t) = e^{j2\pi ft}$ ($-\infty < f < \infty$) that are orthonormal according to Eq.1.28:

$$\langle \phi_f(t), \phi_{f'}(t) \rangle = \int_{-\infty}^{\infty} e^{j2\pi(f-f')t} dt = \delta(f - f') \quad (4.189)$$

The signal $x(t)$ can therefore be expressed as a linear combination (integral) of these uncountable basis functions as:

$$x(t) = \int_{-\infty}^{\infty} X(f) \phi_f(t) df = \int_{-\infty}^{\infty} X(f) e^{j2\pi ft} df \quad (4.190)$$

This is the inverse transform and the coefficient function $X(f)$ can be obtained as the projection of the signal onto each of the basis functions:

$$X(f) = \langle x(t), \phi_f(t) \rangle = \langle x(t), e^{j2\pi ft} \rangle = \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt \quad (4.191)$$

This is the forward transform.

- **II. Periodic continuous signal, discrete non-periodic spectrum**

This is the Fourier series expansion of a continuous and periodic signal $x_T(t + T) = x_T(t)$, considered as a vector in the space of periodic functions spanned by a set of countable basis functions $\phi_k(t) = e^{j2\pi kt/T} / \sqrt{T}$ (for all integer k) that are orthonormal according to Eq.1.33:

$$\langle \phi_k(t), \phi_l(t) \rangle = \int_T e^{j2\pi(k-l)t/T} dt = \delta[k - l] \quad (4.192)$$

The signal $x_T(t)$ can be therefore expressed as a linear combination (summation) of these basis functions as:

$$x_T(t) = \sum_{k=-\infty}^{\infty} X[k] \phi_k(t) = \frac{1}{\sqrt{T}} \sum_{k=-\infty}^{\infty} X[k] e^{j2\pi kt/T} \quad (4.193)$$

This is the inverse transform and the coefficient $X[k]$ can be obtained as the projection of the signal onto the nth basis function:

$$X[k] = \langle x_T(t), \phi_k(t) \rangle = \langle x_T(t), \frac{1}{\sqrt{T}} e^{j2\pi kt/T} \rangle = \frac{1}{\sqrt{T}} \int_T x_T(t) e^{-j2\pi kt/T} dt \quad (4.194)$$

This is the forward transform. These Fourier expansion coefficients for a periodic signal can be considered as the samples of a continuous spectrum:

$$X(f) = \sum_{k=-\infty}^{\infty} X[k] \delta(f - kf_0) \quad (4.195)$$

where any two consecutive frequency components are separated by $f_0 = 1/T$.

- **III. Non-periodic discrete signal, continuous periodic spectrum**

This is the discrete-time Fourier transform of a discrete and non-periodic signal

$$x(t) = \sum_{n=-\infty}^{\infty} x[n] \delta(t - nt_0) \quad (4.196)$$

The sequence of signal samples $x[n]$ (for all integer n) form an infinite dimensional vector $\mathbf{x} = [\dots, x[n], \dots]^T$ in the vector space of all such vectors spanned by an uncountably infinite set of basis vectors $\phi_f = [\dots, e^{j2\pi nf/F}/\sqrt{F}, \dots]^T$ ($0 < f < F$) that are orthonormal according to Eq.1.35:

$$\langle \phi_f, \phi_{f'} \rangle = \frac{1}{F} \sum_{n=-\infty}^{\infty} e^{j2\pi n(f-f')} = \sum_{k=-\infty}^{\infty} \delta(f - f' - kF) \quad (4.197)$$

The signal \mathbf{x} can therefore be expressed as a linear combination (integral) of these uncountable basis vectors as:

$$\mathbf{x} = \int_F X(f) \phi_f df \quad (4.198)$$

or in component form:

$$x[n] = \frac{1}{\sqrt{F}} \int_F X(f) e^{j2\pi nf/F} df \quad (4.199)$$

This is the inverse transform, and the coefficient function $X(f)$ can be obtained as the projection of the signal onto each basis function:

$$X(f) = \langle \mathbf{x}, \phi_f \rangle = \frac{1}{\sqrt{F}} \sum_{n=-\infty}^{\infty} x[n] e^{-j2\pi nf/F} \quad (4.200)$$

This is the forward transform. Here $X(f + F) = X(f)$ is periodic.

- **IV. Periodic discrete signal, discrete periodic spectrum**

This is the discrete Fourier transform (DFT) of a discrete and periodic signal $\mathbf{x} = [x[0], \dots, x[N-1]]^T$, which is an N-D vector in a N-D unitary space spanned by a set of N N-D vectors $\phi_k = [e^{j2\pi 0k/N}, \dots, e^{j2\pi (N-1)k/N}]^T / \sqrt{N}$ that are orthonormal according to Eq.1.40:

$$\langle \phi_k, \phi_l \rangle = \frac{1}{N} \sum_{n=0}^{N-1} e^{j2\pi n(k-l)/N} = \sum_{n=-\infty}^{\infty} \delta[k - l - nN] \quad (4.201)$$

The signal vector can therefore be expressed as a linear combination (summation) of the N basis vectors:

$$\mathbf{x} = \sum_{k=0}^{N-1} X[k] \phi_k \quad (4.202)$$

or in component form:

$$x[n] = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X[k] e^{j2\pi nk/N}, \quad (n = 0, 1, \dots, N-1) \quad (4.203)$$

This is the inverse transform, and the weighting coefficient $X[k]$ can be obtained as the projection of the signal onto each basis function:

$$X[k] = \langle \mathbf{x}, \phi_k \rangle = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x[n] e^{-j2\pi nk/N}, \quad (k = 0, 1, \dots, N-1) \quad (4.204)$$

Here the discrete signal $x[n]$ are the samples of a continuous function:

$$x_T(t) = \sum_{n=0}^{N-1} x[n] \delta(t - nt_0) \quad (4.205)$$

and similarly the frequency coefficients can be considered as the samples of a continuous spectrum:

$$X_F(f) = \sum_{k=0}^{N-1} X[k] \delta(f - kf_0) \quad (4.206)$$

The four forms of Fourier transform can be summarized as below (where $T = 1/f_0$, $F = 1/t_0$, $T/t_0 = F/f_0 = N$):

	Signal $x(t)$	Spectrum $X(f)$
I	Continuous, Non-periodic $x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi ft} df$	Non-periodic, Continuous $X(f) = \int_{-\infty}^{\infty} x(t) e^{-j2\pi ft} dt$
II	Continuous, Periodic (T) $x_T(t) = \sum_{k=-\infty}^{\infty} X[k] e^{j2\pi kf_0 t}$	Non-periodic, Discrete (f) $X[k] = \int_T x_T(t) e^{-j2\pi kf_0 t} dt / T$ $X(f) = \sum_{k=-\infty}^{\infty} X[k] \delta(f - kf_0)$
III	Discrete (t_0), Non-periodic $x(t) = \sum_{n=-\infty}^{\infty} x[n] \delta(t - nt_0)$ $x[n] = \int_F X_F(f) e^{j2\pi f n t_0} df / F$	Periodic (F), Continuous $X_F(f) = \sum_{n=-\infty}^{\infty} x[n] e^{-j2\pi f n t_0}$
IV	Discrete (t_0), Periodic (T) $x[n] = \sum_{k=0}^{N-1} X[k] e^{j2\pi nk/N}$ $x(t) = \sum_{n=0}^{N-1} x[n] \delta(t - nt_0)$ $T/t_0 = N$	Periodic (F), Discrete (f_0) $X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi nk/N}$ $X(f) = \sum_{k=0}^{N-1} X[k] \delta(f - kf_0)$ $F/f_0 = T/t_0 = N$

Note in particular the relationship between time and frequency domains: the continuity and discreteness in one domain correspond to, respectively, the non-periodicity and periodicity in the other.

All four forms of the Fourier transform share the same set of properties, discussed most thoroughly for the continuous and non-periodic case, although they may take different forms for each of the four cases.

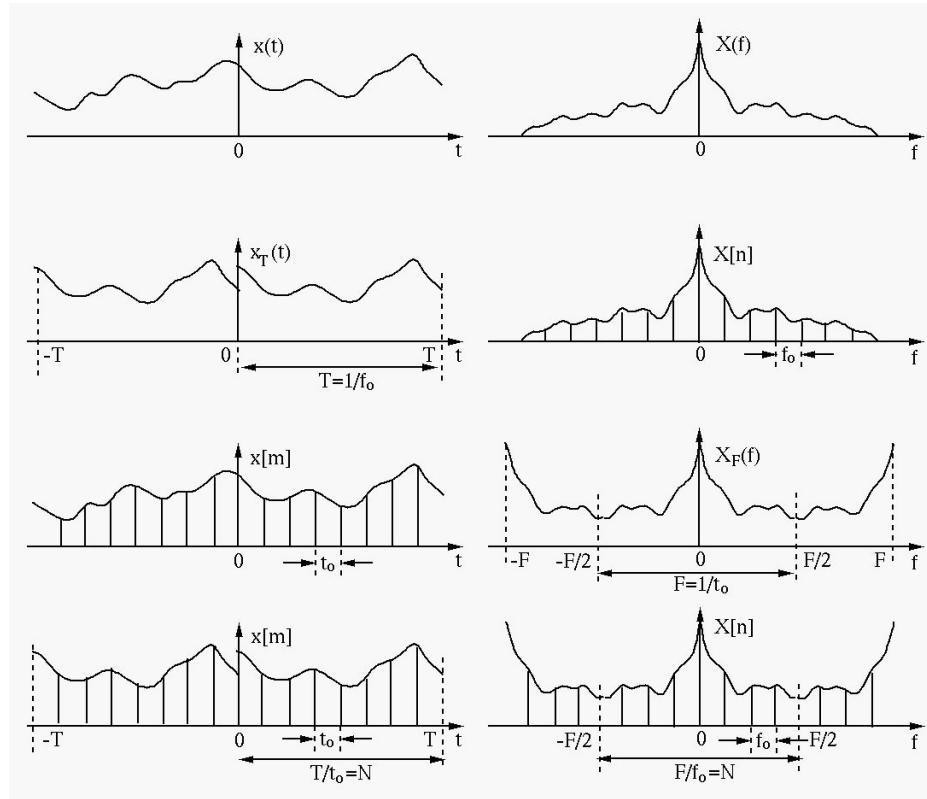


Figure 4.21 Four different forms of Fourier transform

Various types of time signals (continuous/discrete, periodic/non-periodic) on the left and their spectra on the right.

4.3 Two-Dimensional Fourier Transform

4.3.1 Two-Dimensional Signals and Their Spectra

All signals considered so far are assumed to be 1-D time functions. However, a signal could also be a function over a 1-D space, with the spatial frequency defined as the number of cycles in unit length (distance), instead of in unit time. Moreover, the concept of frequency analysis can be extended to various signals in 2 or 3-D spaces. For example, an image can be considered as a 2-D signal, and computer image processing has been a very active field of study for several decades with a wide variety of applications. Like in 1-D case, the Fourier transform is also a powerful tool in two or higher dimensional signals processing and analysis. We will consider the Fourier transform of some generic 2-D continuous signal denoted by $f(x, y)$, with x and y for the two spatial dimensions.

The Fourier transform of a 2-D signal $f(x, y)$ is defined as:

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi(ux+vy)} dx dy \quad (4.207)$$

This is the forward transform where u and v represent two spatial frequencies (cycles per unit distance) along two perpendicular directions of x and y in the 2-D space, respectively. The signal can be reconstructed by the inverse transform:

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) e^{j2\pi(ux+vy)} du dv \quad (4.208)$$

by which the signal is expressed as a linear combination of infinite set of uncountable 2-D orthogonal basis functions $\phi_{u,v}(x, y) = e^{j2\pi(ux+vy)}$, weighted by the Fourier coefficient function $F(u, v)$, the 2-D spectrum of the signal.

In the following discussion, we will always assume $f(x, y) = \bar{f}(x, y)$ is a real 2-D signal. The integrand in the 2-D Fourier transform is the product of two functions, the kernel function $\phi_{u,v}(x, y) = e^{j2\pi(ux+vy)}$ of the integral transform, the orthogonal basis functions, and the spectrum $F(u, v)$, the weighting function for the basis. Below we consider each of them separately.

- First, we consider the basis function $e^{j2\pi(ux+vy)}$.

We define two vectors, one in spatial domain, another in spatial frequency domain:

- \mathbf{r} is a vector associated with each point (x, y) in 2-D spatial domain:

$$\mathbf{r} = [x, y]^T \quad (4.209)$$

- \mathbf{w} is a vector associated with each point (u, v) in 2-D frequency domain:

$$\mathbf{w} = [u, v]^T = w[u/w, v/w]^T = w\mathbf{n} \quad (4.210)$$

where $w = \sqrt{u^2 + v^2}$ is the magnitude and $\mathbf{n} = [u/w, v/w]^T$ is the unit vector ($\|\mathbf{n}\| = 1$) along the direction of \mathbf{w} .

The inner product $\langle \mathbf{r}, \mathbf{n} \rangle = \mathbf{r}^T \mathbf{n} = xu + yv$ is the projection of a spatial point \mathbf{r} onto the direction of \mathbf{n} , and the 2-D basis function $\phi_{u,v}(x, y)$ can be written as:

$$\begin{aligned} \phi_{u,v}(x, y) &= e^{j2\pi(xu+yv)} = e^{j2\pi w \langle \mathbf{r}, \mathbf{n} \rangle} \\ &= \cos(2\pi w \langle \mathbf{r}, \mathbf{n} \rangle) + j \sin(2\pi w \langle \mathbf{r}, \mathbf{n} \rangle) \end{aligned} \quad (4.211)$$

As all spatial points $\mathbf{r} = (x, y)$ along a straight line perpendicular to the direction \mathbf{n} have the same projection $\langle \mathbf{r}, \mathbf{n} \rangle$, the function $\cos(2\pi w \langle \mathbf{r}, \mathbf{n} \rangle)$ takes the same value along the straight line, i.e., it is a planar sinusoid with frequency $w = \sqrt{u^2 + v^2}$ along the direction \mathbf{n} , at an angle $\theta = \tan^{-1}(v/u)$ from the positive direction of u . The same is true for the sine function of the imaginary part $\sin(2\pi w \langle \mathbf{r}, \mathbf{n} \rangle)$. For example, two 2-D sinusoidal functions $\cos(2\pi(3x + 2y))$ and $\cos(2\pi(2x + 3y))$ are shown in Fig.4.22.

- Second, we consider the weighting function $F(u, v)$.

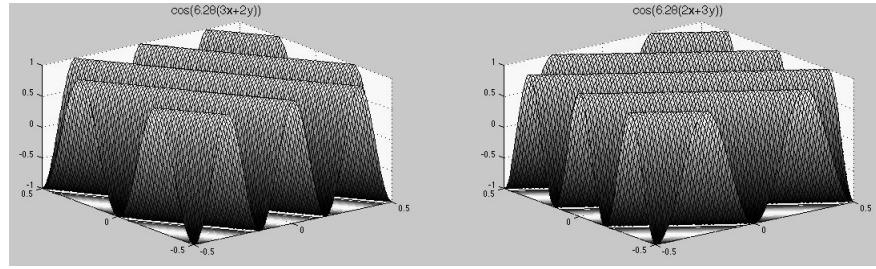


Figure 4.22 Different propagation directions of 2-D sinusoid $\cos(2\pi(ux + vy))$

In the plot on the left for $\cos(2\pi(3x + 2y))$, we see $u = 3$ cycles per unit length along x dimension (right side of plot) and $v = 2$ per unit length along y . In the plot on the right for $\cos(2\pi(2x + 3y))$, we see $u = 2$ cycles per unit length along x and $v = 3$ along y .

As the signal $f(x, y)$ is assumed real, its Fourier coefficient $F(u, v)$ can be written as below in terms of the real and imaginary parts:

$$\begin{aligned} F(u, v) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi(xu+yv)} dx dy \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \cos(2\pi(xu + yv)) dx dy \\ &\quad - j \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \sin(2\pi(xu + yv)) dx dy \\ &= F_r(u, v) + jF_j(u, v) = |F(u, v)| e^{j\angle F(u, v)} \end{aligned} \quad (4.212)$$

where $F_r(u, v)$ and $F_j(u, v)$ are respectively the real and imaginary parts:

$$\begin{cases} F_r(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \cos(2\pi(xu + yv)) dx dy \\ F_j(u, v) = - \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \sin(2\pi(xu + yv)) dx dy \end{cases} \quad (4.213)$$

and $|F(u, v)|$ and $\angle F(u, v)$ are respectively the amplitude and phase of $F(u, v)$:

$$\begin{cases} |F(u, v)| = \sqrt{F_r^2(u, v) + F_j^2(u, v)} \\ \angle F(u, v) = \tan^{-1}[F_j(u, v)/F_r(u, v)] \end{cases}, \quad \begin{cases} F_r(u, v) = |F(u, v)| \cos \angle F(u, v) \\ F_j(u, v) = |F(u, v)| \sin \angle F(u, v) \end{cases} \quad (4.214)$$

Note that $F_r(u, v)$ is even and $F_j(u, v)$ is odd:

$$\begin{cases} F_r(-u, -v) = F_r(u, v) \\ F_r(u, -v) = F_r(-u, v) \end{cases}, \quad \begin{cases} F_j(-u, -v) = -F_j(u, v) \\ F_j(u, -v) = -F_j(-u, v) \end{cases} \quad (4.215)$$

and $|F(u, v)|$ is even and $\angle F(u, v)$ is odd:

$$\begin{cases} |F(-u, -v)| = |F(u, v)| \\ |F(u, -v)| = |F(-u, v)| \end{cases}, \quad \begin{cases} \angle F(-u, -v) = -\angle F(u, v) \\ \angle F(u, -v) = -\angle F(-u, v) \end{cases} \quad (4.216)$$

Now combining the two aspects considered above, we can rewrite the inverse 2-D Fourier transform as:

$$\begin{aligned}
 f(x, y) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |F(u, v)| e^{j\angle F(u, v)} e^{j2\pi(ux+vy)} du dv \\
 &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |F(u, v)| \cos(2\pi(ux + vy) + \angle F(u, v)) du dv \\
 &\quad + j \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |F(u, v)| \sin(2\pi(ux + vy) + \angle F(u, v)) du dv \\
 &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} |F(u, v)| \cos(2\pi w \langle \mathbf{r}, \mathbf{n} \rangle + \angle F(u, v)) du dv \quad (4.217)
 \end{aligned}$$

Note the imaginary part is dropped as $f(x, y)$ is real. We now see that $f(x, y)$ is a superposition of uncountably infinite 2-D spatial sinusoids $|F(u, v)| \cos(2\pi w \langle \mathbf{r}, \mathbf{n} \rangle + \angle F(u, v))$ with

- **frequency** $w = \sqrt{u^2 + v^2}$ and **direction** \mathbf{n} (with angle $\theta = \tan^{-1}(v/u)$ from the positive direction of u in 2-D spatial frequency domain), both determined by u and v , and
- **amplitude** $|F(u, v)| = \sqrt{F_r(u, v)^2 + F_j(u, v)^2}$ and **phase** $\angle F(u, v) = \tan^{-1}(F_j(u, v)/F_r(u, v))$, both determined by $F(u, v)$.

Moreover, as $|H(u, v)|$ is even and $\angle H(u, v)$ is odd, Eq. 4.217 can be further rewritten as:

$$\begin{aligned}
 f(x, y) &= 2 \int_0^{\infty} \int_0^{\infty} |F(u, v)| \cos(2\pi(ux + vy) + \angle F(u, v)) du dv \\
 &\quad + 2 \int_{-\infty}^0 \int_0^{\infty} |F(u, v)| \cos(2\pi(ux - vy) + \angle F(u - v)) du dv \\
 &= 2 \int_0^{\infty} \int_0^{\infty} |F(u, v)| \cos(2\pi w \langle \mathbf{r}, \mathbf{n} \rangle + \angle F(u, v)) du dv \\
 &\quad + 2 \int_{-\infty}^0 \int_0^{\infty} |F(u, v)| \cos(2\pi w \langle \mathbf{r}, \mathbf{n}' \rangle + \angle F(u - v)) du dv \quad (4.218)
 \end{aligned}$$

where \mathbf{n}' is the unit vector in the direction determined by the angle $(\tan^{-1}(-v/u) = -\tan^{-1}(v/u) = -\theta)$. This equation is the 2-D version of Eq. 3.134. The first integral represents superposition of sinusoids in the directions $0 < \theta < 90^\circ$ (NE to SW), while the second integral represents a superposition of sinusoids in the directions $0 > \theta > -90^\circ$ (NW to SE).

The DFT of a 2-D discrete and periodic signal $x[m, n]$ can be similarly considered. First write the 2-D DFT coefficients in polar form:

$$X[k, l] = |X[k, l]| e^{j\angle X[k, l]} \quad (4.219)$$

where

$$\begin{cases} |X[k, l]| = \sqrt{X_r^2[k, l] + X_j^2[k, l]} \\ \angle X[k, l] = \tan^{-1}[X_j[k, l]/X_r[k, l]] \end{cases} \quad \begin{cases} X_r[k, l] = |X[k, l]| \cos(\angle X[k, l]) \\ X_j[k, l] = |X[k, l]| \sin(\angle X[k, l]) \end{cases} \quad (4.220)$$

Then the 2-D signal $x[m, n]$ can be represented as a superposition of a set of planar sinusoids with different frequencies, directions, amplitudes and phase shifts:

$$\begin{aligned} x[m, n] &= \frac{1}{\sqrt{MN}} \sum_{l=0}^{N-1} \sum_{k=0}^{M-1} X[k, l] \exp[j2\pi(\frac{mk}{M} + \frac{nl}{N})] \\ &= \frac{1}{\sqrt{MN}} \sum_{-M/2+1}^{M/2} \sum_{-N/2+1}^{N/2} |X[k, l]| \cos(2\pi(\frac{mk}{M} + \frac{nl}{N}) + \angle X[k, l]), \\ &\quad (m = 0, \dots, M-1, n = 0, \dots, N-1) \end{aligned} \quad (4.221)$$

4.3.2 Fourier Transform of Typical 2-D Functions

- Planar sinusoidal wave:

$$f(x, y) = \cos(2\pi(3x - 2y)) = \frac{1}{2}[e^{j2\pi(3x-2y)} + e^{-j2\pi(3x-2y)}] \quad (4.222)$$

This is a planar sinusoid of spatial frequency $\sqrt{3^2 + 2^2} = \sqrt{13}$ in the direction of $\theta = \tan^{-1}(-2/3)$ with unit amplitude and zero phase. Its 2-D Fourier spectrum is:

$$\begin{aligned} F(u, v) &= \int \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi(ux+vy)} dx dy \\ &= \frac{1}{2} \int \int_{-\infty}^{\infty} [e^{j2\pi(3x-2y)} + e^{-j2\pi(3x-2y)}] e^{-j2\pi(ux+vy)} dx dy \\ &= \frac{1}{2} \int \int_{-\infty}^{\infty} e^{-j2\pi((u-3)x+(v+2)y)} dx dy + \frac{1}{2} \int \int_{-\infty}^{\infty} e^{-j2\pi((u+3)x+(v-2)y)} dx dy \\ &= \frac{1}{2} \int_{-\infty}^{\infty} e^{-j2\pi(u-3)x} dx \int_{-\infty}^{\infty} e^{-j2\pi(v+2)y} dy \\ &\quad + \frac{1}{2} \int_{-\infty}^{\infty} e^{-j2\pi(u+3)x} dx \int_{-\infty}^{\infty} e^{-j2\pi(v-2)y} dy \\ &= \frac{1}{2} [\delta(u-3)\delta(v+2) + \delta(u+3)\delta(v-2)] \end{aligned} \quad (4.223)$$

This transform pair is shown in Fig.4.23(a).

- Superposition of three planar sinusoidal waves:

$$f(x, y) = 3 \cos(2\pi 2x) + 2 \cos(2\pi 3y) + \cos(2\pi 5(x-y)); \quad (4.224)$$

Its 2-D Fourier spectrum is:

$$\begin{aligned} F(u, v) &= \int \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi(ux+vy)} dx dy \\ &= \frac{3}{2} [\delta(u-2) + \delta(u+2)] \delta(v) + \delta(u) [\delta(v-3) + \delta(v+3)] \\ &\quad + \frac{1}{2} [\delta(u-5)\delta(v+5) + \delta(u+5)\delta(v-5)] \end{aligned} \quad (4.225)$$

This transform pair is shown in Fig.4.23(b).

- Rectangular impulse in 2-D space:

$$f(x, y) = \begin{cases} 1 & -\frac{a}{2} < x < \frac{a}{2}, -\frac{b}{2} < y < \frac{b}{2} \\ 0 & \text{else} \end{cases} \quad (4.226)$$

This 2-D function is separable as it can be written as the product of two 1-D functions $f(x, y) = f_x(x)f_y(y)$, where $f_x(x)$ and $f_y(y)$ are each a 1-D square impulse function. The spectrum is the product of the spectra $F_x(u) = \mathcal{F}[f_x(x)]$ and $F_y(v) = \mathcal{F}[f_y(y)]$, a 2-D sinc function.

$$\begin{aligned} F(u, v) &= \int \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi(ux+vy)} dx dy \\ &= \int_{-\infty}^{\infty} f_x(x) e^{-j2\pi ux} dx \int_{-\infty}^{\infty} f_y(y) e^{-j2\pi vy} dy \\ &= \int_{-a/2}^{a/2} e^{-j2\pi ux} dx \int_{-b/2}^{b/2} e^{-j2\pi vy} dy = \frac{\sin(\pi ua)}{\pi u} \frac{\sin(\pi vb)}{\pi v} \end{aligned} \quad (4.227)$$

This transform pair is shown in Fig.4.23(c).

- Cylindrical impulse:

$$f(x, y) = \begin{cases} 1 & x^2 + y^2 < R^2 \\ 0 & \text{else} \end{cases} \quad (4.228)$$

As $f(x, y)$ is not separable but central symmetric, it is more convenient to use polar coordinate system in both spatial and frequency domains. We let

$$\begin{cases} x = r \cos \theta, & r = \sqrt{x^2 + y^2} \\ y = r \sin \theta & \theta = \tan^{-1}(y/x) \end{cases} \quad (4.229)$$

$$dx dy = rdr d\theta \quad (4.230)$$

and

$$\begin{cases} u = \rho \cos \phi, & \rho = \sqrt{u^2 + v^2} \\ v = \rho \sin \phi, & \phi = \tan^{-1}(v/u) \end{cases} \quad (4.231)$$

$$du dv = \rho d\rho d\phi \quad (4.232)$$

then we have:

$$\begin{aligned}
 F(u, v) &= \int \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi(ux+vy)} dx dy \\
 &= \int_0^R \left[\int_0^{2\pi} e^{-j2\pi r\rho(\cos\theta\cos\phi+\sin\theta\sin\phi)} d\theta \right] r dr \\
 &= \int_0^R \left[\int_0^{2\pi} e^{-j2\pi r\rho\cos(\theta-\phi)} d\theta \right] r dr = \int_0^R \left[\int_0^{2\pi} e^{-j2\pi r\rho\cos\theta} d\theta \right] r dr \quad (4.233)
 \end{aligned}$$

To continue, we need to use the 0th order Bessel function $J_0(x)$ defined as

$$J_0(x) = \frac{1}{2\pi} \int_0^{2\pi} e^{-jx\cos\theta} d\theta \quad (4.234)$$

which is related to the 1st order Bessel function $J_1(x)$ by

$$\frac{d}{dx}(x J_1(x)) = x J_0(x) \quad (4.235)$$

i.e.

$$\int_0^x x J_0(x) dx = x J_1(x) \quad (4.236)$$

Substituting $2\pi r\rho$ for x , we have

$$F(u, v) = F(\rho, \phi) = \int_0^R 2\pi r J_0(2\pi r\rho) dr = \frac{1}{\rho} R J_1(2\pi\rho R) \quad (4.237)$$

We see that the spectrum $F(u, v) = F(\rho, \phi)$ is independent of angle ϕ and therefore is central symmetric sinc-like function.

- Ideal low-pass filter:

$$F(u, v) = \begin{cases} 1 & u^2 + v^2 < R^2 \\ 0 & \text{else} \end{cases} \quad (4.238)$$

This cylindrical impulse in frequency domain is called an ideal low-pass filter. When the spectrum of any given 2-D signal is multiplied by the ideal filter, all of its low frequency components inside the radius R are kept, while all higher frequency components outside the circle are suppressed to zero.

Due to the symmetry property of the Fourier transform, the inverse transform of this ideal low-pass filter is the same 2-D sinc-like function shown in Eq.4.237 in spatial domain, as shown in Fig.4.23(d).

- Gaussian function in 2-D space:

$$f(x, y) = \frac{1}{a^2} e^{-\pi(x^2+y^2)/a^2} = \frac{1}{a} e^{-\pi(x/a)^2} \frac{1}{a} e^{-\pi(y/a)^2} \quad (4.239)$$

The spectrum of this function can be found as:

$$\begin{aligned}
 F(u, v) &= \int \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi(ux+vy)} dx dy \\
 &= \frac{1}{a} \int_{-\infty}^{\infty} e^{-\pi(x/a)^2} e^{-j2\pi ux} dx \quad \frac{1}{a} \int_{-\infty}^{\infty} e^{-\pi(y/a)^2} e^{-j2\pi vy} dy \\
 &= e^{-\pi(au)^2} e^{-\pi(av)^2}
 \end{aligned} \tag{4.240}$$

The last equation is due to Eq.3.161. Now we see that the Fourier transform of a 2-D Gaussian function is also a Gaussian, the product of two 1-D Gaussian functions along directions of u and v , respectively, as shown in Fig.4.23(e).

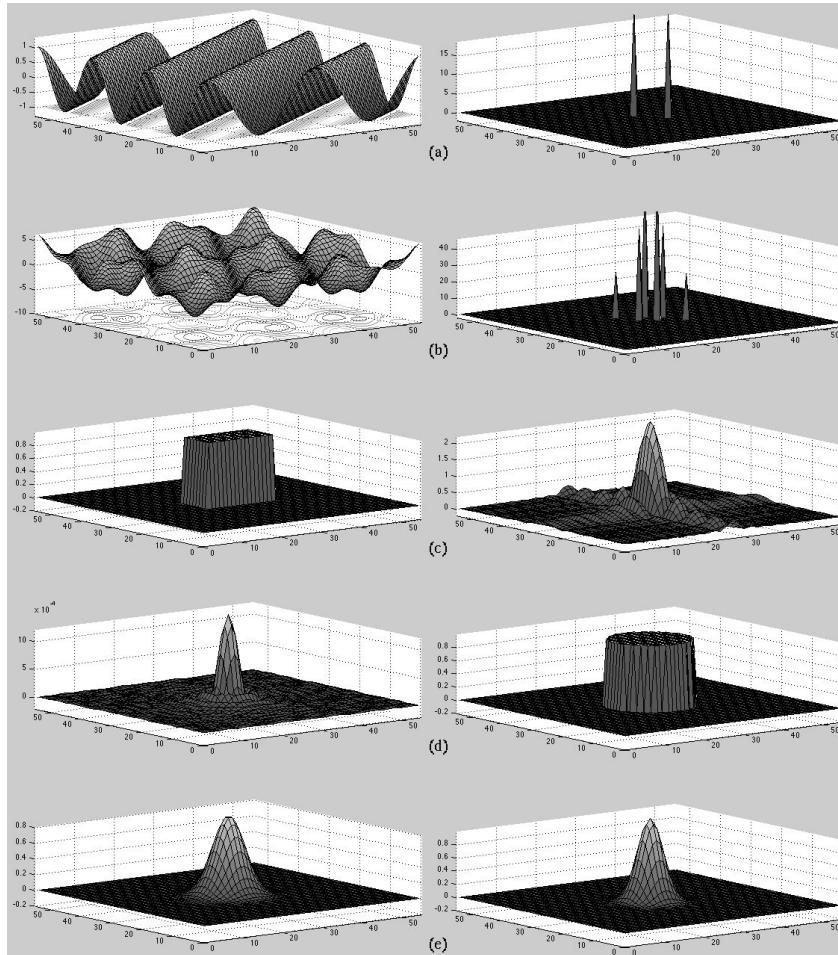


Figure 4.23 Some 2-D signals (left) and their spectra (right)

4.3.3 Four Forms of 2-D Fourier Transform

Same as in the 1-D case, there also exist four different forms of 2-D Fourier transform, depending on whether the given 2-D signal $f(x, y)$ is periodic or non-periodic, and whether it is discrete or continuous.

- **Non-periodic continuous signal, continuous non-periodic spectrum**

$$F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi(ux+vy)} dx dy \quad (4.241)$$

$$f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) e^{j2\pi(ux+vy)} du dv \quad (4.242)$$

This is the most generic 2-D Fourier transform pair from Eqs.4.207 and 4.207.

- **Non-periodic discrete signal, continuous periodic spectrum**

The spatial signal $f[m, n]$ is discrete with spatial intervals x_o and y_o between consecutive signal samples in the x and y directions, respectively.

$$F_{UV}(u, v) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f[m, n] e^{-j2\pi(umx_o+vny_o)} \quad (4.243)$$

$$f[m, n] = \frac{1}{UV} \int_0^U \int_0^V F(u, v) e^{j2\pi(umx_o+vny_o)} du dv \quad (4.244)$$

The 2-D spectrum $F_{UV}(u, v) = F(u + U, v + V)$ is periodic with periods (the sampling frequencies) $U = 1/x_o$ and $V = 1/y_o$ in the two directions.

- **Periodic continuous signal, discrete non-periodic spectrum**

The spatial signal $f_{XY}(x, y) = f_{XY}(x + X, y + Y)$ is periodic with periods X and Y in x and y directions of the 2-D space, respectively.

$$F[k, l] = \frac{1}{XY} \int_0^X \int_0^Y f_{XY}(x, y) e^{j2\pi(kxu_o+lvy_o)} dx dy \quad (4.245)$$

$$f_{XY}(x, y) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} F[k, l] e^{-j2\pi(xku_o+ylv_o)} \quad (4.246)$$

The 2-D spectrum is discrete with intervals $u_o = 1/X$ and $v_o = 1/Y$ between consecutive frequency components $F[k, l]$ in spatial frequency directions u and v , respectively.

- **Periodic discrete signal, discrete periodic spectrum**

This is the 2-D discrete Fourier transform (2-D DFT). The spatial signal is discrete with intervals x_0 and y_0 between consecutive samples in the x and y directions, respectively, and it is also periodic with period X and Y . The 2-D signal has $X/x_0 = M$ and $Y/y_0 = N$ samples along each of the two spatial directions and can be represented as an $M \times N$ array $x[m, n]$ ($m =$

$0, \dots, M-1, n=0, \dots, M-1$). The 2-D DFT pair is

$$\begin{aligned} X[k, l] &= \frac{1}{\sqrt{MN}} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} x[m, n] e^{-j2\pi(\frac{mk}{M} + \frac{nl}{N})} \\ x[m, n] &= \frac{1}{\sqrt{MN}} \sum_{l=0}^{N-1} \sum_{k=0}^{M-1} X[k, l] e^{j2\pi(\frac{mk}{M} + \frac{nl}{N})} \\ (0 \leq m, k \leq M-1, \quad 0 \leq n, l \leq N-1) \end{aligned} \quad (4.247)$$

The spectrum is both discrete and periodic with periods (sampling rates) $U = 1/x_0$ and $V = 1/y_0$ and intervals $u_0 = 1/X$ and $v_0 = 1/Y$ between consecutive frequency components $F[k, l]$ along u and v , respectively. The signal is periodic $x[m+M, n+N] = x[m, n]$, and so is its spectrum $X[k+M, l+N] = X[k, l]$.

Note that the kernel function of the 2-D Fourier transform is separable in the sense that it can be expressed as a product of two 1-D kernel functions in each of the two dimensions:

$$\phi_{u,v}(x, y) = e^{j2\pi(ux+vy)} = e^{j2\pi ux} e^{j2\pi vy} = \phi_u(x) \phi_v(y) \quad (4.248)$$

The 2-D transform can be carried out as:

$$\begin{aligned} F(u, v) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi ux} e^{-j2\pi vy} dx dy \\ &= \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} f(x, y) e^{-j2\pi ux} dx \right] e^{-j2\pi vy} dy \\ &= \int_{-\infty}^{\infty} F'(u, y) e^{-j2\pi vy} dy \end{aligned} \quad (4.249)$$

where $F'(u, y)$ is an intermediate result obtained by a 1-D transform in the dimension of x :

$$F'(u, y) = \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi ux} dx$$

and the 2-D spectrum $F(u, v)$ can be obtained by another 1-D transform in the dimension of y . In other words, the 2-D transform can be carried out in two steps each for one of the two dimensions. Obviously the order of the two steps can be reversed.

As in the case of 1-D Fourier transform, among all four forms of 2-D Fourier transform, only the discrete 2-D Fourier transform with finite and discrete signal samples and frequency components can be carried out numerically. Also, how the total scaling factor $1/MN$ is distributed between the forward and inverse transforms is of little significance.

4.3.4 Computation of the 2-D DFT

A 2-D discrete signal $x[m, n]$ ($m = 0, \dots, M-1$, $n = 0, \dots, N-1$) can be considered as an M by N matrix $\mathbf{x}_{M \times N} = [\mathbf{x}_0, \dots, \mathbf{x}_{N-1}]$ consisting of N M-D

column vectors \mathbf{x}_n ($n = 0, \dots, N - 1$) (or M N-D row vectors). Also, as the kernel function is separable, the 2-D DFT for this signal can be carried out in two steps: N 1-D DFTs of the N columns, followed by M 1-D DFTs of the M rows of the matrix obtained in step 1 (or in reverse order):

$$\begin{aligned} X[k, l] &= \frac{1}{\sqrt{MN}} \sum_{n=0}^{N-1} \left[\sum_{m=0}^{M-1} x[m, n] e^{-j2\pi \frac{mk}{M}} \right] e^{-j2\pi \frac{nl}{N}} \\ &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X'[k, n] e^{-j2\pi \frac{nl}{N}}, \quad (k = 0, \dots, M - 1, \quad l = 0, \dots, N - 1) \end{aligned} \quad (4.250)$$

where $X'[k, n]$ is an intermediate result obtained by column transforms in the first step:

$$X'[k, n] = \frac{1}{\sqrt{M}} \sum_{m=0}^{M-1} x[m, n] e^{-j2\pi \frac{mk}{M}} \quad (k = 0, \dots, M - 1, \quad n = 0, \dots, N - 1) \quad (4.251)$$

Here the summation is with respect to the row index m of $x[m, n]$ and the column index n is treated as a parameter. This 1-D DFT of the n th column of the 2-D signal matrix \mathbf{x} can be written in column vector (vertical) form as:

$$\mathbf{X}'_n = \overline{\mathbf{W}}_M \mathbf{x}_n, \quad (n = 0, \dots, N - 1) \quad (4.252)$$

where $\mathbf{X}'_n = [X[0, n], \dots, X[M - 1, n]]^T$ is an M -D vector and $\overline{\mathbf{W}}_M$ is a $M \times M$ symmetric Fourier transform matrix with the mn -th element $w[m, n] = e^{j2\pi mn/M}/\sqrt{M}$, similar to the matrix in Eq. 4.120. Putting all N columns together we have:

$$[\mathbf{X}'_0, \dots, \mathbf{X}'_{N-1}] = \mathbf{X}_{M \times N} = \overline{\mathbf{W}}_M [\mathbf{x}_0, \dots, \mathbf{x}_{N-1}] = \overline{\mathbf{W}}_M \mathbf{x}_{M \times N} \quad (4.253)$$

where we have defined $\mathbf{X}'_{M \times N} = [\mathbf{X}'_0, \dots, \mathbf{X}'_{N-1}]$.

In the second step, a 1-D DFT is carried out for each of the M rows of the intermediate result \mathbf{X}' :

$$\mathbf{X}_m^T = (\overline{\mathbf{W}}_N \mathbf{X}'_m)^T = \mathbf{X}'_m^T \overline{\mathbf{W}}_N^T = \mathbf{X}'_m^T \overline{\mathbf{W}}_N, \quad (m = 0, \dots, M - 1) \quad (4.254)$$

where \mathbf{X}_m^T is the m th row vector of the 2-D DFT matrix $\mathbf{X}_{M \times N}$, and $\overline{\mathbf{W}}_N$ is a $N \times N$ DFT matrix first given in Eq. 4.120. Putting all M rows together we have:

$$\mathbf{X}_{M \times N} = \begin{bmatrix} \mathbf{X}_0^T \\ \vdots \\ \mathbf{X}_{M-1}^T \end{bmatrix} = \begin{bmatrix} \mathbf{X}'_0^T \\ \vdots \\ \mathbf{X}'_{M-1}^T \end{bmatrix} \overline{\mathbf{W}}_N = \mathbf{X}'_{M \times N} \overline{\mathbf{W}}_N \quad (4.255)$$

Substituting Eq. 4.253 into this equation we get:

$$\mathbf{X}_{M \times N} = \mathbf{X}'_{M \times N} \overline{\mathbf{W}}_N = \overline{\mathbf{W}}_M \mathbf{x}_{M \times N} \overline{\mathbf{W}}_N \quad (4.256)$$

This is the 2-D DFT in matrix form. Similarly, the inverse 2-D DFT can be written as:

$$\mathbf{x}_{M \times N} = \mathbf{W}_M \mathbf{X}_{M \times N} \mathbf{W}_N \quad (4.257)$$

We can now rewrite these two equations as a 2-D DFT pair:

$$\begin{cases} \mathbf{X} = \overline{\mathbf{W}} \mathbf{x} \overline{\mathbf{W}} & \text{(forward)} \\ \mathbf{x} = \mathbf{W} \mathbf{X} \mathbf{W} & \text{(inverse)} \end{cases} \quad (4.258)$$

Here the subscripts of all matrices are dropped.

The DFT matrix \mathbf{W} can be expressed in terms of its rows as well as its columns, and the matrix form of the inverse transform can be expanded to become:

$$\begin{aligned} \mathbf{x} &= [\mathbf{w}_0, \dots, \mathbf{w}_{M-1}] \begin{bmatrix} X[0, 0] & \cdots & X[0, N-1] \\ \vdots & \ddots & \vdots \\ X[M-1, 0] & \cdots & X[M-1, N-1] \end{bmatrix} \begin{bmatrix} \mathbf{w}_0^T \\ \vdots \\ \mathbf{w}_{N-1}^T \end{bmatrix} \\ &= [\mathbf{w}_0, \dots, \mathbf{w}_{M-1}] \begin{bmatrix} \sum_{l=0}^{N-1} X[0, l] \mathbf{w}_l^T \\ \vdots \\ \sum_{l=0}^{N-1} X[M-1, l] \mathbf{w}_l^T \end{bmatrix} \\ &= \sum_{k=0}^{M-1} \mathbf{w}_k \sum_{l=0}^{N-1} X[k, l] \mathbf{w}_l^T = \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} X[k, l] \mathbf{w}_k \mathbf{w}_l^T = \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} X[k, l] \mathbf{B}_{kl} \end{aligned} \quad (4.259)$$

where

$$\mathbf{B}_{kl} = \mathbf{w}_k \mathbf{w}_l^T = \frac{1}{\sqrt{MN}} \begin{bmatrix} \ddots & \cdots & \cdots \\ \vdots & e^{j2\pi(\frac{mk}{M} + \frac{nl}{N})} & \vdots \\ \cdots & \cdots & \ddots \end{bmatrix} \quad (4.260)$$

The result above indicates that the 2-D signal \mathbf{x} can be expressed as a linear combination of a set of MN 2-D (M by N) basis functions each weighted by coefficient $X[k, l]$ ($k = 0, \dots, M-1, l = 0, \dots, N-1$), which is given in the first equation of Eq.4.258 for the forward 2-D DFT:

$$\mathbf{X} = \begin{bmatrix} \overline{\mathbf{w}}_0^T \\ \vdots \\ \overline{\mathbf{w}}_{N-1}^T \end{bmatrix} \mathbf{x}[\overline{\mathbf{w}}_0, \dots, \overline{\mathbf{w}}_{N-1}] \quad (4.261)$$

as the kl-th coefficient:

$$\begin{aligned} X[k, l] &= \overline{\mathbf{w}}_k^T \begin{bmatrix} x[0, 0] & \cdots & x[0, N-1] \\ \vdots & \ddots & \vdots \\ x[M-1, 0] & \cdots & x[M-1, N-1] \end{bmatrix} \overline{\mathbf{w}}_l \\ &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m, n] \overline{B}_{kl}[m, n] = \langle \mathbf{x}, \mathbf{B}_{kl} \rangle \end{aligned} \quad (4.262)$$

This is the inner product of two 2-D matrices \mathbf{x} and the kl-th basis function \mathbf{B}_{kl} (Eq.2.16), representing the projection of the signal \mathbf{x} onto the kl-th basis function \mathbf{B}_{kl} .

The kl-th 2-D DFT basis function \mathbf{B}_{kl} can be found by letting all elements of the coefficient array in Eq.4.259 be zero except $X[k, l] = 1$. For example, when $M = N = 8$, the $M \times N = 64$ such 2-D basis functions are shown in Fig.4.24.

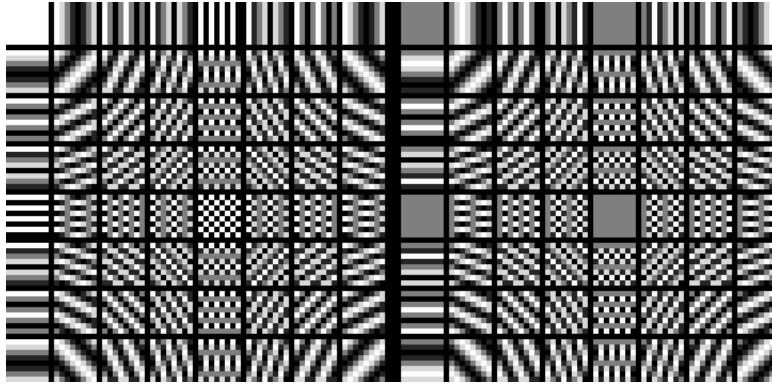


Figure 4.24 $8 \times 8 = 64$ 2-D DFT basis functions \mathbf{B}_{kl} , ($k, l = 0, \dots, 7$)

The left half of the image shows the real part of the 8 by 8 2-D basis functions, while the right half shows the imaginary part. The DC component is at the top-left corner of the real part, and the highest frequency component in both the horizontal and vertical directions is in the middle of the real part.

The C code for both the forward and inverse 2-D DFT is listed below:

```
fft2d(xxr,xxi,m,n,inverse)
    float **xxr, **xxi;
    int m,n,inverse;
{ float *xr, *xi;
    int i,j,k;
    k=m; if (n>m) k=n;
    xr = (float *) malloc(k*sizeof(float));
    xi = (float *) malloc(k*sizeof(float));
    for (j=0; j<n; j++) {      // for n column xforms
        for (i=0; i<m; i++)
```

```

{ xr[i]=xxr[i][j]; xi[i]=xxi[i][j]; }
fft(xr,xi,m,inverse);
for (i=0; i<m; i++)
{ xxr[i][j]=xr[i]; xxi[i][j]=xi[i]; }
}
for (i=0; i<m; i++) { // for m column xforms
    for (j=0; j<n; j++)
    { xr[j]=xxr[i][j]; xi[j]=xxi[i][j]; }
    fft(xr,xi,n,inverse);
    for (j=0; j<n; j++)
    { xxr[i][j]=xr[j]; xxi[i][j]=xi[j]; }
}
free(xr); free(xi);
}

```

Example 4.11: Consider the 2-D DFT of a real 8×8 2-D signal (imaginary part is zero):

$$\mathbf{x}_r = \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 70.0 & 80.0 & 90.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 90.0 & 100.0 & 110.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 110.0 & 120.0 & 130.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 130.0 & 140.0 & 150.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix} \quad (4.263)$$

The 8-point DFT matrix \mathbf{W}_8 is the same as the one shown in Eqs. 4.134 and 4.135. The real and imaginary parts of the 2-D DFT of this signal \mathbf{x} is $\mathbf{X} = \overline{\mathbf{W}}_8 \mathbf{x} \overline{\mathbf{W}}_8$, as below:

$$\mathbf{X}_r = \left[\begin{array}{c|cccc|ccccc} 165.0 & -98.9 & 10.0 & -21.1 & 55.0 & -21.1 & 10.0 & -98.9 \\ -63.1 & -11.3 & 27.7 & 13.2 & -21.0 & 1.6 & -32.7 & 85.7 \\ 15.0 & 0.0 & -5.0 & -2.9 & 5.0 & 0.0 & 5.0 & 17.1 \\ -41.9 & 16.8 & 2.7 & 6.3 & -14.0 & 4.3 & -7.7 & 33.4 \\ \hline 15.0 & -8.5 & 0.0 & -1.5 & 5.0 & -1.5 & 0.0 & -8.5 \\ -41.9 & 33.4 & -7.7 & 4.3 & -14.0 & 6.3 & 2.7 & 16.8 \\ 15.0 & -17.1 & 5.0 & 0.0 & 5.0 & -2.9 & -5.0 & 0.0 \\ -63.1 & 85.7 & -32.7 & 1.6 & -21.0 & 13.2 & 27.7 & -11.3 \end{array} \right] \quad (4.264)$$

and

$$\mathbf{X}_j = \begin{bmatrix} 0.0 & -88.9 & 55.0 & 11.1 & 0.0 & -11.1 & -55.0 & 88.9 \\ -90.5 & 89.2 & -27.1 & 6.9 & -30.2 & 16.8 & 15.0 & 19.9 \\ 15.0 & -17.1 & 5.0 & 0.0 & 5.0 & -2.9 & -5.0 & 0.0 \\ -15.5 & 31.9 & -15.0 & -0.8 & -5.2 & 4.9 & 12.9 & -13.2 \\ \hline 0.0 & -8.5 & 5.0 & 1.5 & 0.0 & -1.5 & -5.0 & -8.5 \\ 15.5 & 13.2 & -12.9 & -4.9 & 5.2 & 0.8 & 15.0 & -31.9 \\ -15.0 & 0.0 & 5.0 & 2.9 & -5.0 & 0.0 & -5.0 & 17.1 \\ 90.5 & -19.9 & -15.0 & -16.8 & 30.2 & -6.9 & 27.1 & -89.2 \end{bmatrix} \quad (4.265)$$

These $8 \times 8 = 64$ elements $X[k, l] = X_r[k, l] + j X_j[k, l]$ of \mathbf{X} are the complex coefficients for the amplitudes $|X[k, l]|$ and phase $\angle X[k, l]$ of the 64 2-D frequency components shown in Fig.4.24. Note that as the signal $x[m, n]$ is real, the real part of its spectrum is even: $X_r[k, l] = X_r[M - k, N - l]$, $X_r[k, N - l] = X_r[M - k, l]$, while the imaginary part is odd: $X_j[k, l] = -X_j[M - k, N - l]$, $X_j[k, N - l] = -X_j[M - k, l]$. Consider specifically the following coefficients:

- $X[0, 0]$ is the amplitude of the DC offset (average) of the signal.
- $X[0, N/2]$ is the amplitude of the highest frequency component $(-1)^n$ in horizontal direction.
- $X[M/2, 0]$ is the amplitude of the highest frequency component $(-1)^m$ in vertical direction.
- $X[M/2, N/2]$ is the amplitude of the highest frequency component $(-1)^{m+n}$ in both directions. The above four coefficients are real with zero phase.
- $X[0, l]$ pairs up with $X[0, N - l]$ ($l = 1, \dots, N/2 - 1$) to represent the amplitude and phase of a planar sinusoid $|X[0, l]| \cos(2\pi(nl/N)) + \angle X[0, l])$ in horizontal direction;
- $X[k, 0]$ pairs up with $X[M - k, 0]$ ($k = 1, \dots, M/2 - 1$) to represent the amplitude and phase of a planar sinusoid $|X[k, 0]| \cos(2\pi(mk/M)) + \angle X[k, 0])$ in vertical direction;

The coefficients in the rest of the array $X[k, l]$ can be divided into four quadrants with the top-left paired up with the low-right to represent sinusoids in NW-SE directions, while the top-right paired up with the low-right to represent sinusoids in NE-SW directions.

As shown in the example above, when the signal in spatial domain is real with its imaginary part $x_j[m, n] = 0$ equal to zero, half of the data points are redundant; correspondingly in spatial frequency domain, both the real and imaginary parts of $X[k, l]$ are symmetric (even or odd). More specifically, we note that the real part $X_r[k, l]$ has $MN/2 + 2$ independent variables, and the imaginary part $X_j[k, l]$ has $MN/2 - 2$ independent variables. Taking advantage of the symmetry property an algorithm can be designed to cut by half the computation for the 2-D DFT of real signals.

In the 2-D spectrum matrix, the DC component $X[0, 0]$ at zero frequency is at the upper-left corner, the low frequency components are around the edges, while the high frequency components are in the area around the center $(M/2, N/2)$, as shown in the example above. However, sometimes it is preferable to centralize the spectrum so that the DC component $X[0, 0]$ is in the middle, while the high frequency components are farther away from the center around the corners and edges, so that the 2-D spectrum is in consistent with the convention that the DC component at the origin is always in the center of the 2-D coordinate system of the frequency domain. Similar to the case of 1-D DFT discussed before, the centralization of the 2-D spectrum can be simply realized by shifting the 2-D spectrum in both dimensions by half of the corresponding length. Alternatively, based on the frequency shift property, the centralization can be equivalently realized in spatial domain by negating every other spatial samples, similar to the 1D case in Eq.4.172:

$$\begin{aligned}\mathcal{F}^{-1}[X[k - M/2, l - N/2]] &= x[m, n]e^{j2\pi(\frac{mM/2}{M} + \frac{nN/2}{N})} \\ &= x[m, n]e^{j\pi(m+n)} = x[m, n](-1)^{m+n}\end{aligned}\quad (4.266)$$

If we negate the sign of any spatial sample $x[m, n]$ when $m + n$ is odd, i.e.

$$\left[\begin{array}{cccccc} x[0, 0] & -x[0, 1] & x[0, 2] & \cdots \\ -x[1, 0] & x[1, 1] & -x[1, 2] & \cdots \\ x[2, 0] & -x[2, 1] & x[2, 2] & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{array} \right] \quad (4.267)$$

then the resulting 2-D Fourier spectrum will be centralized. For the example above, the centralized spectrum becomes

$$\mathbf{X}_r = \left[\begin{array}{c|cccc|cccc} 5.0 & -1.5 & 0.0 & -8.5 & 15.0 & -8.5 & 0.0 & -1.5 \\ \hline -14.0 & 6.3 & 2.7 & 16.8 & -41.9 & 33.4 & -7.7 & 4.3 \\ 5.0 & -2.9 & -5.0 & 0.0 & 5.0 & -17.1 & 5.0 & 0.0 \\ -21.0 & 13.2 & 27.7 & -11.3 & -63.1 & 85.7 & -32.7 & 1.6 \\ \hline 55.0 & -21.1 & 10.0 & -98.9 & 165.0 & -98.9 & 10.0 & -21.1 \\ -21.0 & 1.6 & -32.7 & 85.7 & -63.1 & -11.3 & 27.7 & 13.2 \\ 5.0 & 0.0 & 5.0 & 17.1 & 15.0 & 0.0 & -5.0 & -2.9 \\ -14.0 & 4.3 & -7.7 & 33.4 & -41.9 & 16.8 & 2.7 & 6.3 \end{array} \right] \quad (4.268)$$

and

$$\mathbf{X}_j = \left[\begin{array}{cccc|cccc} 0.0 & -1.5 & -5.0 & -8.5 & 0.0 & -8.5 & 5.0 & 1.5 \\ 5.2 & 0.8 & 15.0 & -31.9 & 15.5 & 13.2 & -12.9 & -4.9 \\ -5.0 & 0.0 & -5.0 & 17.1 & -15.0 & 0.0 & 5.0 & 2.9 \\ 30.2 & -6.9 & 27.1 & -89.2 & 90.5 & -19.9 & -15.0 & -16.8 \\ \hline 0.0 & -11.1 & -55.0 & 88.9 & 0.0 & -88.9 & 55.0 & 11.1 \\ -30.2 & 16.8 & 15.0 & 19.9 & -90.5 & 89.2 & -27.1 & 6.9 \\ 5.0 & -2.9 & -5.0 & 0.0 & 15.0 & -17.1 & 5.0 & 0.0 \\ -5.2 & 4.9 & 12.9 & -13.2 & -15.5 & 31.9 & -15.0 & -0.8 \end{array} \right] \quad (4.269)$$

4.4 Homework Problems

1. Prove the following DTFT properties:
 - a. DTFT of time and frequency shift (Eqs.4.32 and 4.31)
 - b. DTFT of correlation (Eq.4.34)
 - c. DTFT of time convolutions (Eq.4.37)
 - d. DTFT of frequency convolutions (Eq.4.38)
 - e. DTFT of accumulation (Eq.4.40) (Hint: Note that $X(f + n) = X(f)$ is periodic with period 1 and $X(n) = X(0)$.)
 - f. DTFT of frequency differentiation (Eq.4.43)
 - g. DTFT of modulation (Eq.4.44) (Hint: note that $(-1)^n = (e^{-j\pi})^n$. Alternatively, we could also let $f_0 = 1/2$ in Eq.4.33 for the frequency shift property.)
2. Find the Fourier transform of each of the following signals:
 - a.

$$x[n] = n(1/2)^n u[n] \quad (4.270)$$

b.

$$x[n] = (1/2)^{|n|} \sin((n-1)\pi/4) \quad (4.271)$$

c.

$$x[n] = (n-1)(1/2)^{|n|} \quad (4.272)$$

d.

$$x[n] = \cos(n\pi/3) \frac{\sin(n\pi/4)}{n\pi} \quad (4.273)$$

3. Given the input $x[n]$ and the corresponding output $y[n]$ of an LTI system shown below

$$x[n] = (1/2)^n u[n], \quad (1/3)^n u[n] \quad (4.274)$$

- a. find its frequency response function $H(f)$ and impulse response function $h[n]$;
- b. Carrying out the convolution $h[n] * x[n] = y[n]$ to verify your result.
4. A signal $x[n] = 2 \cos(n\pi/8) + 2 \cos(n\pi/3)$ is taken as the input to each of the following LTI systems:

a.

$$h_1[n] = \frac{\sin(n\pi/6)}{n\pi}$$

b.

$$h_2[n] = \frac{\sin(n\pi/2)}{n\pi} + \frac{\sin(n\pi/6)}{n\pi}$$

c.

$$h_3[n] = \frac{\sin(n\pi/2)}{n\pi} - \frac{\sin(n\pi/6)}{n\pi}$$

d.

$$h_4[n] = \frac{\sin(n\pi/6)}{n\pi} \frac{\sin(n\pi/2)}{n\pi} \frac{3}{\pi}$$

find the corresponding output $y_i[n]$ ($i = 1, 2, 3, 4$).

5. Two signals $x_1(t)$ and $x_2(t)$ are both band-limited, i.e., $X_1(f) = 0$ for $|f| > f_{max1}$ and $X_2(f) = 0$ for $|f| > f_{max2}$. Find the minimum sampling frequency for sampling each of the following signals with out aliasing or folding.
- a. $x_1(t) + x_2(t - \tau)$
 - b. $x_1(t)x_2(t)$
 - c. $x_1(t) * x_2(t)$
 - d. $x_1(t) \cos(2\pi f_0 t)$
 - e. $dx_1(t)/dt$
 - f. $x_1(at)$
6. The following signal is sampled with sampling frequency $F = 1/t_0 = 8$ Hz ($t_0 = 1/F = 1/8$ is the sampling period):

$$x(t) = \sin(2\pi f_0 t) = \frac{1}{2j}[e^{j2\pi f_0 t} - e^{-j2\pi f_0 t}] \quad (4.275)$$

The resulting discrete samples can be represented as:

$$x[n] = x(t)|_{t=nt_0} = x(nt_0) = x(n/F) = x(n/8) \quad (4.276)$$

For each of the following possible frequencies f_0 of $x(t)$, (a) give the expression $x[n]$ for the sampled signal and indicate whether the signal is sufficiently sampled, aliased, or folded, (b) plot the sampling process in time domain to show how the continuous signal is sampled, and (c) show the spectrum of the sampled signal in frequency domains, thereby explain whether and why aliasing/folding happens.

- $f_0 = 3 < F/2 = 4$ Hz
- $f_0 = 5 > F/2 = 4$ Hz

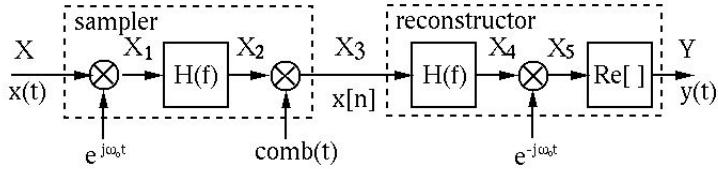


Figure 4.25 A pair of sampler and reconstructor

- $f_0 = 9 > F/2 = 4 \text{ Hz}$
- 7. Assume the signal energy is concentrated within a frequency band $f_{min} < |f| < f_{max}$ where $f_{min} = 2 \text{ kHz}$ and $f_{max} = 3 \text{ kHz}$, as shown in Fig.4.10 in the text. What is the lowest sampling frequency F with which a perfect reconstruction is possible. Which of these possible sampling frequencies 2, 2.5, 3, 3.5, 4, 4.5, 5, 5.5, 6 kHz will allow a perfect reconstruction of the signal from its samples?

Hint: Consider using some graphic tool to visualize the periodic spectrum after sampling with different rates F .

- 8. Fig.4.25 shows the combination of a sampler and the corresponding reconstructor by which the sampling frequency could be significantly reduced, based on the assumption that the signal $x(t)$ is real and its energy is concentrated within a frequency band $f_{min} < |f| < f_{max}$. In the sampler, before $x(t)$ is sampled, it is first multiplied by $e^{j\omega_0 t} = e^{j2\pi f_0 t}$ where $f_0 = (f_{min} + f_{max})/2$ and then filtered by an ideal filter $H(f)$. In the reconstructor, after the same ideal filter $H(f)$, the signal is further multiplied by $e^{-j\omega_0 t}$ and then its real part is taken as the output.

Assume the energy of signal $x(t)$ is again concentrated within a frequency band $f_{min} < |f| < f_{max}$ as shown in Fig.4.10.

- a. Sketch the spectra $X_i(f)$ ($i = 1, 2, 3, 4, 5$) of the signals along the path in both the sampler and reconstructor;
- b. Determine the minimum cut-off frequency f_c of the two ideal low-pass filters;
- c. Determine the lowest sampling frequency F of the comb function $comb(t) = \sum_m \delta(t - m/F)$ for a perfect reconstruction. (Without using this method, the lowest sampling frequency for perfect reconstruction is $F > 2f_{max}$.)
- d. Show that the reconstructed signal $y(t)$ or its spectrum $Y(f)$ is the same as the input $x(t)$ or its spectrum $X(f)$ (up to a scaling factor which is neglected).

Hint:

- If $\mathcal{F}[x(t)] = X(f) = X_r(f) + jX_j(f)$, where $X_r(f) = Re[X(f)]$ and $X_j(f) = Im[X(f)]$, then $\mathcal{F}[Re[x(t)]] = Even[X_r(f)] + jOdd[X_j(f)]$ i.e., taking the real part of $x(t)$ in time domain corresponds to taking the even and odd parts of the real and imaginary parts of $X(f)$ in frequency, respectively.

- $X_{even}(f) = [X(f) + X(-f)]/2$, $X_{odd}(f) = [X(f) - X(-f)]/2$
9. Provided in the web site for the book as well as the CD attached to the book, a Matlab function guidemo_sampling allows the user to specify the parameters (frequency, amplitude and phase) of two sinusoids as well as the sampling rate and displays the combination of the two sinusoids and the discrete samples in both time and frequency domains. Use this function to explore different combinations of the two signals as well as the sampling rate, and inspect the possible aliasing and folding in both time and frequency domain.
 10. Find the discrete signal $x[n]$ obtained by sampling each of the following continuous signals at sampling rate $F = 10$ samples/second. Confirm the three cases of different folding shown in the three panels of Fig.4.9. Use the provided Matlab function guidemo_sampling to reproduce these cases and explore other possible combinations of different signal frequencies and sampling rates.
 - a. $x_1(t) = 2 \cos(2\pi 7t) + \cos(2\pi 2t)$
 - b. $x_2(t) = 2 \cos(2\pi 8t) + \cos(2\pi 2t)$
 - c. $x_3(t) = 2 \cos(2\pi 8t) - 2 \cos(2\pi 2t)$
 11. Let $x[n] = n$ be a discrete signal with period $N = 4$. Find its DFT by matrix multiplication $\mathbf{X} = \overline{\mathbf{W}}\mathbf{x}$, where \mathbf{W} is given in Eq.4.132. Then carry out the inverse DFT by $\mathbf{x} = \mathbf{W}\mathbf{X}$ to confirm the signal is perfectly reconstructed.
 12. Let $\mathbf{x} = [1, 1, -1, -1, 1, 1, -1, -1]^T$ be the input to an LTI system with impulse response $\mathbf{h} = [1, 2, 3]^T$. Find the output $y[n] = h[n] * x[n]$ in two different ways: (1) time domain convolution, and (2) frequency domain multiplication. Write a Matlab program to Confirm your result.
 Note that given any two of three variables in frequency domain, the frequency response function $H[k] = \mathcal{F}[h[n]]$, the input $X[k] = \mathcal{F}[x[n]]$, and the corresponding output $Y[k] = \mathcal{F}[y[n]]$, we can always easily find the third based on the simple relationship $Y[k] = H[k] X[k]$. This is not possible in time domain. Now verify your solution $y[n]$ in two ways: (1) Find $x[n]$ given $h[n]$ and $y[n]$, and (2) Find $h[n]$ given $x[n]$ and $y[n]$.
 13. The impulse response of a discrete LTI system is $h[n] = a^n u[n]$ with $|a| < 1$ and the input is $x[n] = \cos(2\pi n f_0)$. Find the corresponding output $y[n] = h[n] * x[n]$ in both time and frequency domains.

5 Applications of the Fourier Transforms

As a general mathematical tool, the Fourier transform finds a wide variety of applications in both science and engineering. Essentially, any field that deals with signals, either sinusoidal waves or any combination thereof, may benefit from the Fourier transform method for data processing and analysis. In this chapter we only consider a small set of some typical applications.

5.1 LTI Systems in Time and Frequency Domains

Previously, we considered mostly the Fourier transform of a given signal $x(t)$ and the resulting spectrum $X(f) = \mathcal{F}[x(t)]$ that characterizes the frequency contents of the signal. However, the Fourier transform can also be used to characterize a linear, time-invariant (LTI) system. Recall that the output of an LTI system can be found as the convolution of the input and the impulse response function $h(t)$ of the system (Eq. 1.85):

$$y(t) = \mathcal{O}[x(t)] = h(t) * x(t) = \int_{-\infty}^{\infty} h(\tau)x(t - \tau)d\tau \quad (5.1)$$

Due to the convolution theorem (Eq.3.116) the output can be also obtained in frequency domain as:

$$Y(f) = H(f)X(f) \quad (5.2)$$

where $X(f)$ and $Y(f)$ are respectively the spectra of the input and output, and $H(f)$ is the Fourier transform of the impulse response $h(t)$, the *frequency response function (FRF)* of the system (Eq.1.91):

$$H(f) = \mathcal{F}[h(t)] = \int_{-\infty}^{\infty} h(t)e^{-j2\pi ft}dt \quad (5.3)$$

In particular, when the input is a complex exponential $x(t) = e^{j2\pi ft}$, the corresponding output is:

$$\begin{aligned} y(t) &= \mathcal{O}[e^{j2\pi ft}] = h(t) * e^{j2\pi ft} = \int_{-\infty}^{\infty} h(\tau) e^{j2\pi f(t-\tau)} d\tau \\ &= e^{j2\pi ft} \int_{-\infty}^{\infty} h(\tau) e^{-j2\pi f\tau} d\tau = H(f) e^{j2\pi ft} \\ &= |H(f)| e^{j\angle H(f)} e^{j2\pi ft} = |H(f)| e^{j(2\pi ft + \angle H(f))} \end{aligned} \quad (5.4)$$

This is the eigenequation of the LTI system indicating that when its input is a complex exponential, its output is the same exponential scaled by its FRF $H(f) = |H(f)| e^{j\angle H(f)}$.

If the system is real with $h(t) = \bar{h}(t)$, then taking the real part on both sides of the equation above we get:

$$\begin{aligned} \mathcal{O}[Re[e^{j2\pi ft}]] &= \mathcal{O}[\cos 2\pi ft] = Re[|H(f)| e^{j(2\pi ft + \angle H(f))}] \\ &= |H(f)| \cos(2\pi ft + \angle H(f)) \end{aligned} \quad (5.5)$$

Of course we can also take the imaginary part of Eq.5.4 to get:

$$\mathcal{O}[\sin 2\pi ft] = |H(f)| \sin(2\pi ft + \angle H(f)) \quad (5.6)$$

We see that the response of any real LTI system to a sinusoidal input is the same sinusoid with its amplitude scaled by the magnitude of the FRF, and its phase shifted by the phase angle of the FRF.

The result in Eq.5.4 can be generalized to cover any input that can be expressed as a linear combination of a set of sinusoids (inverse Fourier transform in Eq.3.58):

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{j\omega t} df \quad (5.7)$$

The corresponding output of the LTI system is $y(t) = h(t) * x(t)$. However, due to the linearity of the system, we can also get the output as:

$$\begin{aligned} y(t) &= \mathcal{O}[x(t)] = \mathcal{O}\left[\int_{-\infty}^{\infty} X(f) e^{j2\pi ft} df\right] = \int_{-\infty}^{\infty} X(f) \mathcal{O}[e^{j2\pi ft}] df \\ &= \int_{-\infty}^{\infty} X(f) H(f) e^{j2\pi ft} df = \mathcal{F}^{-1}[X(f) H(f)] = \mathcal{F}^{-1}[Y(f)] \end{aligned} \quad (5.8)$$

where $Y(f) = H(f)X(f)$ (Eq.5.2). We see that the output $y(t)$ happens to be the inverse Fourier transform of $Y(f) = H(f)X(f)$. In other words, while in time domain the output is the convolution of the input and the impulse response function $y(t) = h(t) * x(t)$, in frequency domain the output is the product $Y(f) = H(f)X(f)$ of the input and the frequency response function.

All results derived above for continuous signals can be extended to discrete signals. If the discrete input to an LTI system is a complex exponential $x[n] =$

$e^{j2\pi f n} = e^{j\omega n}$, the corresponding output is:

$$\begin{aligned} y[n] &= \mathcal{O}[x[n]] = h[n] * x[n] = \sum_{\nu=-\infty}^{\infty} h[\nu] e^{j2\pi f(n-\nu)} \\ &= e^{j2\pi f n} \sum_{\nu=-\infty}^{\infty} h[\nu] e^{-j2\pi f \nu} = e^{j2\pi f n} H(f) = e^{j\omega n} H(\omega) \end{aligned} \quad (5.9)$$

where

$$H(f) = \mathcal{F}[h[n]] = \sum_{n=-\infty}^{\infty} h[n] e^{-j2\pi f n} \quad (5.10)$$

is the Fourier transform of the impulse response $h[n]$ (Eq.4.13, also the frequency response function of the system, first given in Eq.1.111. Also, similar to the continuous case, we have:

$$\mathcal{O}[\cos(2\pi n f)] = \operatorname{Re}[|H(f)| e^{j(2\pi n f + \angle H(f))}] = |H(f)| \cos(2\pi n f + \angle H(f)) \quad (5.11)$$

As in general a discrete input $x[n]$ to the LTI system can be expressed as (Eq.4.13):

$$x[n] = \int_0^1 X(f) e^{j2n\pi f} df \quad (5.12)$$

the corresponding output can be found to be:

$$\begin{aligned} y[n] &= \mathcal{O}[x[n]] = \mathcal{O}\left[\int_0^1 X(f) e^{j2n\pi f} df\right] = \int_0^1 X(f) \mathcal{O}[e^{j2n\pi f}] \\ &= \int_0^1 X(f) H(f) e^{j2n\pi f} = \mathcal{F}^{-1}[Y(f)] \end{aligned} \quad (5.13)$$

which is the inverse DTFT of the of $Y(f) = X(F)H(f)$.

The results above for both continuous and discrete cases are of course the same as the convolution theorems given in Eqs.3.113 and 4.37, which is illustrated in Fig.5.1. We see that an LTI system can be described by its impulse response function $h(t)$ in time domain, or by its frequency response function $H(f) = \mathcal{F}[h(t)]$ in frequency domain. Correspondingly, the response of the system to a given input $x(t)$ can be obtained as a convolution $y(t) = h(t) * x(t)$ in time domain, or as a product $Y(f) = H(f)X(f)$ in frequency domain. Although both the forward and inverse Fourier transforms are needed for the frequency domain method, we can gain some benefits not possible in time domain. Most obviously, the response of an LTI system to an input $x(t)$ can be conveniently obtained in frequency domain by a multiplication, instead of the corresponding convolution in time domain.

Moreover, as the output of an LTI system can be expressed as a product $Y(f) = H(f)X(f)$, given any two of the three variables $X(f)$, $H(f)$ and $Y(f)$, we can always conveniently find the third, as shown in the following three cases:

1. Prediction of system output:

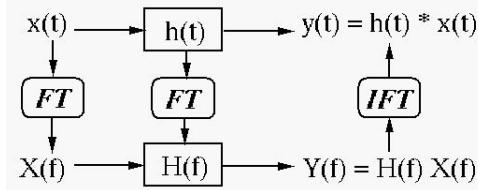


Figure 5.1 Signal through system in time and frequency domains

Given input $X(f)$ to the FRF $H(f)$, we can find the output $Y(f)$. This operation can also be carried out equivalently as a convolution in time domain.

2. System identification/filter design:

Given input $X(f)$ and the observed output $Y(f)$, we can determine the FRF $H(f) = Y(f)/X(f)$ of an unknown system. This process is also useful in design of a system, called a filter in signal processing, given the input and desired output. Correspondingly in time domain, it is difficult to find $h(t)$ given $x(t)$ and $y(t)$.

3. Signal restoration:

Based on the observed output $Y(f)$ from a measuring system with known FRF $H(f)$, we can find the input $X(f)$ without the distortion caused by the system. In time domain, it is difficult to find $x(t)$ given $y(t)$ and $h(t)$.

5.2 Solving Differential and Difference Equations

An important type of LTI systems can be described by a *linear constant-coefficient differential equation (LCCDE)* that relates its output $y(t)$ to its input $x(t)$:

$$\sum_{k=0}^N a_k \frac{d^k}{dt^k} y(t) = \sum_{k=0}^M b_k \frac{d^k}{dt^k} x(t) \quad (5.14)$$

If the input is a complex exponential $x(t) = e^{j\omega t}$, then according to Eq.5.4, the output is also a complex exponential $y(t) = H(\omega)e^{j\omega t}$ with a complex coefficient $H(\omega)$, the frequency response function (FRF) of the system. Note that this output here is the *steady state response* of the system to the complex exponential input. (Initial conditions and transient response of the system will be considered later). Substituting such $x(t)$ and $y(t)$ into the LCCDE above and applying the time differentiation property (Eq.3.117), we get:

$$H(\omega) \sum_{k=0}^N a_k (j\omega)^k e^{j\omega t} = \sum_{k=0}^M b_k (j\omega)^k e^{j\omega t} \quad (5.15)$$

Solving this we get the FRF of the system:

$$H(\omega) = \frac{\sum_{k=0}^M b_k(j\omega)^k}{\sum_{k=0}^N a_k(j\omega)^k} = \frac{N(\omega)}{D(\omega)} \quad (5.16)$$

where $N(\omega) = \sum_{k=0}^M b_k(j\omega)^k$ and $D(\omega) = \sum_{k=0}^N a_k(j\omega)^k$ are the numerator and denominator of $H(\omega)$, respectively.

More generally, consider an input $x(t) = X(\omega)e^{j\omega t}$ with a complex coefficient $X(\omega) = |X(\omega)|e^{j\angle X(\omega)}$ called the *phasor* of $x(t)$. The corresponding output can be assumed to be also a complex exponential $y(t) = Y(\omega)e^{j\omega t}$ with a phasor coefficient $Y(\omega)$. Substituting such $x(t)$ and $y(t)$ into the differential equation, we get:

$$Y(\omega) \sum_{k=0}^N a_k(j\omega)^k e^{j\omega t} = X(\omega) \sum_{k=0}^M b_k \frac{d^k}{dt^k} x(t) \quad (5.17)$$

This result can also be directly obtained by taking the Fourier transform on both sides of the LCCDE in Eq.5.14. We see that the FRF of the LTI system can also be found as the ratio of the output phasor $Y(\omega)$ and input phasor $X(\omega)$:

$$H(\omega) = \frac{Y(\omega)}{X(\omega)} = \frac{\sum_{k=0}^M b_k(j\omega)^k}{\sum_{k=0}^N a_k(j\omega)^k} = \frac{N(\omega)}{D(\omega)} \quad (5.18)$$

This is also the definition of the FRF of a continuous LTI system described by the LCCDE Eq.5.14. In this case of a continuous LTI system described by a LCCDE, the frequency $\omega = 2\pi f$ only appears in the form of $j\omega$ in all functions in frequency domain including $H(\omega)$, $X(\omega)$, $Y(\omega)$, $N(\omega)$, and $D(\omega)$. For this reason, these functions could also be denoted as functions of $j\omega$, such as $H(j\omega)$.

Moreover, due to the linearity of the system, if the input is linear combination of complex exponentials:

$$x(t) = \int_{-\infty}^{\infty} X(\omega) e^{j\omega t} d\omega \quad (5.19)$$

we can get $X(\omega) = \mathcal{F}[x(t)]$. Given the FRF $H(\omega)$ of the system, we can find the output:

$$y(t) = \mathcal{F}^{-1}[Y(\omega)] = \mathcal{F}^{-1}[H(\omega)X(\omega)] = \int_{-\infty}^{\infty} H(\omega)X(\omega) e^{j\omega t} d\omega \quad (5.20)$$

In parallel with the continuous LTI systems described by the LCCDE in Eq.5.14, one particular type of discrete LTI systems can be described by a *linear constant-coefficient difference equation (LCCDE)* that relates the output $y[n]$ to the input $x[n]$:

$$\sum_{k=0}^N a_k y[n-k] = \sum_{k=0}^M b_k x[n-k] \quad (5.21)$$

If the input is a complex exponential $x[n] = e^{j\omega n}$, then according to Eq.5.9 the output is also a complex exponential $y[n] = H(\omega)e^{j\omega n}$. Substituting such $x[n]$

and $y[n]$ into the equation above, we get:

$$H(\omega) \sum_{k=0}^N a_k e^{-j\omega k} = \sum_{k=0}^M b_k e^{-j\omega k} \quad (5.22)$$

Solving for $H(\omega)$ we get:

$$H(\omega) = \frac{\sum_{k=0}^M b_k e^{-j\omega k}}{\sum_{k=0}^N a_k e^{-j\omega k}} = \frac{N(e^{j\omega})}{D(e^{j\omega})} \quad (5.23)$$

where $N(\omega) = \sum_{k=0}^M b_k e^{-j\omega k}$ and $D(\omega) = \sum_{k=0}^N a_k e^{-j\omega k}$. Alternatively taking the DTFT on both sides of Eq.5.21 and applying the time shift property (Eq.4.32), we get:

$$Y(\omega) \sum_{k=0}^N a_k e^{-j\omega k} = X(\omega) \sum_{k=0}^M b_k e^{-j\omega k} \quad (5.24)$$

and we also get the FRF:

$$H(\omega) = \frac{Y(\omega)}{X(\omega)} = \frac{\sum_{k=0}^M b_k e^{-j\omega k}}{\sum_{k=0}^N a_k e^{-j\omega k}} = \frac{N(\omega)}{D(\omega)} \quad (5.25)$$

This is also the definition of the FRF of a discrete LTI system described by the LCCDE Eq.5.21. In this case of a discrete LTI system described by a LCCDE, the frequency $\omega = 2\pi f$ only appears in the form of $e^{j\omega}$ in all functions in frequency domain including $H(\omega)$, $X(\omega)$, $Y(\omega)$, $N(\omega)$, and $D(\omega)$. For this reason, these functions could also be denoted as functions of $e^{j\omega}$, such as $H(e^{j\omega})$.

Given the input $X(\omega) = \mathcal{F}[x[n]]$ and the FRF $H(\omega)$ of the system, we can find the output in time domain:

$$y[n] = \mathcal{F}^{-1}[Y(\omega)] = \mathcal{F}^{-1}[H(\omega)X(\omega)] \quad (5.26)$$

In summary, we can solve an LCCDE system, either continuous or discrete, by following these steps:

- Find the FRF of the system $H(\omega) = N(\omega)/D(\omega)$;
- Carry out the CTFT of a continuous input $x(t)$ to find $X(\omega) = \mathcal{F}[x(t)]$, or the DTFT of a discrete input $x[n]$ to find $X(\omega) = \mathcal{F}[x[n]]$;
- Obtain the response in frequency domain $Y(\omega) = H(\omega)X(\omega)$;
- Carry out the inverse DTFT or CTFT on $Y(\omega)$ to get $y(t)$ or $y[n]$.

Example 5.1: In a circuit composed of a resistor R and a capacitor C as shown in Fig.5.2, the input $x(t) = v_{in}(t)$ is the voltage across both R and C in series, and the output $y(t) = v_C(t)$ is the voltage across C . Find both the step and impulse response of the system, and the FRF $H(\omega)$ of the system.

- Set up differential equation:

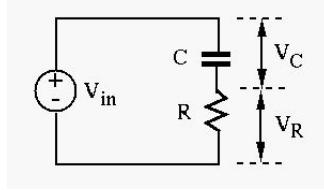


Figure 5.2 An RC circuit

The current through both \$C\$ and \$R\$ is \$i(t) = C dv_C(t)/dt = C\dot{y}(t)\$, and by Ohm's law, the voltage across \$R\$ is \$v_R(t) = Ri(t) = RC\dot{y}(t)\$. The input voltage \$x(t)\$ is the sum of \$v_R(t)\$ and \$v_C(t)\$:

$$v_R(t) + v_C(t) = RC\dot{y}(t) + y(t) = \tau\dot{y}(t) + y(t) = x(t) = v_{in}(t) \quad (5.27)$$

where \$\tau = RC\$ is the time constant of the system. Dividing both sides by \$\tau\$ we get:

$$\dot{y}(t) + \frac{1}{\tau}y(t) = \frac{1}{\tau}x(t) \quad (5.28)$$

- Find step response:

- Find homogeneous solution \$y_h(t)\$ when \$x(t) = 0\$:

Assume \$y_h(t) = Ae^{st}\$ and we have \$\dot{y}_h(t) = sAe^{st}\$, now the homogeneous differential equation becomes:

$$(s\tau + 1)Ae^{st} = 0, \quad \text{i.e.,} \quad s\tau + 1 = 0 \quad (5.29)$$

we therefore get \$s = -1/\tau\$ and \$y_h(t) = Ae^{-t/\tau}\$.

- Find the particular solution \$y_p(t)\$ when \$x(t) = u(t)\$:

As the right-hand side is a constant \$1/\tau\$ for \$t > 0\$, we assume the corresponding output is also a constant \$y_p(t) = C\$ and \$\dot{y}_p(t) = 0\$. Substituting these into the equation we get \$y_p(t) = 1\$.

- Find the complete response to unit step:

$$y(t) = y_h(t) + y_p(t) = [Ae^{-t/\tau} + 1]u(t) \quad (5.30)$$

Given the initial condition \$y(t)|_{t<0} = y_0\$ (initial voltage across \$C\$), we get \$A = y_0 - 1\$, and the complete response to \$x(t) = u(t)\$ is:

$$y(t) = [(y_0 - 1)e^{-t/\tau} + 1]u(t) = [(1 - e^{-t/\tau}) + y_0e^{-t/\tau}]u(t) \quad (5.31)$$

Physically the first term is for the charging of the capacitor due to the step input while the second term is for the discharge of the capacitor with a non-zero initial voltage. In particular, when \$y_0 = 0\$, we have:

$$y(t) = (1 - e^{-t/\tau})u(t) \quad (5.32)$$

- Find impulse response \$h(t)\$:

Due to the fact that if \$\mathcal{O}[x(t)] = y(t)\$, then \$\mathcal{O}[x(t)] = \dot{y}(t)\$ (Eq.1.75) (valid for a DE under zero initial condition), we can get the impulse response \$h(t)\$ to

$\delta(t) = du(t)/dt$ by taking the derivative of the unit response to $u(t)$ obtained above:

$$\begin{aligned} h(t) &= \dot{y}(t) = \frac{d}{dt}[(1 - e^{-t/\tau})u(t)] \\ &= \frac{1}{\tau}e^{-t/\tau}u(t) + (1 - e^{-t/\tau})\delta(t) = \frac{1}{\tau}e^{-t/\tau}u(t) \end{aligned}$$

Given the impulse response $h(t)$, we can also find the step response by convolution (see Example 1.4):

$$h(t) * u(t) = \frac{1}{\tau} \int_0^t e^{-(t-t')/\tau} dt' = \frac{1}{\tau}e^{-t/\tau}\tau(e^{t/\tau} - 1)u(t) = (1 - e^{-t/\tau})u(t) \quad (5.33)$$

where $u(t)$ is included to reflect the fact that this result is valid only if $t > 0$. This result is the same as the one above.

- A different method to find $h(t)$:

As the system is causal, $h(t) = 0$ for all $t < 0$ when the input is zero, we can assume

$$h(t) = f(t)u(t) = \begin{cases} f(t) & t > 0 \\ 0 & t < 0 \end{cases} \quad (5.34)$$

where $f(t)$ is a function to be determined, and have:

$$\dot{h}(t) = \dot{f}(t)u(t) + f(t)\dot{u}(t) = \dot{f}(t)u(t) + f(0)\delta(t) \quad (5.35)$$

Now Eq.5.28 becomes:

$$\tau\dot{f}(t)u(t) + \tau f(0)\delta(t) + f(t)u(t) = \delta(t) \quad (5.36)$$

Separating terms containing $u(t)$ and $\delta(t)$ respectively, we get two equations:

$$\begin{cases} \tau\dot{f}(t) + f(t) = 0 \\ f(0) = 1/\tau \end{cases} \quad (5.37)$$

This homogeneous equation with an initial condition can be solved to get

$$f(t) = \frac{1}{\tau}e^{-t/\tau} \quad (5.38)$$

and the impulse response same as above:

$$h(t) = f(t)u(t) = \frac{1}{\tau}e^{-t/\tau}u(t) \quad (5.39)$$

- Find impulse responses by the CTFT:

Taking the CTFT on both sides of Eq.5.28, we get

$$Y(\omega) \left(j\omega + \frac{1}{\tau} \right) = X(\omega) \frac{1}{\tau} \quad (5.40)$$

and the FRF of the system is:

$$H(\omega) = \frac{Y(\omega)}{X(\omega)} = \frac{1/\tau}{j\omega + 1/\tau} = \frac{1}{j\omega\tau + 1} \quad (5.41)$$

Taking the inverse CTFT of $H(\omega)$ we get the impulse response:

$$h(t) = \mathcal{F}^{-1}[H(\omega)] = \mathcal{F}^{-1}\left[\frac{1/\tau}{j\omega + 1/\tau}\right] = \frac{1}{\tau}e^{-t/\tau}u(t) \quad (5.42)$$

- Find step response:

In frequency domain, the response to a unit step $U(\omega) = \mathcal{F}[u(t)]$ is:

$$\begin{aligned} Y(\omega) &= H(\omega)U(\omega) = \frac{1}{j\omega\tau + 1} \left[\frac{1}{2}\delta(f) + \frac{1}{j\omega} \right] \\ &= \frac{1}{2} \frac{\delta(f)}{j\omega\tau + 1} + \frac{1}{j\omega\tau + 1} \frac{1}{j\omega} = \frac{1}{2}\delta(f) + \frac{1}{j\omega} - \frac{\tau}{j\omega\tau + 1} \end{aligned} \quad (5.43)$$

Note that $\delta(f)x(f) = \delta(f)x(0)$. Taking the inverse CTFT of the above we get the step response in time domain:

$$y(t) = \mathcal{F}^{-1}[Y(\omega)] = \mathcal{F}^{-1}\left[\frac{1}{2}\delta(f) + \frac{1}{j\omega}\right] - \mathcal{F}^{-1}\left[\frac{\tau}{j\omega\tau + 1}\right] = (1 - e^{-t/\tau})u(t) \quad (5.44)$$

Example 5.2: Consider an LTI system described by a first order difference equation:

$$y[n] - a y[n - 1] = x[n], \quad \text{or} \quad y[n] = x[n] + a y[n - 1] \quad (5.45)$$

This system is a *recursive filter* as the current output $y[n]$ depends on the past output $y[n - 1]$ as well as the current input $x[n]$. We assume the system is causal, i.e., $h[n] = 0$ for $n < 0$.

- Find impulse response by solving the difference equation:

If the input is $x[n] = \delta[n]$, then the output is $y[n] = h[n]$ and Eq.5.45 becomes $h[n] - a h[n - 1] = \delta[n]$. $h[n]$ can be found recursively:

$$\begin{cases} n = 0 & h[0] - ah[-1] = h[0] = \delta[0] = 1, \quad \text{i.e. } h[0] = 1 \\ n = 1 & h[1] - ah[0] = h[1] - a = \delta[1] = 0, \quad \text{i.e. } h[1] = a \\ n = 2 & h[2] - ah[1] = h[2] - a^2 = \delta[2] = 0, \quad \text{i.e. } h[2] = a^2 \\ \dots\dots\dots & \end{cases} \quad (5.46)$$

Summarizing the above we get $h[n] = a^n u[n]$.

Alternatively, we can also assume a general solution $h[n] = Ae^{jn\omega}$ and Eq.5.45 becomes:

$$Ae^{jn\omega} - a Ae^{j(n-1)\omega} = \delta[n] = 0, \quad (n > 0) \quad (5.47)$$

from which we get $e^{j\omega} = a$ and $h[n] = Aa^n$. Using the initial condition $h[0] = 1$ obtained above, we get $A = 1$ and therefore we also get $h[n] = a^n$. Note that the system is stable only if $|a| < 1$.

- Find step response by convolution:

The response to a unit step $x[n] = u[n]$ can be found by convolution:

$$y[n] = h[n] * u[n] = \sum_{m=-\infty}^{\infty} a^m u[m] u[n-m] = \sum_{m=0}^n a^m = \frac{1-a^{n+1}}{1-a} u[n] \quad (5.48)$$

- Find impulse response by the DTFT:

Taking the DTFT on both sides on Eq.5.45 we get:

$$Y(\omega)(1 - a e^{-j\omega}) = X(\omega) \quad (5.49)$$

and the FRF of the system is:

$$H(\omega) = \frac{Y(\omega)}{X(\omega)} = \frac{1}{1 - a e^{-j\omega}} \quad (5.50)$$

Taking inverse DTFT of $H(f)$, we get the impulse response:

$$h[n] = \mathcal{F}^{-1}[H(\omega)] = a^n u[n] \quad (5.51)$$

- Find step response:

In frequency domain, the response to a unit step $U(\omega) = \mathcal{F}[u[n]]$ is:

$$\begin{aligned} Y(\omega) &= H(\omega)U(\omega) = \frac{1}{1 - ae^{j\omega}} \left[\frac{1}{1 - e^{-j\omega}} + \frac{1}{2} \sum_{n=-\infty}^{\infty} \delta(f-n) \right] \\ &= \frac{1}{1 - ae^{j\omega}} \frac{1}{1 - e^{-j\omega}} + \frac{1}{2} \sum_{n=-\infty}^{\infty} \frac{\delta(f-n)}{1 - ae^{j\omega}} \\ &= \frac{1}{1 - a} \left[\frac{1}{1 - e^{-j\omega}} - \frac{a}{1 - ae^{-j\omega}} \right] + \frac{1}{2} \sum_{n=-\infty}^{\infty} \frac{\delta(f-n)}{1 - a} \\ &= \frac{1}{1 - a} \left[\frac{1}{1 - e^{-j\omega}} - \frac{a}{1 - ae^{-j\omega}} + \frac{1}{2} \sum_{n=-\infty}^{\infty} \delta(f-n) \right] \end{aligned} \quad (5.52)$$

Taking the inverse DTFT of the above we get the step response in time domain (Eqs.4.1.3 and 4.67):

$$y[n] = \mathcal{F}^{-1}[Y(\omega)] = \mathcal{F}^{-1} = \frac{1 - a^{n+1}}{1 - a} u[n] \quad (5.53)$$

5.3 Magnitude and Phase Filtering

In the context of signal processing, an LTI system can be treated as a filter, and the process of a signal $x(t)$ going through the system $h(t)$ becomes a filtering process, which is either as a convolution in time domain or, equivalently, a multiplication in frequency domain (Fig.5.1):

$$y(t) = h(t) * x(t), \quad \text{or} \quad Y(f) = H(f)X(f) \quad (5.54)$$

Note that sometimes a filter may not be causal as its impulse response function $h(t)$ may not be zero for $t < 0$. Obviously a non-causal filter is not actually implementable in real time unless certain delay is allowed, i.e., the filtering is not truly real-time. However, non-causal filter can be readily implemented off-line, as it can be applied to pre-recorded data containing all signal samples at the same time.

The filtering process in frequency domain has the benefit that the signal can be easily manipulated by various filters based on the frequency contents of the signal, due to the frequency locality gained by the Fourier transform (with the cost of sacrificing temporal locality), we could modify and manipulate the phase as well as the magnitude of the frequency components of the signal. The complex multiplication $Y(f) = H(f) X(f)$ in frequency domain can be written in terms of both magnitude and phase:

$$|Y(f)|e^{j\angle Y} = |H(f)||X(f)|e^{j(\angle H + \angle X)}, \quad \text{i.e.} \quad \begin{cases} |Y(f)| = |H(f)||X(f)| \\ \angle Y(f) = \angle H(f) + \angle X(f) \end{cases} \quad (5.55)$$

We now consider both aspects of the filtering process.

- **Magnitude filtering:**

Various filtering schemes can be implemented based on the magnitude of the frequency response function $|H(f)|$ of the filter. Typically, depending on which part of the signal spectrum is enhanced or attenuated, a filter can be classified as one of these different types: low-pass (LP), high-pass (HP), band-pass (BP), and band-stop (BS) filters, as illustrated in Fig.5.3. Moreover, in the case when $|H(f)| = c$ is a constant independent of frequency f (although $\angle H(f)$ may vary as a function of frequency), then $H(f)$ is said to be an all-pass filter. Two parameters are commonly used to characterize a filter:

- The *cutoff frequency* f_c of a filter is the frequency at which $|H(f)|$ is reduced to $1/\sqrt{2} = 0.707$ of the maximum magnitude at the peak frequency:

$$|H(f_c)| = \frac{1}{\sqrt{2}}|H_{max}|, \quad \text{i.e.} \quad |H(f_c)|^2 = \frac{1}{2}|H_{max}|^2 \quad (5.56)$$

where $|H_{max}|$ is the maximum magnitude of $H(f)$ at some peak frequency.

As the power of the filtered signal (proportional to its magnitude squared) at the cutoff frequency f_c is half of the maximum power, the cutoff frequency is also called the *half-power frequency*.

- The *Bandwidth* Δf of a bandpass filter is the interval between two cutoff frequencies on either side of the peak frequency:

$$\Delta f = f_{c1} - f_{c2} \quad (5.57)$$

For a lowpass filter, the lower cutoff frequency $f_{c2} = 0$ is zero and the bandwidth is the same as the cutoff frequency $\Delta f = f_c$.

- **Phase filtering:**

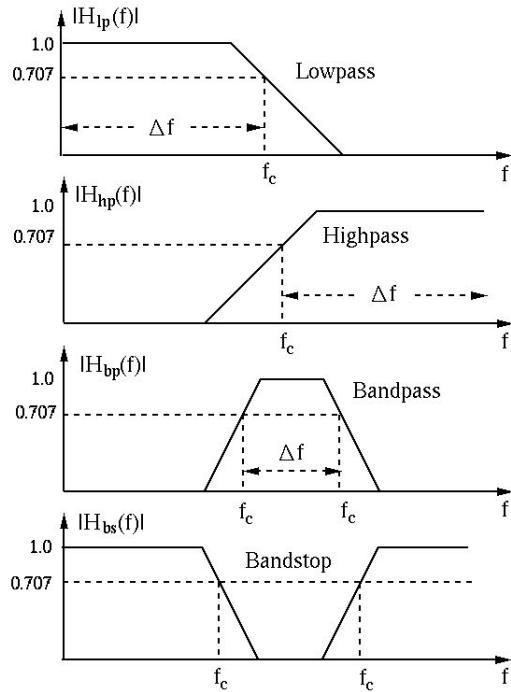


Figure 5.3 Illustration of four different types of filters (low-pass, high-pass, band-pass and ban-stop)

The filtering process affects the phase angle of the signal as well as its magnitude, due to the phase shift $\angle H(f)$ of the filter, which is non-zero in general.

- *Linear phase filtering:* First consider an all-pass filter $H(f)$ with a unity gain and linear phase shift proportional to frequency f :

$$|H(f)| = 1, \quad \angle H(f) = -2\pi\tau f \quad (5.58)$$

For example, if the input $x(t) = \cos(2\pi f_1 t) + \cos(2\pi f_2 t) = \cos(2\pi 2t) + \cos(2\pi 4t)$ contains only two sinusoidal components of frequencies $f_1 = 2$ and $f_2 = 4$ Hz, respectively, as shown in the top panel of Fig.5.4, then the phase shifts of the sinusoids of 2 Hz and 4 Hz are $4\pi\tau$ and $8\pi\tau$, respectively, and their relative positions in time remain the same before and after filtering. Consequently the waveform of the signal also remains the same, except it is delayed in time by a constant amount τ , as shown in the middle panel of Fig.5.4. This result can be generalized to any signal

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi ft} dt \quad (5.59)$$

The output of the all-pass filter corresponding to one particular frequency component $X(f)e^{j2\pi ft}$ of $x(t)$ is:

$$H(f)X(f)e^{j2\pi ft} = |H(f)|e^{j\angle H(f)}e^{j2\pi ft} = X(f)e^{j2\pi f(t-\tau)} \quad (5.60)$$

and the output corresponding to the entire input is the linear combination of all these individual outputs:

$$y(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi f(t-\tau)} df = x(t - \tau) \quad (5.61)$$

which is just a delayed version of the input $x(t)$. Of course this is actually the time shift property of the Fourier transform. The amount of time delay caused by a linear phase filter $H(f)$ can be obtained from its linear phase $\angle H(f)$ as:

$$\tau_\phi = -\frac{\angle H(f)}{2\pi f} = -\frac{\angle H(\omega)}{\omega} \quad (5.62)$$

This is called the *phase delay* of the linear phase filter, whose gain $|H(f)|$ can be either a constant or a function of frequency.

- *Non-linear phase*: If the phase of the filter is not a linear function of frequency, then relative temporal positions of the frequency components of the input signal will not be maintained by the filtering process, consequently the waveform of the output will not be the same as that of the input, i.e., the signal will be distorted by the filter, even if it has a constant gain. Again, in the example above, if the phase shift of the filter is $6\pi\tau$ for both frequencies f_1 and f_2 (and $-6\pi\tau$ for $-f_1$ and $-f_2$) instead of being proportional to them, the waveform of the filtered signal looks very different from the original, as shown in the bottom panel of Fig.5.4. Another example is shown in Fig.5.5, where a square impulse signal is filtered, first by a linear phase filter (top), which causes a pure time delay without any distortion, and then by a constant-phase (non-linear) filter (bottom) by which the signal is distorted.

Although the time delay caused by a non-linear phase filter varies as a function of frequency, we can still find the time delay for any specific frequency f by:

$$\tau_g(f) = -\frac{d\angle H(f)}{2\pi df} = -\frac{d\angle H(\omega)}{d\omega} \quad (5.63)$$

This is a function of frequency (instead of a constant in the previous case) and is defined as the *group delay* of the non-linear filter, approximately representing the time delay of a group of frequency components within a narrow band around this frequency f .

The frequency response function $H(f)$ can be conveniently represented by the *Bode plot*, where both the magnitude $|H(f)|$ and phase angle $\angle H(f)$ are plotted in base-10 logarithmic scale of the frequency f so that the range of frequencies can be increased to several decades. Moreover, the magnitude of $H(f)$ is also be plotted in logarithmic scale, called *log-magnitude* defined as:

$$LmH(f) = 20 \log_{10} |H(f)| \quad (5.64)$$

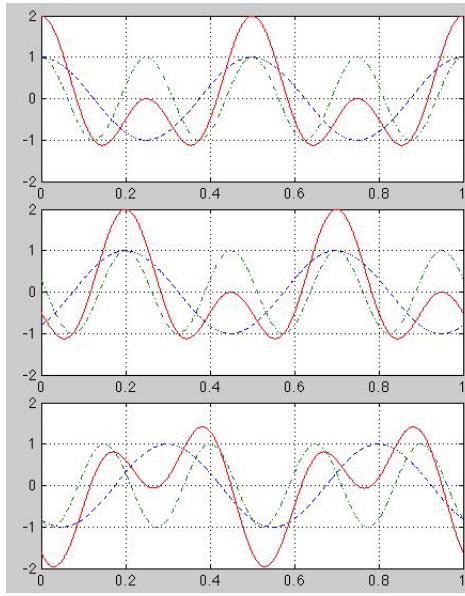


Figure 5.4 Filtering with linear and non-linear phase shift

The original signal containing two sinusoidal components (top panel) of frequencies $f_1 = 2$ and $f_2 = 4$, respectively, is filtered by a filter with linear phase (middle panel) and nonlinear phase (bottom panel). The signals before and after are plotted in solid lines while the two frequency components are plotted in dashed lines.

The unit of the log-magnitude is *decibel* denoted by *dB*. Based on the log-magnitude representation, at the corner frequency $f = f_c$ we have:

$$20 \log_{10} \frac{|H(f_c)|}{|H_{max}|} = 20 \log_{10} \frac{1}{\sqrt{2}} = -3.01 \text{ dB} \approx -3 \text{ dB} \quad (5.65)$$

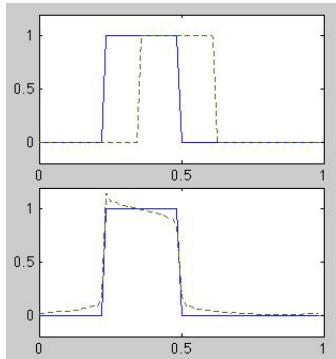


Figure 5.5 Filtering with linear and constant phase shift

The square wave (solid line) is filtered first by a linear-phase filter without distortion (dashed line in top panel) and then by a constant (non-linear) filter with distortion (dashed line in bottom panel).

i.e., the log-magnitude of $H(f_c)$ at the cutoff or half-power frequency is 3 dB lower than the maximum log-magnitude.

One major convenience of using the log-magnitude is that the log-magnitude plot of a frequency response function composed of multiple factors can be obtained as the algebraic sum of the individual plots of these factors. For example, if $H(f) = N_1(f)N_2(f)/[D_1(f)D_2(f)]$, then we can get:

$$\text{Lm}H(f) = \text{Lm} \left[\frac{N_1(f)N_2(f)}{D_1(f)D_2(f)} \right] = \text{Lm}N_1(f) + \text{Lm}N_2(f) - \text{Lm}D_1(f) - \text{Lm}D_2(f) \quad (5.66)$$

same as operation for the phase plot:

$$\angle H(f) = \angle \left[\frac{N_1(f)N_2(f)}{D_1(f)D_2(f)} \right] = \angle N_1(f) + \angle N_2(f) - \angle D_1(f) - \angle D_2(f) \quad (5.67)$$

Also the Bode plot of a cascade of two filters can be found as the algebraic sum of the their individual Bode plots.

5.4 Implementation of 1-D Filtering

Here we consider how the filtering process is carried out computationally, using four different types of low-pass (LP) filters as examples. We will discuss their implementation and filtering effects when applied to a square impulse train shown in the top row of Fig.5.6.

- The *moving average LP-filtering* is carried out in time domain by replacing each sample in the discrete signal by the average of a sequence of neighboring samples. This operation of moving average is actually a convolution of the signal with a square window covering the neighborhood of the sample in question:

$$h(t) * x(t), \quad \text{where } h(t) = \begin{cases} 1 & -w/2 < t < w/2 \\ 0 & \text{otherwise} \end{cases} \quad (5.68)$$

where w is the width of the square window. Correspondingly in frequency domain, the moving average filtering is a multiplication of the signal spectrum $X(f)$ by the FRF $H(f) = \mathcal{F}[h(t)]$, a sinc function first shown in Eq. 3.153. As shown in the 2nd and 3rd rows of Fig.5.6, we see that the sinc FRF $H(f)$ has a lot of leakage, i.e., many high frequency components can still leak through the filter to appear in the output.

- The *ideal LP-filter (rectangular)* is defined in frequency domain as:

$$H(f) = \begin{cases} 1 & |f| < f_c \\ 0 & |f| > f_c \end{cases} \quad (5.69)$$

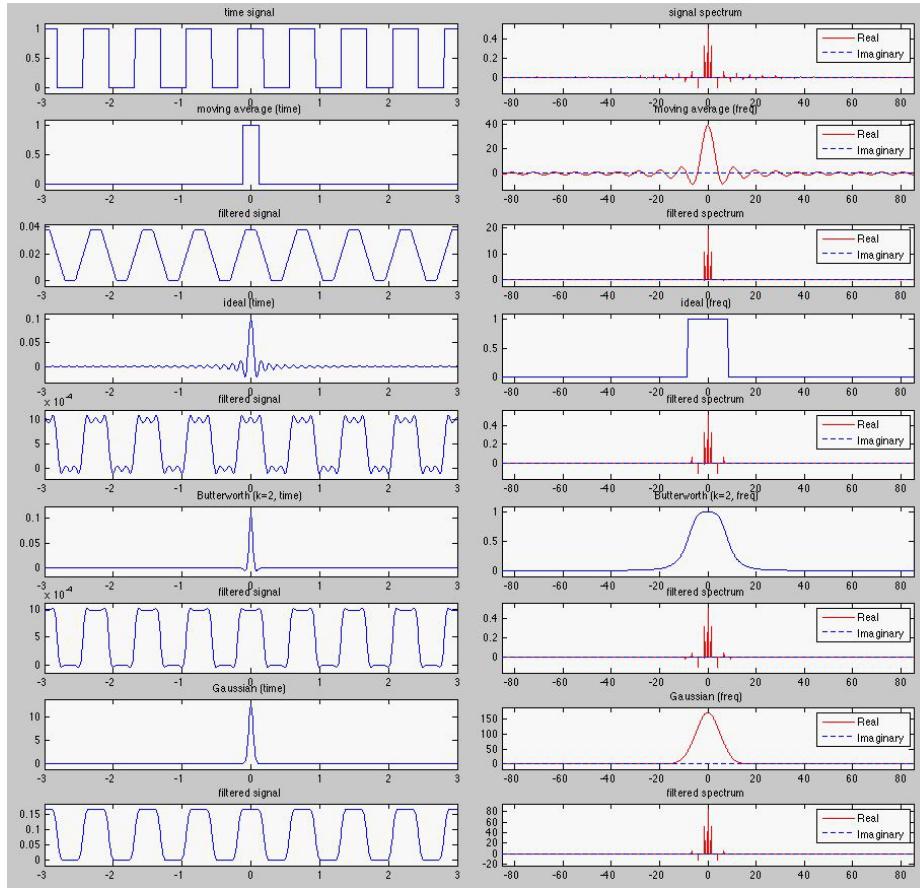


Figure 5.6 1-D low-pass filters in both time (left) and frequency (right) domains
A square impulse train $x(t)$ and its spectrum $X(f)$ are shown in the top row. The following eight rows show $h(t)$ and $H(f)$ of four filters (moving average, ideal, Butterworth and Gaussian), and the corresponding output $y(t) = h(t) * x(t)$ and its spectrum $Y(f) = H(f)X(f)$.

where f_c is the cut-off frequency. As shown in Eq.3.155, in time domain, the impulse response of the ideal filter is a sinc function:

$$h(t) = \frac{\sin(2\pi f_c t)}{\pi t} = 2f_c \operatorname{sinc}(2f_c t) \quad (5.70)$$

After filtering all frequency components outside the passing band are totally removed while those within remain unchanged. As shown in the 4th and 5th rows of Fig.5.6, the ideal LP-filter causes some severe ringing artifacts in the filtered signal, due obviously to the convolution of the signal with the ringing sinc function $h(t) = \mathcal{F}^{-1}[H(f)]$. So the ideal filter in frequency domain does not look ideal in time domain.

- The *Butterworth LP-filter* defined below avoids the ringing artifacts of the ideal filter:

$$H(f) = \frac{1}{\sqrt{1 + (f/f_c)^{2n}}} = \begin{cases} 1 & f = 0 \\ 1/\sqrt{2} & f = f_c \\ 0 & f = \infty \end{cases} \quad (5.71)$$

where f_c is the cut-off frequency at which $H(f) = H(f_c) = 1/\sqrt{2}$, and n is a positive integer for the order of the filter. By adjusting n one can control the shape of the filter and thereby making a proper tradeoff between the ringing effects and how accurately the passing band is specified. As shown in Fig.5.7, when n is small, the shape of the filter is smooth (low frequency accuracy) with little ringing; when n is large, the filter becomes sharper (higher frequency accuracy) but with stronger ringing effect. When $n \rightarrow \infty$, the Butterworth filter becomes an ideal filter. The Butterworth filter with $n = 4$ and its effect are shown respectively in the 6th and 7th rows of Fig.5.6.

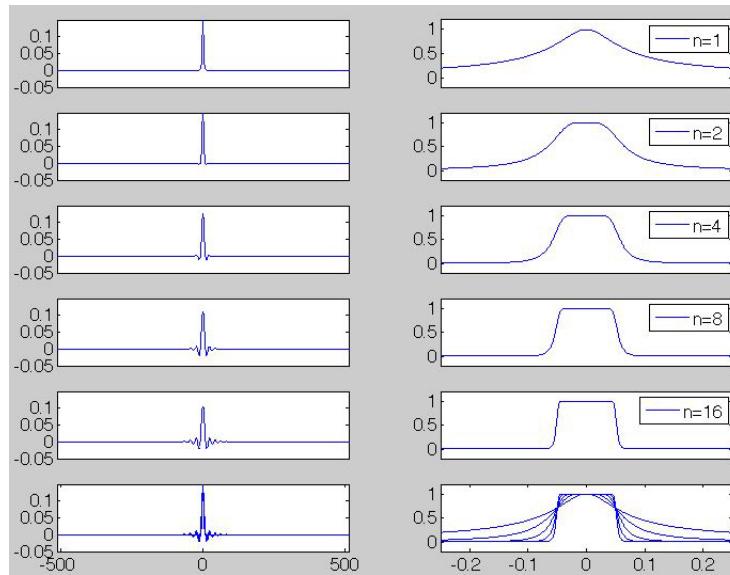


Figure 5.7 Butterworth filters of different orders in both time (left) and frequency (right) domains

The plot in the last row compares all five filters of different orders.

- The *Gaussian filter* can be defined in either frequency or time domain as (3.160):

$$H(f) = e^{-a(f/f_c)^2}, \quad \text{or} \quad h(t) = f_c \sqrt{\pi/a} e^{-(\pi f_c t)^2/a} \quad (5.72)$$

where $a = \ln 2/2 = 0.347$ so that at the cut-off frequency $f = f_c$ we have $H(f_c) = H(0)/\sqrt{2} = 1/\sqrt{2}$. Obviously the Gaussian filter is smooth in both time and frequency domains without any ringing effect. The Gaussian filter and its effect are shown respectively in the 8th and 9th rows of Fig.5.6.

Inspecting the filtered signal in Fig.5.6, we see that the sharp corners of the ideal filter corresponding to some high frequency components are smoothed out by all other type of low-pass filters, consequently the undesirable ringing artifact is much reduced. However, the tradeoff is the sacrifice of the accuracy in defining the passing band in frequency domain, as a smooth low-pass filtering window means necessarily certain high frequency leakage. There also exist some other smooth filters based on the cosine function such as Hann, Hamming and cosine windows.

As all of these filters $H(f)$ are real with zero phase $\angle H(f) = 0$, i.e., they are special linear phase filters with zero delay $\tau_\phi = -\angle H(f)/2\pi f = 0$, only the magnitude of the signal spectrum is modified by the filtering process, while the phase remains the same:

$$\begin{aligned}|Y(f)| &= |H(f)| |X(f)| \neq |X(f)| \\ \angle Y(f) &= \angle H(f) + \angle X(f) = \angle X(f)\end{aligned}\quad (5.73)$$

Consequently the relative positions of the frequency components remain the same, and the waveform of the signal is modified only due to the magnitude of the filter FRF $H(f)$.

Other types of high-pass, band-pass and band-stop filters can be easily derived from the low-pass filters considered above. Specifically, let $H_{lp}(f)$ be a low-pass filter with $H_{lp}(0) = 1$, then a high-pass filter can be easily obtained as:

$$H_{hp}(f) = 1 - H_{lp}(f) \quad (5.74)$$

Also a band-pass filter can be obtained as the difference between two low-pass filters $H_{lp1}(f)$ and $H_{lp2}(f)$ with their corresponding cut-off frequencies satisfying $f_1 > f_2$:

$$H_{bp}(f) = H_{lp1}(f) - H_{lp2}(f) \quad (5.75)$$

and a band-stop filter is obtained simply as

$$H_{bs}(f) = 1 - H_{bp}(f) \quad (5.76)$$

Examples of such filters based on a 4th order Butterworth LP-filters are shown in Fig.5.8.

The discussion above is based on the assumption that the DC component corresponding to origin at zero frequency $f = 0$ is in the middle of the spectrum, while the higher frequency components are farther away on each side. However, in computational implementation of these filters, the signal and the filter are discrete and finite in both time and frequency domain, and the DC component $X[0]$ is the first element, the left most element of the N-D array $[X[0], \dots, X[N-1]]$ for the discrete spectrum, while the high frequency components are around the middle point $n = N/2$. In other words, in order to use the filters given above as LP filters, all spectra need to be centralized as discussed in Chapter 4 (Eq.4.172). Alternatively, without centralizing, a low-pass filter can still be used as a high-pass filter and vice versa.

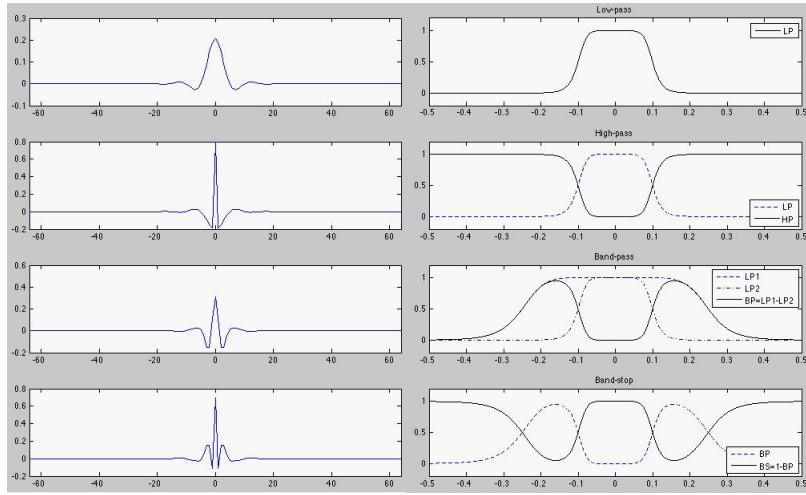


Figure 5.8 Frequency response functions $H(f)$ of LP, HP, BP and BS filters in frequency domain (right) and their corresponding impulse response functions $h(t)$ in time domain (left)

Note that all of these filters are even functions $H(f) = H(-f)$, i.e., both the positive and negative frequencies in $X(f)$ are modified identically by the filter. This is an important requirement of any filter in order to maintain the symmetry property of any real signal being filtered ($X_r(f) = X_r(-f)$, $X_j(f) = -X_j(f)$), so that the output signal obtained by inverse Fourier transform of the filtered spectrum remains real. Any non-even filter will necessarily change the symmetry of the signal spectrum and thereby cause the output to be complex, which makes little sense in general.

Also note that as all filters discussed above are non-causal as their impulse response $h(t)$ is non-zero for $t < 0$, they cannot be implemented in real time. This non-causality can be avoided if the impulse response $h(t)$ has a finite duration, or if it can be truncated without major distortion, so that when it is right shifted in time by certain amount τ , it becomes causal $h(t - \tau) = 0$ for $t < 0$. Correspondingly, the filtered signal is delayed by τ . For example, if we delay the moving average filter by $\tau = w/2$, i.e.,

$$h(t - w/2) = \begin{cases} 1 & 0 < t < w \\ 0 & \text{otherwise} \end{cases} \quad (5.77)$$

This delayed version of the average filter is causal and realizable in real time. Of course these non-causal filtering can all be implemented off-line when all data samples are available and can therefore be arbitrarily manipulated.

Example 5.3: The annual precipitation in Los Angeles area in the $N = 126$ years from 1878 to 2003, treated as a discrete time signal $x[n]$, and the DFT spec-

trum $X[k]$ are shown in the top row of Fig.5.9. Here the average of the data is removed, i.e., the DC component in the middle of the spectrum is zero, so that other frequency components with much smaller magnitudes can be better seen. Four Butterworth filters, including a LP-filter and three BP-filters with different passing bands, are shown in the 2nd, 4th, 6th and 8th rows, while the signals filtered by the corresponding filter are shown respectively in the following 3rd, 5th, 7th and 9th rows.

A *filter bank* can be formed by these four filters. Due to the specific arrangement of the passing bands and the bandwidths of these filters, the filter bank is an *all-pass (AP)* filter, in the sense that its component filters $H_k(f)$ ($k = 1, \dots, 4$) add up approximately to a constant through out all frequencies, i.e., the combined outputs of the filter bank contain approximately all information in the signal. This result is further confirmed by the last (10th) row in Fig.5.9 where the filtered signals in both time and frequency domain are added up and compared to the original signal. As expected, the difference between the sum of the filtered signal and the original one is negligible, i.e., the filtered signals, when combined, contain all information in the signal.

Example 5.4: In *amplitude modulation (AM)* radio broadcasting, a *carrier wave* $c(t) = \cos(2\pi f_c t)$ with *radio frequency (RF)* f_c is modulated by the audio signal $s(t)$ before it is transmitted. The modulation is implemented as a multiplication carried out by a modulator (mixer):

$$x(t) = s(t) c(t) = s(t) \cos(2\pi f_c t) = s(t) \frac{1}{2} [e^{j2\pi f_c t} + e^{-j2\pi f_c t}] \quad (5.78)$$

This multiplication in time domain corresponds to a convolution in frequency domain:

$$\begin{aligned} X(f) &= S(f) * C(f) = S(f) * \frac{1}{2} [\delta(f - f_c) + \delta(f + f_c)] \\ &= \frac{1}{2} [S(f - f_c) + S(f + f_c)] \end{aligned} \quad (5.79)$$

Let $f_m \ll f_c$ be the highest frequency contained in the signal, i.e., $S(f) = 0$ for $|f| > f_m$ (1st panel in Fig.5.11), then the bandwidth occupied by the AM signal is $\Delta f = 2f_m$ ($f_c \pm f_m$ and $-f_c \pm f_m$) (3rd panel in Fig.5.11). The AM signal is transmitted and then received by a radio receiver, where the audio signal is separated from the carrier wave by a *demodulation* process, which is essentially implemented by another multiplication (4th panel of Fig.5.11):

$$y(t) = x(t) \cos(2\pi f_c t) = s(t) \cos^2(2\pi f_c t) = \frac{s(t)}{2} + \frac{s(t) \cos(4\pi f_c t)}{2} \quad (5.80)$$

To obtain the audio signal $s(t)$, a low-pass filter is used to remove the higher frequency components centered around $\pm 2f_c$, while the audio signal centered around the origin $f = 0$ is further amplified and then sent to the speaker.

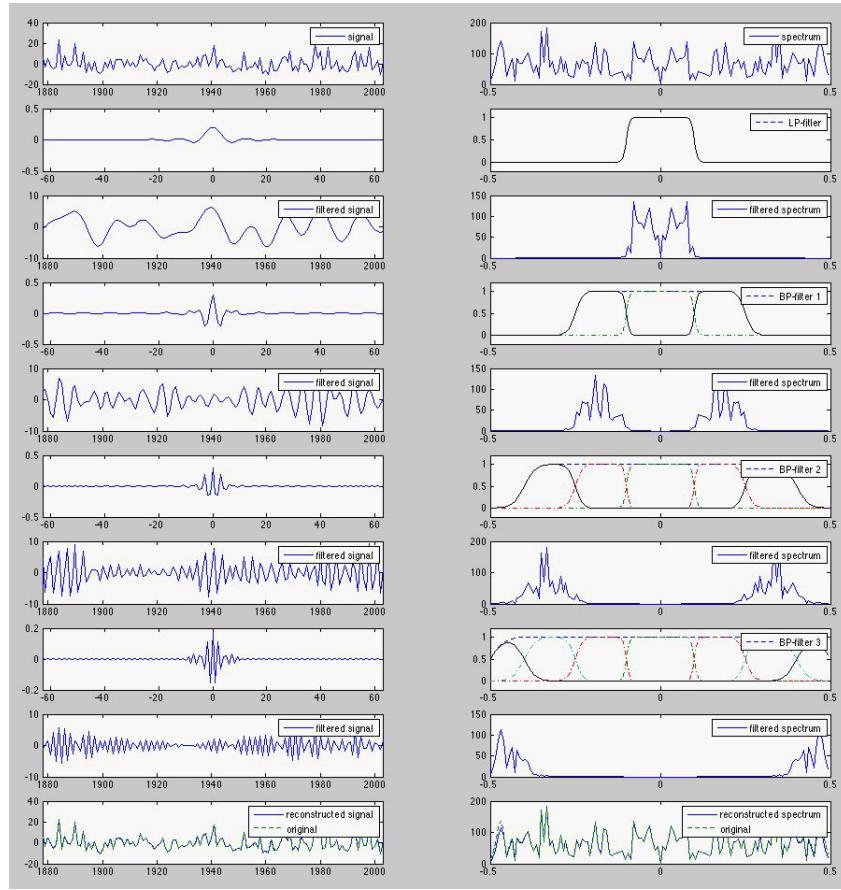


Figure 5.9 Annual precipitation from 1878 to 2003 (left) and its spectrum (right)

Here only the magnitude of each spectrum is shown while the phase is neglected. For each filter, the impulse response function $h(f)$ and the filtered time signal $y(t) = h(t) * x(t)$ are shown on the left, while the frequency response function $H(f)$ and the filtered signal spectrum $Y(f) = H(f)X(f)$ are shown on the right. The dashed curves in the plots on the right show all previous filters and their partial sum.

This process of both modulation and demodulation in frequency domain is illustrated in Fig.5.10 for an artificial signal with a triangular spectrum and also in Fig.5.11 for a real music signal.

Example 5.5: A two-dimensional shape in an image can be described by all the pixels along its boundary, in terms of there coordinates $(x[n], y[n])$, $(n = 1, \dots, N)$, where N is the total number of pixels along the boundary. The coor-

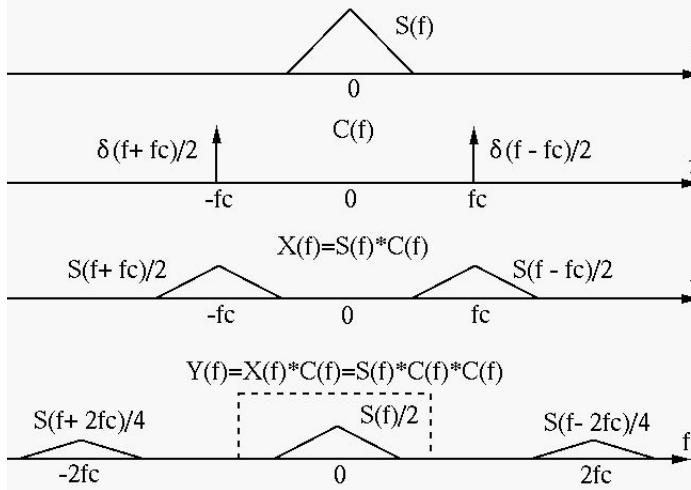


Figure 5.10 AM modulation and demodulation

In top-down order: the audio signal, the carrier sinusoid, the AM signal, and its demodulation and lowpass filtering.

dinates $x[n]$ and $y[n]$ can be treated, respectively, as the real and imaginary components of a complex number $z[n] = x[n] + j y[n]$, and the Fourier transform can be carried out to obtain the Fourier coefficients, called the *Fourier descriptors* of the shape:

$$Z[k] = \frac{1}{\sqrt{N}} \sum_{n=1}^N z[n] e^{-j2\pi nk/N}, \quad k = 1, \dots, N \quad (5.81)$$

Based on all N of these coefficients $Z[k]$, the original shape can be perfectly reconstructed by inverse Fourier transform:

$$z[n] = \frac{1}{\sqrt{N}} \sum_{k=1}^N Z[k] e^{j2\pi nk/N}, \quad n = 1, \dots, N \quad (5.82)$$

It is interesting to observe the reconstructed shape using only the first $M < N$ low frequency components. Note that the inverse transform with M components needs to contain both positive and negative terms symmetric to the DC component in the middle:

$$\hat{z}[n] = \sum_{k=-M/2}^{M/2} Z[k] e^{j2\pi nk/N} \quad (n = 1, \dots, N) \quad (5.83)$$

As an example, the shape of Gumby shown in Fig.5.12 is represented by a chain of $N = 1,157$ pixels along the boundary, in terms of their coordinates $\{x[n], y[n]\}$ ($n = 0, 1, \dots, N - 1$), which are then Fourier transformed to obtain the same number of Fourier coefficients as the Fourier descriptors of the Gumby figure.

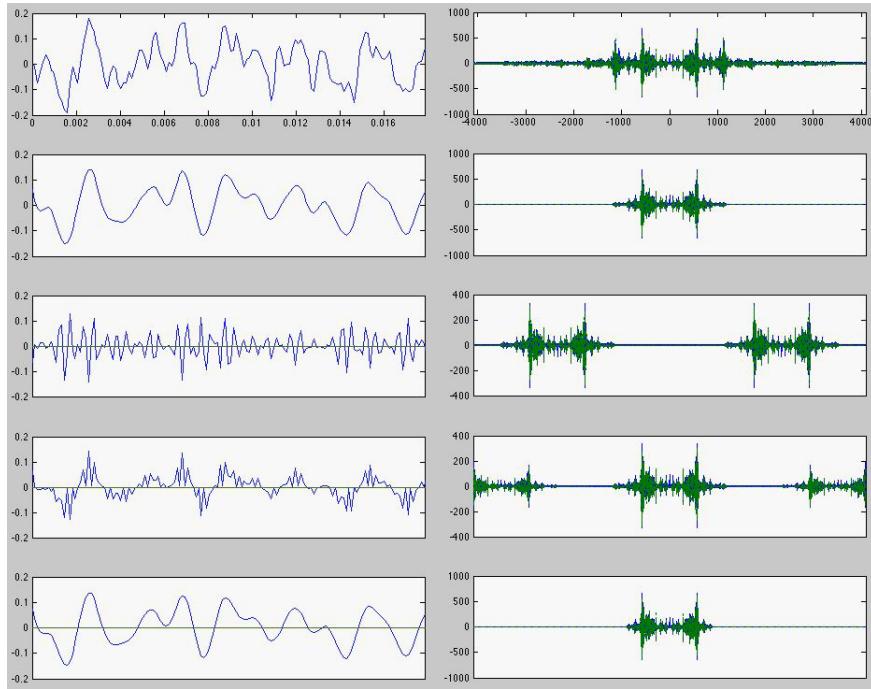


Figure 5.11 AM modulation and demodulation of a real signal

The signals are shown on the left while their spectra are on the right. In top-down: the original signal, the low-pass filtered signal, the AM modulated signal, the demodulated signal and low-pass filtered signal.

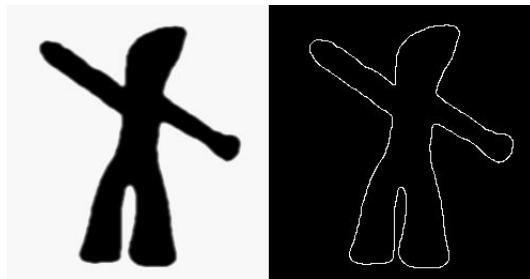


Figure 5.12 Gumby (left) and its boundary pixels (right)

The two different representations of the shape, $z[n]$ in spatial domain and $Z[k]$ in frequency domain are plotted in Fig. 5.13. Note that as the magnitudes of a small number of complex coefficients for DC and some low frequency components are much larger than the rest of the coefficients, a mapping $y = x^{0.5}$ is applied to the magnitudes of all DFT coefficients, so that those coefficients with small magnitudes do not appear to be zero in the plots. The reconstructed shapes corresponding to different M values are shown in Fig. 5.14. We see that

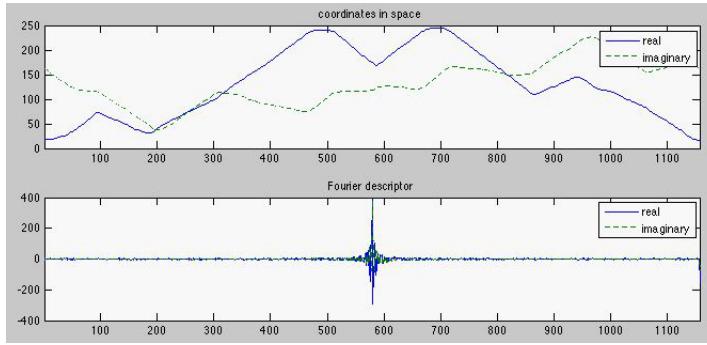


Figure 5.13 The vertical and horizontal components of 2-D shape (top) and its Fourier descriptors (bottom)

the original shape can be almost perfectly reconstructed using only the first few tens of the frequency components. For example, the second to the last figure in the bottom row reconstructed based on the first $M = 30$ components looks almost identical to the last figure based on all $N = 1,257$ components, except the latter may have some very minor details of the shape, such as the sharper corners corresponding to very high frequency components. This result shows that the remaining $N - M = 1,257 - 30 = 1,127$ frequency components contain little information, and can therefore be neglected (treated as zero) in the inverse DFT with little effect in terms of the quality of the reconstruction. Moreover, it may be beneficial to remove the higher frequency components anyway as they are likely to be caused by some random noise instead of the signal of interest.

Some observations can be made based on this example.

- A few coefficients corresponding to mostly low frequency components have significantly higher magnitudes than the rest, indicating that most of the signal energy is concentrated around the low frequency region of the spectrum. This phenomenon is common in general, due to the fact that in most physical signals relatively slow changes over time or space are more significant compared to rapid and sudden changes, i.e., they tend to be continuous and smooth due to their physical nature.
- The plots of the x and y-coordinates in space are much smoother compared to the real and imaginary parts of the Fourier coefficients. Given a signal value $x[n]$ at position n , one can estimate the value $x[n + 1]$ at the next position with reasonable confidence. However, this is not the case in spatial frequency domain. The magnitudes of the DFT coefficients seem random. Given $X[k]$, one has little idea about the next value $X[k + 1]$. In other words, the signal is highly correlated in spatial domain but significantly decorrelated in frequency domain after the Fourier transform.
- As most of the signal energy is concentrated in a small number of low frequency components, little error will result if only $M \ll N$ of the coefficients corresponding to low frequencies are used in the inverse DFT for the recon-

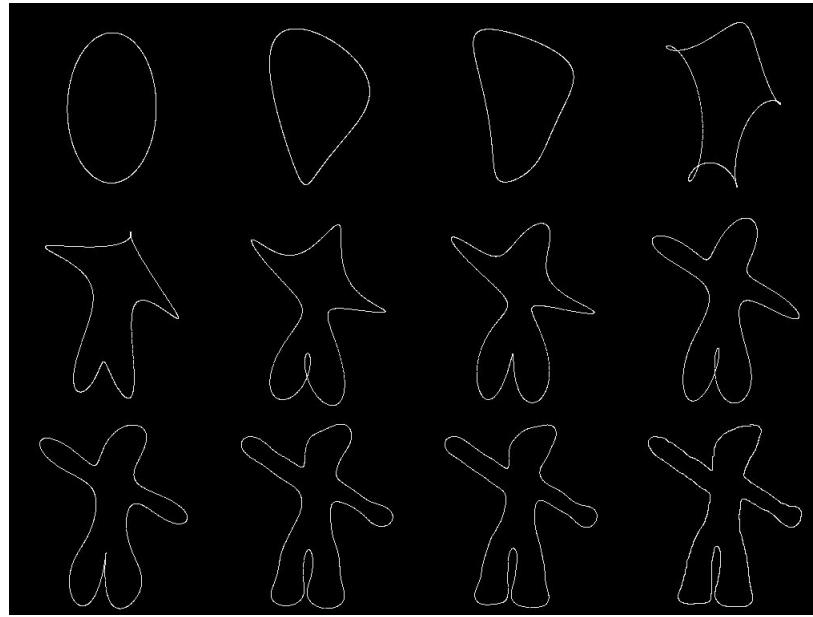


Figure 5.14 Reconstructions of Gumby based on the first M frequency components
Top: $M = 1, 2, 3$ and 4 ; Middle: $M = 5, 6, 7$, and 8 (middle); Bottom: $M = 10, 20, 30$ and $M = N = 1,257$.

struction of the figure in space. Such a low-pass filtering may also have the effect of removing unwanted high frequency noise.

This example illustrates some general applications of the Fourier transform, namely, information extraction and data compression. Useful features contained in a signal, such as the basic shape of a figure in an image, may be extracted by keeping a small number of the Fourier coefficients with most others ignored. It is possible to process, store and transmit only a small portion of the data without losing much information. Moreover, the observations made here for the Fourier transform are also valid in general for all other orthogonal transforms, as we will see in later chapters.

5.5 Implementation of 2-D Filtering

The filtering of 2-D spatial signal $f(x, y)$ (e.g., an image) can be carried out in frequency domain by multiplying its spectrum $F(u, v)$ by the FRF $H(u, v)$ of a filter:

$$G(u, v) = H(u, v) F(u, v) \quad (5.84)$$

The filtered spectrum can then be inverse transformed back to spatial domain to get the filtered signal:

$$g(x, y) = \mathcal{F}^{-1}[G(u, v)] \quad (5.85)$$

We consider below a few 2-D filters which are 2-D extension of the 1-D filters discussed above. These filters are centrally symmetric, and all of them keep the frequency components around the central area unchanged and suppress the frequency components farther away from the center around the corners and edges of the 2-D discrete spectrum. They are low-pass filters if the 2-D spectrum is centralized (Eq. 4.266) so that the DC component $F(0, 0)$ at the origin $u = v = 0$ is in the middle of the spectrum, and the distance of any frequency component $F(u, v)$ to the origin is simply $\sqrt{u^2 + v^2}$.

- **Ideal filter**

$$H_{ideal}(u, v) = \begin{cases} 1 & \sqrt{u^2 + v^2} < w_c \\ 0 & \text{otherwise} \end{cases} \quad (5.86)$$

where w_c is to a cut-off frequency. Ideal filter completely removes any frequency components outside the circle determined by the cut-off frequency. Similar to the 1-D case, some severe ringing artifacts will be caused in 2-D ideal LP filtering.

- **Gaussian filter**

$$H_{gauss}(u, v) = \exp[-a(u^2 + v^2)/w_c^2] \quad (5.87)$$

where $a = \ln 2/2$ so that at the cut-off frequency $u^2 + v^2 = w_c^2$ we have $H_{gauss}(u, v) = H_{gauss}(0, 0)/\sqrt{2} = 1/\sqrt{2}$.

- **Butterworth filter**

$$H_{butterworth}(u, v) = \frac{1}{\sqrt{1 + ((u^2 + v^2)/w_c^2)^n}} \quad (5.88)$$

where w_c is the cut-off frequency at which $|H(u, v)| = 1/\sqrt{2}$ (when $u^2 + v^2 = w_c^2$). When the order n of the Butterworth filter is low it is smooth but when $n \rightarrow \infty$, the Butterworth filter approaches an ideal filter.

These filters are shown in Fig. 5.15, in terms of their impulse response functions (right) in spatial domain as well as the frequency response function (left) in frequency domains.

These filters can be readily used for high-pass filtering in either of two ways. First, if the spectrum is not centralized, then the high frequency components around the middle area of the spectrum will be mostly kept unchanged while the low frequency components farther away from the center are reduced by these filters. Alternatively, corresponding to each LP-filter $H_{lp}(u, v)$ above, a HP-filter can be easily obtained as $H_{hp}(u, v) = 1 - H_{lp}(u, v)$ for a centralized spectrum.

We also note that if the 2-D signal is real, the real and imaginary parts of its spectrum are respectively even and odd, and when it is filtered by any of the

central symmetric filters above, the even/odd symmetry of the spectrum is to be maintained, and the filtered signal obtained by inverse transform remains real. Any filter that fails to maintain the even/odd symmetry of the spectrum of a real signal will necessarily cause the output to be complex.

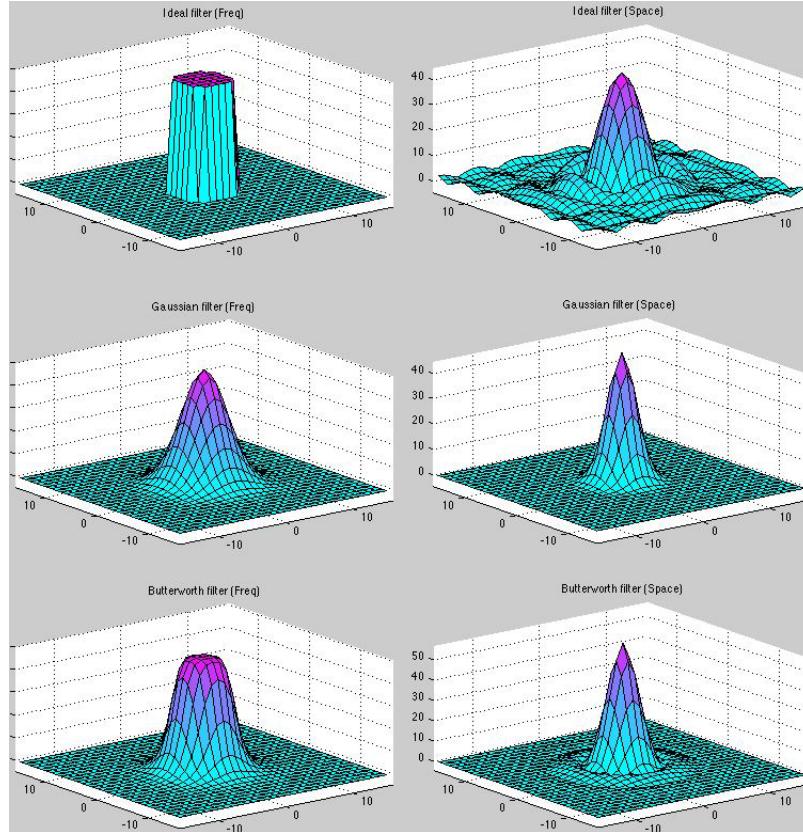


Figure 5.15 2-D filters in both frequency (left) and spatial (right) domains
From top down: deal, Gaussian and Butterworth LP-filters

Example 5.6: Consider the Fourier transform of a 2-D signal shown in the left panel of Fig.5.16. Here the image of a panda is treated as the real part of a 2-D complex signal, while the imaginary part is set to zero. The real (even) and imaginary (odd) parts of the spectrum are shown also in image forms in the middle and right panel in the figure respectively.

As the signal energy is mostly concentrated in a small number of low frequency components around the DC component (typical for most 2-D signals), they show up as a bright spot in the middle of the centralized spectrum, while the rest of the image corresponding to higher frequency components containing little energy

appears dark. In order for all frequency components to be visible, a nonlinear mapping $y = x^\alpha$ ($\alpha = 0.3$ in this case) is applied to all pixel values of the image, so that the low pixel values representing frequency components of low magnitudes are relatively enhanced and become visible in the image. The spectrum can also be represented alternatively in terms of its magnitude and phase components, as shown in Fig.5.17, where two images, one of a panda and the other a cat, together with the magnitude and phase of their corresponding spectra are shown (the first three panels of both rows).

Obviously the real and imaginary parts of the spectrum are equally important in terms of the amount of information they each carry to represent the image signal. But are the magnitude and phase components of the spectrum also equally important in this regard? To answer this question, two images, a panda and a cat, are reconstructed based on the magnitude of the spectrum of one image but the phase of the other, as shown in the two panels on the right in Fig.5.17, where the top image is based on the phase of the panda, while the bottom one is based on the phase of the cat. As an image so reconstructed always looks similar to the image whose phase is used in the reconstruction, it is obvious that the phase component plays a more significant and dominant role than the magnitude components. This result can be easily understood in light of the previous discussion regarding linear phase filtering. Specifically, if the relative positions of all frequency components of a signal remain unchanged by a linear phase filter, then the waveform of a signal remains the same (although they may all be delayed by the same amount of time), otherwise distortion will result if the signal is filtered by a nonlinear phase filter. In other words, the phases of the frequency components are more essential in terms of maintaining the waveform of a signal, in comparison with their magnitudes.

For this reason, the real and imaginary parts $Re[X]$ and $Im[X]$ of the spectrum should always be filtered identically so that the phase angle $\angle X = \tan^{-1} Im[X]/Re[X]$ of each frequency component remains the same, and so does the relative positions of different frequency components, thereby the waveform of the signal is only modified by the magnitude of the filter as desired.

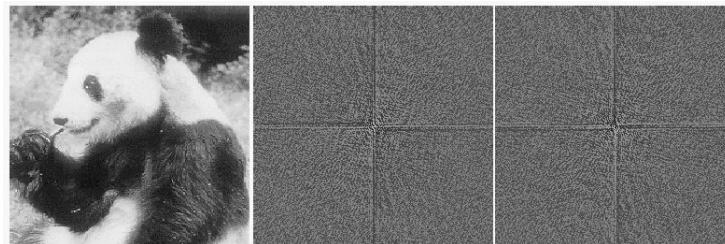


Figure 5.16 An image (left) and the real (middle) and imaginary (right) parts of its Fourier spectrum

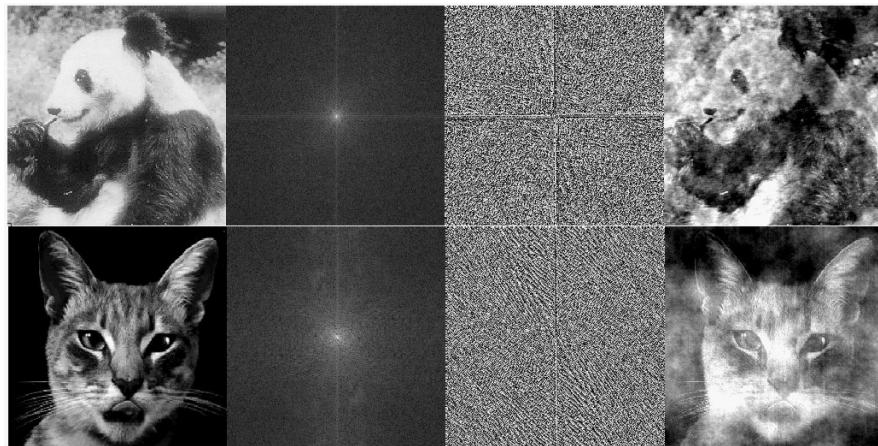


Figure 5.17 Magnitude and phase of Fourier spectra

Two images are shown on the left, and the magnitude and phase of their corresponding spectra are respectively shown in the 2nd and 3rd panels from left. The top-right image is reconstructed based on the phase of panda but magnitude of cat, while the bottom-right image is reconstructed based on the phase of cat but magnitude of panda.

Example 5.7: In this example we illustrate the effect of different types of filtering of an image shown in the previous example in Fig. 5.16. First, the effects of ideal filtering are shown in Fig. 5.18. Corresponding to such filtering in frequency domain shown in the top row, the original image in spatial domain is convolved with a 2-D sinc function, the inverse DFT of the ideal low-pass filter (Eq.4.237), as shown in the bottom row. Note that in either LP and HP cases the filtered images have some obvious ringing artifacts caused by the convolution with the ringing sinc function. If the Butterworth filter without sharp edges is used instead, the filtered images no longer suffer from the ringing artifacts, as shown in Fig. 5.19.

Moreover, in 2-D filtering we can also modify the coefficients for different frequency components in terms of their spatial directions as well as their spatial frequencies. In Fig. 5.20, the 2-D spectrum of the image of panda is low-pass filtered in four different directions: N-S, NW-SE, E-W, and NE-SW (top). In the corresponding images reconstructed by the inverse transform of each directionally low-pass filtered spectrum (bottom), the image features in the orientation favored by the directional filtering are emphasized. Note that all of these four directional filters maintain the even/odd symmetry of the spectrum of the real image.

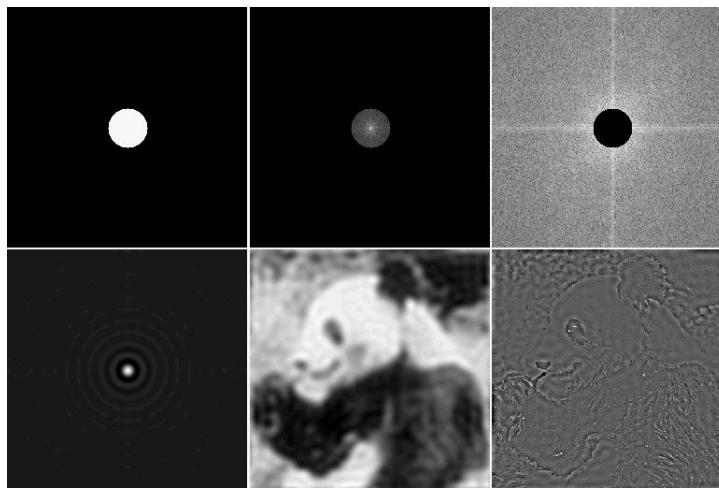


Figure 5.18 Ideal filtering of an image

This figure shows an ideal filter (left) and the low-pass (middle) and high-pass (right) filtered images. The top row shows the spectra of the filter and the filtered images in frequency domain, while the bottom row shows the corresponding images in spatial domain.

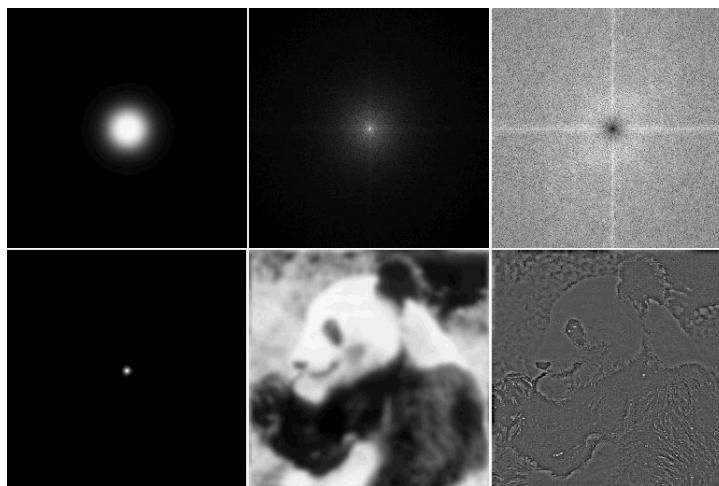


Figure 5.19 Butterworth filtering (from left to right, ideal filter, low-pass, high-pass)

Example 5.8: The example shown in Fig. 5.21 illustrates why the Fourier transform can be used for data compression. After 80% of the DFT coefficients with magnitudes less than a certain threshold value (corresponding mostly to high frequency components) are set to zero (upper right panel), the image is reconstructed based on the remaining 20% of the coefficients still containing over 99% of the signal energy (lower right panel). We see that the reconstructed

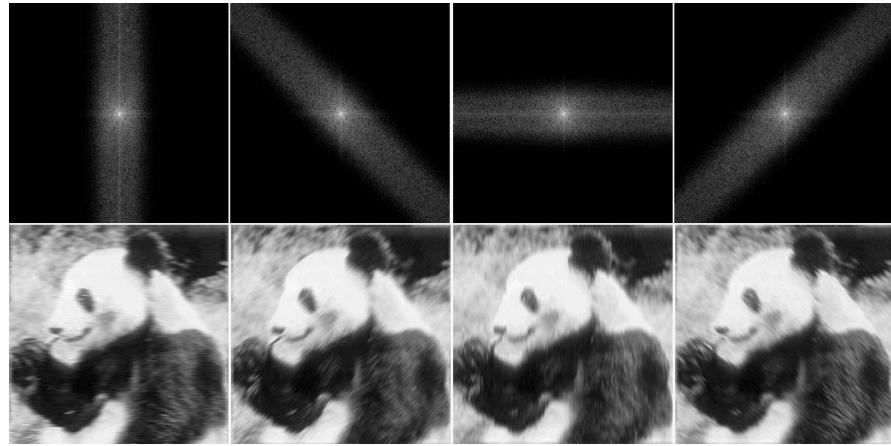


Figure 5.20 Directional low-pass filtering

image looks very much the same as the original one except some very fine details (e.g., the fur on the left arm) corresponding to those high frequency components suppressed.

Why can we throw away 80% of the coefficients but still keep over 99% of the energy in frequency domain, while it is highly unlikely to do so in spatial domain? This is obviously due to the two general properties of all orthogonal transforms: (a) decorrelation of signal components, and (b) compaction of signal energy. Of course this is an over-simplified example only to illustrate the basic ideas of transform based data compression. In practice, there are some other aspects in a compression process such as the quantization and encoding of frequency components. Interested reader can do some further reading about image compression standards, such as the JPEG.

5.6 Hilbert Transform and Analytic Signals

The *Hilbert transform* of a time function $x(t)$ is another time function, denoted by $\hat{x}(t)$, defined as the following convolution with $1/\pi t$:

$$\mathcal{H}[x(t)] = \hat{x}(t) = x(t) * \frac{1}{\pi t} = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(\tau)}{t - \tau} d\tau = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(t - \tau)}{\tau} d\tau \quad (5.89)$$

As the integrand is not integrable due to its pole at $\tau = 0$, the integral of the Hilbert transform is defined in the sense of the *Cauchy principal value* of the integral as:

$$\mathcal{H}[x(t)] = \frac{1}{\pi} \lim_{\epsilon \rightarrow 0} \left[\int_{-\infty}^{-\epsilon} \frac{x(t - \tau)}{\tau} d\tau + \int_{\epsilon}^{\infty} \frac{x(t - \tau)}{\tau} d\tau \right] \quad (5.90)$$

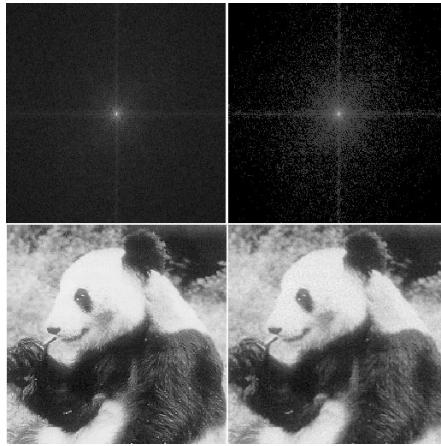


Figure 5.21 Image Compression based on DFT

An image (lower left) and its 2-D DFT spectrum (upper left), together with the reconstructed image (lower right) based on 20% of its DFT coefficients containing 99% of the total energy (upper right).

In particular, if $x(t) = c$ is a constant, the sum of the two integrals above is zero, indicating the Hilbert transform will remove the DC component of the signal. The Hilbert transform can be more conveniently studied in frequency domain as a multiplication corresponding to the time convolution in Eq.5.89. First, to find the spectrum of $1/\pi t$, we apply the property of time-frequency duality to the Fourier transform of the sign function $\text{sgn}(t)$ (Eq.3.76) and get

$$\mathcal{F}\left(\frac{1}{\pi t}\right) = -j \text{Sgn}(f) = -j \begin{cases} -1 & (f < 0) \\ 0 & (f = 0) \\ 1 & (f > 0) \end{cases} = \begin{cases} j & (f < 0) \\ 0 & (f = 0) \\ -j & (f > 0) \end{cases} \quad (5.91)$$

Now the Hilbert transform $\hat{x}(t) = x(t) * 1/\pi t$ can be expressed in frequency domain as a multiplication:

$$\hat{X}(f) = \mathcal{F}[\hat{x}(t)] = [-j \text{Sgn}(f)] X(f) = \begin{cases} jX(f) & (f < 0) \\ 0 & (f = 0) \\ -jX(f) & (f > 0) \end{cases} \quad (5.92)$$

The effect of the Hilbert transform applied of a signal $x(t)$ becomes clear: it multiplies the negative part of the signal spectrum $X(f)$ by $j = e^{j\pi/2}$, (a rotation of $\pi/2$ in complex plane) and the positive part by $-j = e^{-j\pi/2}$ (a rotation of $-\pi/2$). Therefore the Hilbert transform is also called a *quadrature filter*.

As the Hilbert transform of a time function is still a time function, it can be applied to a signal $x(t)$ multiple times, and the result is most conveniently obtained in frequency domain:

$$\mathcal{F}[\mathcal{H}^n[x(t)]] = [-j \text{Sgn}(f)]^n X(f) \quad (5.93)$$

In particular, as $Sgn^2(f) = 1$, we have

$$[-j Sgn(f)]^2 = -1, \quad [-j Sgn(f)]^3 = j Sgn(f), \quad [-j Sgn(f)]^4 = 1 \quad (5.94)$$

Correspondingly in time domain, we have:

$$\mathcal{H}[x(t)] = \hat{x}(t), \quad \mathcal{H}^2[x(t)] = -x(t), \quad \mathcal{H}^3[x(t)] = -\hat{x}(t), \quad \mathcal{H}^4[x(t)] = x(t) \quad (5.95)$$

We see that applying the Hilbert transform to $x(t)$ once we get $\mathcal{H}[x(t)] = \hat{x}(t)$, and applying the transform three more times we get the original signal back, which is actually the inverse Hilbert transform:

$$\begin{cases} \mathcal{H}[x(t)] = x(t) * 1/\pi t = \hat{x}(t) \\ \mathcal{H}^{-1}[\hat{x}(t)] = \mathcal{H}^3[\hat{x}(t)] = -\mathcal{H}[\hat{x}(t)] = x(t) \end{cases} \quad (5.96)$$

Example 5.9: Consider a simple sinusoid:

$$\cos(2\pi f_0 t) = \frac{1}{2}e^{j2\pi f_0 t} + \frac{1}{2}e^{-j2\pi f_0 t} \quad (5.97)$$

When the Hilbert transform is applied to the signal, the coefficient $1/2$ for $f < 0$ is rotated by 90° to become $e^{j\pi/2}/2 = -1/2j$ while the other coefficient $1/2$ for $f > 0$ is rotated by -90° to become $e^{-j\pi/2}/2 = 1/2j$, and the transformed signal becomes:

$$\mathcal{H}[\cos(2\pi f t)] = \frac{1}{2j}e^{j2\pi f_0 t} - \frac{1}{2j}e^{-j2\pi f_0 t} = \sin(2\pi f t) \quad (5.98)$$

Similarly we have $\mathcal{H}[\sin(2\pi f t)] = -\cos(2\pi f t)$, $\mathcal{H}[-\cos(2\pi f t)] = -\sin(2\pi f t)$ and $\mathcal{H}[-\sin(2\pi f t)] = \cos(2\pi f t)$.

Next we consider the concept of analytic signals. A real-valued signal $x_a(f)$ is said to be *analytic* if its Fourier spectrum $X_a(f) = \mathcal{F}[x_a(t)]$ is zero when $f < 0$. Any signal $x(t)$ can be turned into an analytic signal by multiplying its spectrum $X(f) = \mathcal{F}[x(t)]$ with a step function $2u(f)$ in frequency domain:

$$X_a(f) = X(f)2u(f) = \begin{cases} 0 & (f < 0) \\ X(0) & (f = 0) \\ 2X(f) & (f > 0) \end{cases} \quad (5.99)$$

Applying the time-frequency duality to the Fourier transform of the unit step in Eq.3.72 we get the inverse Fourier transform of the unit step spectrum $u(f)$:

$$\mathcal{F}^{-1}[u(f)] = \frac{1}{-j2\pi t} + \frac{1}{2}\delta(-t) = \frac{j}{2\pi t} + \frac{1}{2}\delta(t) \quad (5.100)$$

and the analytic signal can be obtained by taking the inverse Fourier transform on both sides of Eq.5.99:

$$\begin{aligned} x_a(t) &= \mathcal{F}^{-1}[X_a(f)] = \mathcal{F}^{-1}[X(f)] * \mathcal{F}^{-1}[2u(f)] = x(t) * [\delta(t) + \frac{j}{\pi t}] \\ &= x(t) + j x(t) * \frac{1}{\pi t} = x(t) + j \hat{x}(t) \end{aligned} \quad (5.101)$$

Alternatively, an analytic signal can also be initially defined in time domain by Eq.5.101, and if we take the Fourier transform on both sides, we have

$$X_a(f) = X(f) + j \hat{X}(f) = X(f) + j \begin{cases} jX(f) & (f < 0) \\ 0 & (f = 0) \\ -jX(f) & (f > 0) \end{cases} = \begin{cases} 0 & (f < 0) \\ X(0) & (f = 0) \\ 2X(f) & (f > 0) \end{cases} \quad (5.102)$$

where $\hat{X}(f) = \mathcal{F}[\hat{x}(t)]$.

When the signal $x(t)$ is real, its spectrum $X(f)$ satisfies $X(f) = \overline{X}(-f)$, indicating the corresponding analytic signal $x_a(t) = x(t) + j \hat{x}(t)$ contains the complete information in $x(t)$, even though the negative half of its spectrum is suppressed to zero. In fact the original spectrum $X(f)$ can also be reconstructed from $X_a(f)$. When $f > 0$, obviously we get $X(f) = X_a(f)/2$, when $f < 0$, we have:

$$X(f) = \overline{X}(-f) = \overline{X}(|f|) = \frac{1}{2} \overline{X}_a(|f|) \quad (5.103)$$

Combining these two cases, we have:

$$X(f) = \frac{1}{2} \begin{cases} X_a(f) & (f > 0) \\ \overline{X}_a(|f|) & (f < 0) \end{cases} = \frac{X_a(f) + \overline{X}_a(-f)}{2} \quad (5.104)$$

the second equality is due to the fact that $\overline{X}_a(-f) = 0$ when $f > 0$ and $X_a(f) = 0$ when $f < 0$.

Example 5.10: In Example 5.4 concerning the AM modulation and demodulation, the bandwidth $\Delta f = 2f_m$ is twice the highest frequency f_m contained in the signal, one sideband of f_m on each side of the carrier frequency f_c (double sideband). In order to efficiently use the broadcast spectrum as a limited resource, it is desirable to minimize the bandwidth needed for each broadcast transmission. The *single-sideband modulation (SSB)* is such a method by which the bandwidth is reduced by half (from $2f_m$ to f_m). One implementation of the SSB is based on the Hilbert transform and analytic signals, taking advantage of the fact that the negative half of the spectrum of an analytic signal is always zero and therefore does not need to be transmitted. Specifically, an analytic signal is first constructed based on the signal $s(t)$ to be transmitted:

$$s_a(t) = s(t) + j \hat{s}(t) \quad (5.105)$$

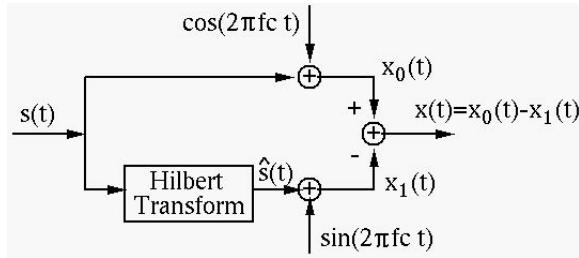


Figure 5.22 Single sideband modulation using Hilbert transform

where $\hat{s}(t) = \mathcal{H}[x(t)]$ is the Hilbert transform of $s(t)$. Then $s_a(t)$ is used to modulate a carrier frequency represented as an complex exponential $e^{j2\pi f_c t}$. The real part of the resulting AM signal $s_a(t)e^{j2\pi f_c t}$ is then transmitted:

$$\begin{aligned} x(t) &= \operatorname{Re}[s_a(t)e^{j2\pi f_c t}] = \operatorname{Re}[(s(t) + j\hat{s}(t))(\cos(2\pi f_c t) + j\sin(2\pi f_c t))] \\ &= s(t)\cos(2\pi f_c t) - \hat{s}(t)\sin(2\pi f_c t) = x_0(t) - x_1(t) \end{aligned} \quad (5.106)$$

where $x_0(t) = s(t)\cos(2\pi f_c t)$ and $x_1(t) = \hat{s}(t)\sin(2\pi f_c t)$ are two modulated RF signals with 90° phase difference. The block diagram of the single sideband modulation is illustration in Fig.5.22. In frequency domain Eq.5.106 becomes:

$$\begin{aligned} X(f) &= X_0(f) - X_1(f) \\ &= S(f) * \frac{1}{2}[\delta(f - f_c) + \delta(f + f_c)] - \hat{S}(f) * \frac{1}{2j}[\delta(f - f_c) - \delta(f + f_c)] \\ &= \frac{1}{2}[S(f - f_c) + S(f + f_c) + j\hat{S}(f - f_c) - j\hat{S}(f + f_c)] \end{aligned} \quad (5.107)$$

Note that $\hat{S}(f - f_c)$ and $\hat{S}(f + f_c)$ are related to $S(f - f_c)$ and $S(f + f_c)$ by Eq.5.92, and in two of the following four cases, they cancel each other:

$$\begin{aligned} f + f_c < 0 : \hat{S}(f + f_c) &= jS(f + f_c), \quad X(f + f_c) = 2S(f + f_c) \\ f + f_c > 0 : \hat{S}(f + f_c) &= -jS(f + f_c), \quad X(f + f_c) = 0 \\ f - f_c < 0 : \hat{S}(f - f_c) &= jS(f - f_c), \quad X(f - f_c) = 0 \\ f - f_c > 0 : \hat{S}(f - f_c) &= -jS(f - f_c), \quad X(f - f_c) = 2S(f - f_c) \end{aligned} \quad (5.108)$$

The spectra of the signals in the process are shown in Fig.5.23, from which we see that the bandwidth of this modulated signal $x(t)$ is indeed reduced by half. The SSB modulation is carried out on a real music signal as shown in Fig.5.24, where the signal and its spectrum at various stages of the process are shown on the left and right respectively.

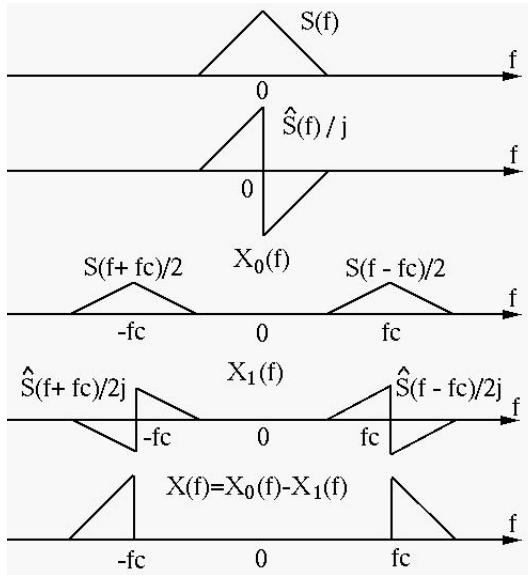


Figure 5.23 The SSB modulation in frequency domain

5.7 Radon Transform and Image Restoration from Projections

Like the Fourier transform, the Radon transform is also an integral transform, as illustrated in Fig.5.25, that integrates a 2-D function $f(x, y)$ along a straight line $L(\theta)$ specified by an angle θ (measured from the positive direction of x). The resulting 1-D function $g_\theta(s)$ of s , the distance between the origin and line $L(\theta)$, is in fact the projection of $f(x, y)$ onto a straight line in the direction of s . In particular, if the direction is along either x or y (corresponding to $\theta = 0$ or $\theta = \pi/2$), we get:

$$g(y) = \int_{-\infty}^{\infty} f(x, y) dx, \quad \text{or} \quad g(x) = \int_{-\infty}^{\infty} f(x, y) dy \quad (5.109)$$

The projections along all different directions θ can be considered as a 2-D function $g(s, \theta)$, from which the original 2-D function $f(x, y)$ can be reconstructed by the inverse Radon transform. This forward and inverse Radon transform pair can be expressed as:

$$\begin{cases} g(s, \theta) = \mathcal{R}[f(x, y)] \\ f(x, y) = \mathcal{R}^{-1}[g(s, \theta)] \end{cases} \quad (5.110)$$

The Radon transform is widely used in X-ray computerized tomography (CT) to get the image of a cross section, a slice, of certain part of the body. Moreover, a 3-D volume of data can be obtained as a sequence of such slices along the direction perpendicular to cross sections. Let I_o denote the intensity of the source X-ray and $f(x, y)$ denote the absorption coefficient of the tissue at position (x, y) .

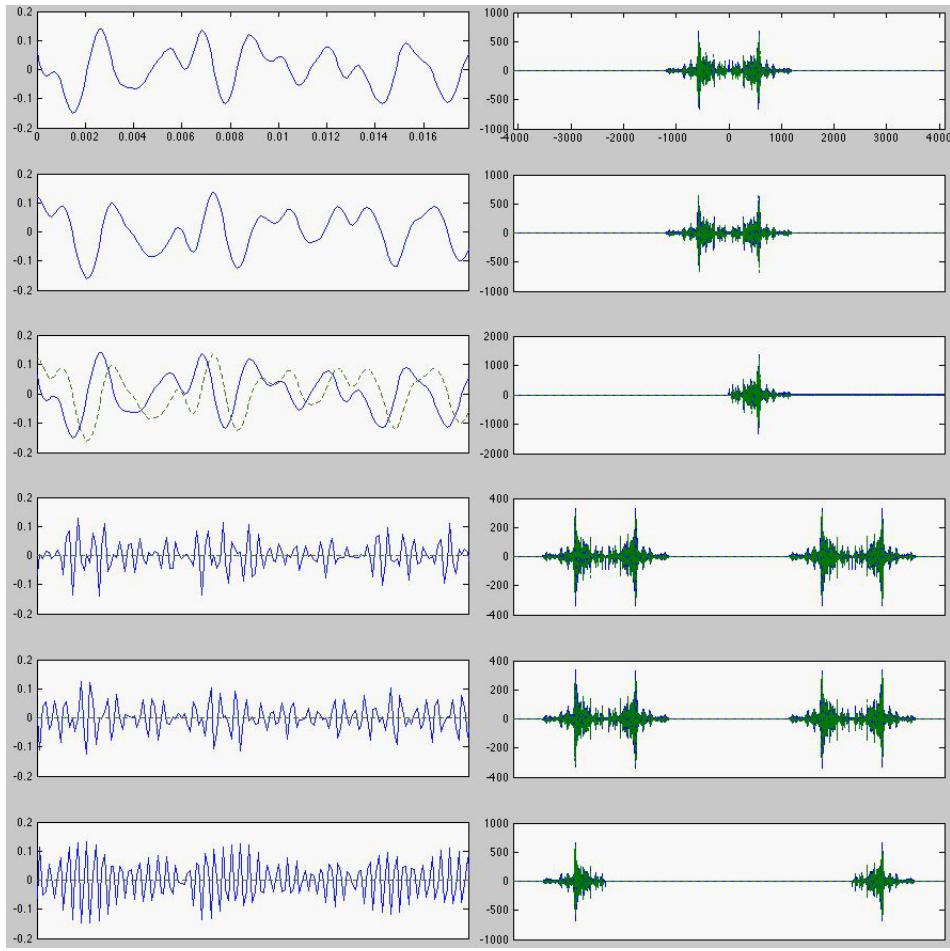


Figure 5.24 The SSB modulation of a music signal

In top-down order: the original signal $s(t)$, its Hilbert transform $\hat{s}(t)$, the corresponding analytic signal $s_a(t) = s(t) + j\hat{s}(t)$ (not its spectrum $S_a(f) = 0$ for $f < 0$), AM modulation of $x_0(t)$ and $x_1(t)$, SSB modulated $x(t) = x_0(t) - x_1(t)$.

The detected signal intensity I can be obtained according to this simple model:

$$I = I_o \exp \left(- \int_{L(\theta)} f(x, y) dt \right) \quad (5.111)$$

Here t is the integral variable along the pathway $L(\theta)$ of the X-ray through the tissue. The exponent, the absorption coefficient integrated along $L(\theta)$, is just the Radon transform $g(s, \theta)$ of $f(x, y)$, which can be obtained given the detected I :

$$g(s, \theta) = \int_{L(\theta)} f(x, y) dt = \ln (I_o/I) \quad (5.112)$$

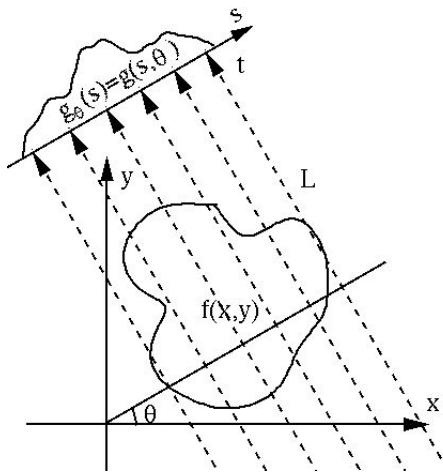


Figure 5.25 Radon transform

and the cross section $f(x, y)$ representing the tissue absorption coefficient can then be obtained by the inverse Radon transform.

Now let us further formulate the Radon transform. The straight line $L(\theta)$ along which the projection of a 2-D function is obtained can be specified by the following equation (i.e., any point (x, y) on $L(\theta)$ satisfies the equation):

$$x \cos \theta + y \sin \theta - s = 0 \quad (5.113)$$

with two parameters s and θ , as shown in Fig.5.26 (left). Now the 1-D integral along $L(\theta)$ of the Radon transform in Eq.5.112 can be written as the following 2-D integral:

$$g(s, \theta) = \mathcal{R}[f(x, y)] = \int \int_{-\infty}^{\infty} f(x, y) \delta(x \cos \theta + y \sin \theta - s) dx dy \quad (-\infty < s < \infty, 0 \leq \theta < 2\pi) \quad (5.114)$$

which converts the 2-D spatial function $f(x, y)$ into a function $g(s, \theta)$ in a 2-D parameter space.

Next we define a new coordinate system (s, t) in the 2-D space by rotating the (x, y) coordinate system by an angle θ :

$$\begin{cases} s = x \cos \theta + y \sin \theta \\ t = -x \sin \theta + y \cos \theta \end{cases} \quad \text{or} \quad \begin{cases} x = s \cos \theta - t \sin \theta \\ y = s \sin \theta + t \cos \theta \end{cases} \quad (5.115)$$

where t is the coordinate along the direction of the projection line $L(\theta)$, perpendicular to the direction of s . Note that this rotation is a unitary transformation which conserves vector norm, i.e., $x^2 + y^2 = s^2 + t^2$. In the new (s, t) coordinate system, the Radon transform can be expressed as a 1-D integral along the

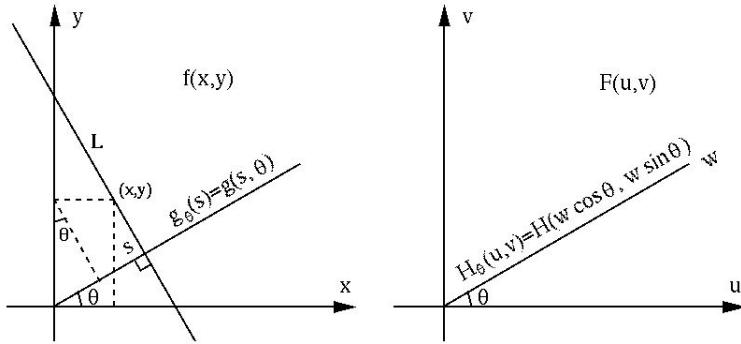


Figure 5.26 Radon transform and projection-slice theorem

direction of t :

$$g(s, \theta) = \mathcal{R}[f(x, y)] = \int_{-\infty}^{\infty} f(s \cos \theta - t \sin \theta, s \sin \theta + t \cos \theta) dt \quad (5.116)$$

Example 5.11: First consider the Radon transform of a 2-D Gaussian function $f(x, y) = e^{-(x^2+y^2)} = e^{-(s^2+t^2)}$:

$$g(s, \theta) = \int_{-\infty}^{\infty} e^{-(s^2+t^2)} dt = e^{-s^2} \int_{-\infty}^{\infty} e^{-t^2} dt = \sqrt{\pi} e^{-s^2} \quad (5.117)$$

We see that $g(s, \theta)$ is a 1-D Gaussian function of s , independent of θ , as a 2-D Gaussian function is central symmetric.

Next consider the Radon transform of a plane wave

$$f(x, y) = \cos(2\pi(2x + 3y)) = \frac{1}{2}[e^{j2\pi(2x+3y)} + e^{-j2\pi(2x+3y)}] \quad (5.118)$$

which propagates along the direction of $\phi = \tan^{-1}(3/2)$ (with respect to the horizontal direction). As the Radon transform is obviously linear, we can find the transforms of $e^{j2\pi(2x+3y)}$ and $e^{-j2\pi(2x+3y)}$ separately. The first term can be expressed in terms of the rotated coordinate system (s, t) :

$$\begin{aligned} e^{j2\pi(2x+3y)} &= e^{j2\pi 2x} e^{j2\pi 3y} = e^{j2\pi(2(s \cos \theta - t \sin \theta))} e^{j2\pi(3(s \sin \theta + t \cos \theta))} \\ &= e^{j2\pi 2(2 \cos \theta + 3 \sin \theta)} e^{j2\pi t(-2 \sin \theta + 3 \cos \theta)} \end{aligned}$$

Its Radon transform is:

$$\begin{aligned} \mathcal{G}[e^{j2\pi(2x+3y)}] &= e^{j2\pi s(2 \cos \theta + 3 \sin \theta)} \int_{-\infty}^{\infty} e^{j2\pi t(-2 \sin \theta + 3 \cos \theta)} dt \\ &= e^{j2\pi s(2 \cos \theta + 3 \sin \theta)} \delta(-2 \sin \theta + 3 \cos \theta) \end{aligned} \quad (5.119)$$

Similarly we can get:

$$\mathcal{G}[e^{-j2\pi(2x+3y)}] = e^{-j2\pi s(2 \cos \theta + 3 \sin \theta)} \delta(2 \sin \theta - 3 \cos \theta) \quad (5.120)$$

Adding these two results we get

$$\mathcal{G}[\cos(2\pi(2x + 3y))] = \cos(2\pi s(2\cos\theta + 3\sin\theta))\delta(2\sin\theta - 3\cos\theta) \quad (5.121)$$

We see that this Radon transform is zero except when $2\sin\theta = 3\cos\theta$ or $\theta = \tan^{-1}(3/2) = \phi$, i.e., the straight line $L(\theta)$ for the Radon transform is perpendicular to the propagation direction of the plane wave. In this case the Radon transform is a delta function (due to the infinite integral of a constant along the direction of $L(\theta)$), weighted by a sinusoidal function of s along the direction of propagation. When $\theta \neq \phi$, the integrand in Eq.5.119 along $L(\theta)$ is a sinusoid with frequency $3\cos\theta - 2\sin\theta$, and the infinite integral is always zero.

Projection-slice theorem: The 1-D Fourier transform of the Radon transform $g(s, \theta) = \mathcal{R}[f(x, y)]$ with respect to s (with θ treated as a parameter) is equal to the slice of the 2-D Fourier transform $F(u, v) = \mathcal{F}[f(x, y)]$ through the origin along the direction θ :

$$G(w, \theta) = \mathcal{F}[g(s, \theta)] = F_\theta(u, v) \quad (5.122)$$

where $F_\theta(u, v)$ denotes a slice of $F(u, v)$ through the origin along direction θ .

Proof: First find the 1-D Fourier transform of the Radon transform $g(s, \theta) = \mathcal{R}[f(x, y)]$ with respect to s :

$$G(w, \theta) = \mathcal{F}[g(s, \theta)] = \int_{-\infty}^{\infty} g(s, \theta) e^{-j2\pi ws} ds \quad (5.123)$$

where w is the spatial frequency of $f(x, y)$ along the direction of s . Substituting the expression of $g(s, \theta)$ in Eq.5.114 into the above equation, we get:

$$\begin{aligned} G(w, \theta) &= \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \delta(x \cos\theta + y \sin\theta - s) dx dy \right] e^{-j2\pi ws} ds \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \left[\int_{-\infty}^{\infty} \delta(x \cos\theta + y \sin\theta - s) e^{-j2\pi ws} ds \right] dx dy \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi w(x \cos\theta + y \sin\theta)} dx dy \\ &= F(w \cos\theta, w \sin\theta) = F_\theta(u, v) \end{aligned}$$

where

$$\begin{cases} u = w \cos\theta \\ v = w \sin\theta \end{cases} \quad \text{or} \quad \begin{cases} w = \sqrt{u^2 + v^2} \\ \theta = \tan^{-1}(v/u) \end{cases} \quad (5.124)$$

and $F(w \cos\theta, w \sin\theta) = F_\theta(u, v)$ is the 2-D Fourier transform $F(u, v)$ of the signal $f(x, y)$ evaluated at $u = w \cos\theta$ and $v = w \sin\theta$, along the direction of θ .

Inverse Radon theorem: Given its Radon transform $g(s, \theta)$, the original 2-D signal $f(x, y)$ can be reconstructed by:

$$f(x, y) = \mathcal{R}^{-1}[g(s, \theta)] = \frac{1}{2\pi^2} \int_0^\pi \int_{-\infty}^\infty \left[\frac{\partial}{\partial s} g(s, \theta) \right] \frac{1}{x \cos \theta + y \sin \theta - s} ds d\theta \quad (5.125)$$

or in polar form:

$$f(r, \phi) = \frac{1}{2\pi^2} \int_0^\pi \int_{-\infty}^\infty \left[\frac{\partial}{\partial s} g(s, \theta) \right] \frac{1}{r \cos(\phi - \theta) - s} ds d\theta \quad (5.126)$$

where

$$\begin{cases} x = r \cos \phi \\ y = r \sin \phi \end{cases} \quad \begin{cases} r = \sqrt{x^2 + y^2} \\ \phi = \tan^{-1}(y/x) \end{cases} \quad (5.127)$$

Proof: Based on Eq.5.124, the Fourier spectrum $F(u, v)$ can be written in polar form as $F(w, \theta)$ and the inverse transform $f(x, y) = \mathcal{F}^{-1}[F(u, v)]$ becomes:

$$\begin{aligned} f(x, y) &= \int_{-\infty}^\infty \int_{-\infty}^\infty F(u, v) e^{j2\pi(ux+vy)} du dv \\ &= \int_0^{2\pi} \int_0^\infty F(w, \theta) e^{j2\pi w(x \cos \theta + y \sin \theta)} w dw d\theta \\ &= \int_0^\pi \int_{-\infty}^\infty F(w, \theta) e^{j2\pi w(x \cos \theta + y \sin \theta)} |w| dw d\theta \end{aligned}$$

Here $F(w, \theta)$ is a slice of $F(u, v)$ along the direction θ , which, according to the projection-slice theorem Eq.5.122, is equal to the Fourier transform of the Radon transform of $f(x, y)$, i.e., $F(w, \theta) = G(w, \theta) = \mathcal{F}[g(s, \theta)]$, then the equation above becomes:

$$\begin{aligned} f(x, y) &= \int_0^\pi \left[\int_{-\infty}^\infty |w| G(w, \theta) e^{j2\pi w(x \cos \theta + y \sin \theta)} dw \right] d\theta \\ &= \int_0^\pi g'(x \cos \theta + y \sin \theta, \theta) d\theta \end{aligned} \quad (5.128)$$

where $g'(s, \theta)$ is defined as the inverse Fourier transform of $|w|G(w, \theta)$:

$$\begin{aligned} g'(s, \theta) &= g'(x \cos \theta + y \sin \theta, \theta) \\ &= \int_{-\infty}^\infty |w| G(w, \theta) e^{j2\pi w(x \cos \theta + y \sin \theta)} dw = \mathcal{F}^{-1}[|w|G(w, \theta)] \end{aligned}$$

We can consider $|w|G(w, \theta)$ as a filtering process of $g(s, \theta)$ by a filter $|w|$ in frequency domain w , i.e., $g'(s, \theta)$ is the filtered version of $g(s, \theta)$ in space domain s . As $|w|$ can be written as a product $|w| = w \operatorname{sgn}(w)$ (a high-pass filter), the inverse Fourier transform above for $|w|G(w, \theta) = wG(w, \theta)\operatorname{sgn}(w)$ becomes:

$$\begin{aligned} g'(s, \theta) &= \mathcal{F}^{-1}[wG(w, \theta) \operatorname{sgn}(w)] = \mathcal{F}^{-1}[wG(w, \theta)] * \mathcal{F}^{-1}[\operatorname{sgn}(w)] \\ &= \left[\frac{1}{j2\pi} \frac{\partial}{\partial s} g(s, \theta) \right] * \left[\frac{1}{-j\pi s} \right] = \frac{1}{2\pi^2} \int_{-\infty}^\infty \left[\frac{\partial}{\partial t} g(t, \theta) \right] \frac{1}{s-t} dt \end{aligned} \quad (5.129)$$

Here we have used the convolution theorem and also Eqs.3.117 and 3.137 for the two inverse transforms. Comparing this expression with the definition of the Hilbert transform in Eq.5.89, we see that $g'(s, \theta)$ is also the Hilbert transform of $\partial g(s, \theta)/\partial s/2\pi$:

$$g'(s, \theta) = \mathcal{H} \left[\frac{1}{2\pi} \frac{\partial}{\partial s} g(s, \theta) \right] \quad (5.130)$$

Substituting Eq.5.129 back into Eq.5.128 for $f(x, y)$, we get

$$f(x, y) = \frac{1}{2\pi^2} \int_0^\pi \int_{-\infty}^\infty \left[\frac{\partial}{\partial t} g(t, \theta) \right] \frac{1}{s - t} dt d\theta \quad (5.131)$$

Replacing s by $x \cos \theta + y \sin \theta$, we get Eq.5.125. Q.E.D.

In practice, the inverse Radon transform can be carried out based on Eq.5.128, instead of Eq.5.125 or 5.126, in the following steps:

1. Fourier transform of $g(s, \theta)$ with respect to s for all directions θ :

$$G(w, \theta) = \mathcal{F}[g(s, \theta)] \quad (5.132)$$

2. Filtering in frequency domain by $|w|$:

$$G'(w, \theta) = |w| G(w, \theta) \quad (5.133)$$

3. Inverse Fourier transform:

$$g'(s, \theta) = \mathcal{F}^{-1}[G'(w, \theta)] \quad (5.134)$$

4. Summation of $g'(x \cos \theta + y \sin \theta, \theta)$ over all directions θ (called “back projection”):

$$f(x, y) = \int_0^\pi g'(s, \theta) d\theta = \int_0^\pi g'(x \cos \theta + y \sin \theta, \theta) d\theta \quad (5.135)$$

As the higher frequency components of most signals contain little energy and are more susceptible to noise (lower signal-to-noise ratio), the high-pass filter $|w|$ that is likely to amplify noise in the signal is typically modified so that its magnitude is reduced in the high frequency range.

Example 5.12: Consider the Radon transform, both forward transform for projection and the inverse transform for reconstruction, of two 2-D signals, a shape in black-white image and a gray scale image, as shown on the left in Fig.5.27. In each of the two cases, we obtain the projections $g(s, \theta)$ (2nd from left) of all 180 angles, one degree apart, of the image $f(x, y)$, and then reconstruct the image, first without filtering to produce a blurred reconstruction (3rd from left), and then with high-pass filtering by $|w|$ to produce an almost perfectly reconstruction (right).

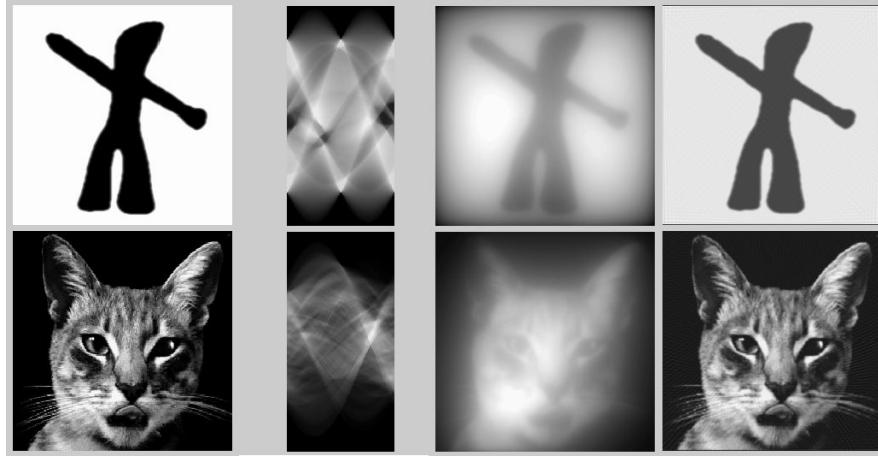


Figure 5.27 Forward and inverse Radon transform

From left to right: Original image $f(x, y)$, Radon projections $g(s, \theta)$, Back project without filtering, Back projection with filtering.

The Matlab code for both forward and inverse Radon transforms is listed below. The projection directions are given in vector theta in degrees.

```

function proj = Radon(im,theta) % forward Radon transform
    K=length(theta);           % number of projection directions
    [m,n]=size(im);           % size of image
    d=fix(sqrt(2)*max(m,n)); % diagonal of image
    tmp=zeros(d);
    i=(d-m)/2;
    j=(d-n)/2;
    tmp(i:i+m-1,j:j+n-1)=im; % copy input image to tmp
    proj=zeros(d,K);          % K projections of length d
    for k=1:K                 % for all directions
        a=theta(k);            % rotation angle
        proj(:,k)=sum(imrotate(tmp,a,'bilinear','crop'));
                                % image rotation and projection
    end
end

function im=iRadon(proj,theta) % inverse Radon transform
    [d,K]=size(proj);         % diagonal of image
    n=ceil(d/sqrt(2));        % size of image
    im=zeros(n);
    n2=n/2;
    d2=d/2;
    v=pi/180;                 % for radian/degree conversion

```

```

F=zeros(d,1); % filter in frequency domain
d1=ceil((d-1)/2);
for i=2:d1+1; % setup filter
    F(i)=i-1;
    F(d+2-i)=i-1;
end
for k=1:K % for all directions
    g=proj(:,k); % g(s,theta)
    G=fft(g); % Fourier transform of g
    G=G.*F; % filtering in frequency domain
    g=real(ifft(G)); % inverse Fourier transform
    c=cos(v*theta(k)); % cos(theta)
    s=sin(v*theta(k)); % sin(theta)
    for i=1:n
        for j=1:n % for all pixels in image
            y=i-n2;
            x=j-n2; % image center is at origin
            t=fix(x*c+y*s)+d2;
            im(i,j)=im(i,j)+g(t); % back projection
        end
    end
end

```

5.8 Orthogonal Frequency Division Modulation (OFDM)

A digital communication system (DCS) transmits and receives messages consisting of a finite number of symbols representing various types of information. Consider the transmission of a block of N_c complex numbers (symbols) $\{d_1, d_2, \dots, d_{N_c}\}$, where $d_k = a_k + jb_k$, during a time interval of T seconds, which can be carried out in either of the following two ways:

- Serial Transmission: Represent each symbol by a unique waveform over a time interval of $T_s = T/N_c$, called *symbol time*, and transmit the waveforms sequentially.
- Parallel Transmission: Represent each symbol by a unique waveform over the entire time interval $T_s = T$ as the symbol time and sum the N_c waveforms representing the group of N_c symbols for parallel transmission. The individual waveforms must then be separated at the receiver to recover all N_c symbols.

Note that for either method the transmission rate is $R = N_c/T$ symbols per second.

Many of today's wireless communication systems operate in an environment where signals are reflected from a variety of objects such as buildings and walls on their way from transmitter to receiver. This means that the signal at the receiver is the sum of a number of copies of the transmitted signal with various delays and attenuations (referred to as multi-path). As the maximum delay increases from a small fraction of the symbol interval, inter-symbol interference (ISI) increases with a consequent increase in the probability of error at the receiver. Thus the parallel transmission of a group of symbols is advantageous due to the longer symbol time $T_s = T$.

We now consider the implementation of the parallel transmission by the *Orthogonal Frequency Division Multiplexing (OFDM)* method. Specifically, we use each of the N_c orthogonal sinusoids $e^{j2\pi kt/T} = \cos(2\pi kt/T) + j \sin(2\pi kt/T)$ of different frequencies $kf_0 = k/T$ ($k = 1, \dots, N_c$) to represent one of the N_c complex values d_k , so that the signal to be transmitted is a linear combination of these sinusoids weighted by d_k :

$$x(t) = \operatorname{Re} \left[\sum_{k=1}^{N_c} d_k e^{j2\pi kt/T} \right] = \sum_{k=1}^{N_c} [a_k \cos(2\pi kt/T) - b_k \sin(2\pi kt/T)] \quad (0 \leq t < T = 1/f_0) \quad (5.136)$$

This continuous signal is then be discretized by sampling at interval T/N or sampling rate $F_s = N/T$ to become:

$$x[n] = x(nT/N) = \sum_{k=1}^{N_c} [a_k \cos(2\pi kn/N) - b_k \sin(2\pi kn/N)], \quad (n = 0, \dots, N-1) \quad (5.137)$$

Note that if the sampling rate $F_s = N/T$ is higher than twice the maximum frequency component $N_c f_0 = N_c/T$ in the signal, i.e., $F_s = N/T > 2N_c f_0 = 2N_c/T$, i.e., $N > 2N_c$, then the Nyquist condition is satisfied and the continuous signal $x(t)$ can be reconstructed by a digital to analog (D/A) converter from the N samples $x[0], x[1], \dots, x[N-1]$.

The transmission of the N_c symbols can be carried out in the following steps:

1. Generate N samples $x[n]$ ($n = 0, \dots, N-1$) based on the N_c complex values d_k ($k = 1, \dots, N_c$) as in Eq.5.137;
2. Transmit $x[n]$ through the digital communication channel;
3. Reconstruct $x(t)$ from the received $x[n]$ by D/A conversion;
4. Separate $x(t)$ to recover the N_c symbols d_k .

Some analog circuits are necessary to generate the signal $x[n]$ in step 1 above. However, we now shown that such hardware requirement can be avoided as $x[n]$ can be completely generated by the following digital signal processing approach.

First we construct the following vector of $N = 2(N_c + 1)$ elements:

$$\begin{aligned} & [Y[0], Y[1], \dots, Y[N_c], Y[N_c + 1], Y[N_c + 2], \dots, Y[2N_c + 1]] \\ & = [0, d_1, \dots, d_{N_c}, 0, \bar{d}_{N_c}, \dots, \bar{d}_1] \end{aligned} \quad (5.138)$$

and then carry out the inverse DFT to get:

$$\begin{aligned} y[n] &= \mathcal{F}^{-1}[Y[k]] = \sum_{k=0}^{N-1} Y[k] e^{j2\pi nk/N} \\ &= \sum_{k=1}^{N_c} d_k e^{j2\pi kn/N} + \sum_{k=N_c+2}^{2N_c+1} \bar{d}_{2N_c+2-k} e^{j2\pi kn/N} \\ &\quad (n = 0, \dots, N-1 = 2N_c + 1) \end{aligned} \quad (5.139)$$

We let $m = N - k = 2N_c + 2 - k$ (i.e., $k = N - m = 2N_c + 2 - m$) so that the second summation becomes:

$$\sum_{m=N_c}^1 \bar{d}_m e^{-j2\pi mn/N} \underbrace{e^{jN(2\pi/N)n}}_1 = \sum_{m=1}^{N_c} \bar{d}_m e^{-j2\pi mn/N} \quad (5.140)$$

Now $y[n]$ above can be further written as:

$$\begin{aligned} y[n] &= \sum_{k=1}^{N_c} [d_k e^{j2\pi kn/N} + \bar{d}_k e^{-j2\pi kn/N}] \\ &= \sum_{k=1}^{N_c} [(a_k + jb_k) e^{j2\pi kn/N} + (a_k - jb_k) e^{-j2\pi kn/N}] \\ &= 2 \sum_{k=1}^{N_c} \left[a_k \left(\frac{e^{j2\pi kn/N} + e^{-j2\pi kn/N}}{2} \right) - b_k \left(\frac{e^{j2\pi kn/N} - e^{-j2\pi kn/N}}{2j} \right) \right] \\ &= 2 \sum_{k=1}^{N_c} [a_k \cos(2\pi kn/N) - b_k \sin(2\pi kn/N)] = 2x[n], \quad (n = 0, \dots, N-1) \end{aligned} \quad (5.141)$$

which happens to be the signal we need to generate in step 1 above. After this signal is transmitted and then received, we can carry out the DFT to get:

$$Y[k] = \frac{1}{N} \sum_{n=0}^{N-1} y[n] e^{-j2\pi kn/N}, \quad (k = 0, \dots, N-1) \quad (5.142)$$

The N_c original symbols d_k carried in the signal $x[n]$ can then be easily recovered as $d_k = Y[k]$ ($k = 1, \dots, N_c$) according to Eq. 5.138.

5.9 Homework Problems

Some of the problems below can be carried out in Matlab (or any other programming language of choice).

1. Assume a real LTI system $h(t) = \bar{h}(t)$, re-derive Eqs.5.5 and 5.6 by applying Eq.5.4 to the following:

$$\mathcal{O}[\cos(2\pi ft)] = \frac{1}{2}\mathcal{O}[e^{j2\pi ft}] + \mathcal{O}[e^{-j2\pi ft}] \quad (5.143)$$

and

$$\mathcal{O}[\sin(2\pi ft)] = \frac{1}{2j}\mathcal{O}[e^{j2\pi ft}] - \mathcal{O}[e^{-j2\pi ft}] \quad (5.144)$$

2. Sketch the response of the system in Example 5.1 when the input is a square wave with period T :

$$v_{in}(t) = \begin{cases} 1 & 0 < t < T/2 \\ -1 & T/2 < t < T \end{cases} \quad (5.145)$$

Consider the different cases when $T \approx \tau$ and $T \gg \tau$.

3. Consider the same RC circuit in Example 5.1 (Fig.5.2), with an input voltage $x(t) = v_{in}(t)$ across the two components in series, but the output $y(t) = V_R(t)$ is the voltage across resistor R .

The impulse response of the system can be most easily obtained based on the result of Example 5.1 and the Kirchhoff's voltage law: $v_{in}(t) = v_C(t) + v_R(t)$:

$$v_R(t) = v_{in}(t) - v_C(t) = \delta(t) - \frac{1}{\tau}e^{-t/\tau}u(t) \quad (5.146)$$

However, let us solve this system independently without using the previous result by following the following steps:

- Set up the differential equation of the system
 - Find the impulse response function $h(t)$ in two methods when $x(t) = \delta(t)$:
 - (a) $v_R(t) = v_{in}(t) - v_C(t)$. When $v_{in}(t) = \delta$, $v_R(t) = h(t)$ and $v_C(t)$ is obtained in Example 5.1.
 - (b) Solve the differential equation for $y'(t) = f(t)u(t)$ when $x'(t) = u(t)$. Then find $h(t) = y(t) = y'(t)$ corresponding to $x(t) = \dot{x}(t) = \delta(t)$.
 - Find the frequency response function $H(\omega)$ by assuming $x(t) = e^{j\omega t}$.
 - Verify that $H(\omega) = \mathcal{F}[h(t)]$.
 - Sketch the response of the system when the input is a square wave in the previous problem.
4. Implement various filtering schemes to filter an input sound signal.
 - a. Load a sound file such as a piece of music. For example, “load Handel” in Matlab will load the first 9 seconds of Hallelujah Chorus by Handel into a vector y with sampling rate in variable F_s .
 - b. Implement a set of five band-pass filters, both ideal and Butterworth, that divide the entire frequency range into five frequency bands, so that the first

filter (low-pass) passes only low frequency components including DC, the second (band-pass) passes the frequency components in the next higher frequency band, etc., until and the last (high-pass) passes all high frequency components in the last band including the highest frequency component contained in the signal. Note that your filters need to cover negative frequencies as well as positive ones.

- c. Listen to the output from each of these filters to experience the different filtering effects. Compare the filtering effects of the ideal and Butterworth filters of different orders. (In Matlab, to listen to a signal in vector y , do “play(audiophyer(y,Fs))”.)
- d. Repeat the above with band-pass filters replaced by band-stop filters.
- e. If the gains of the band-pass filters can be individually adjusted, they are called *equalization (EQ) filters* and can be used to compensate for the unequal (uneven) frequency response of the signal processing system to reduce the signal distortion caused by the system and improve the sound quality of the signal.
5. Find and sketch the Bode plots, including both the log-magnitude (Lm) plot of $LmH(\omega)$ and phase plot $\angle H(\omega)$ versus $\log_{10}\omega$, of the following frequency response functions of some typical LTI systems.
 - a. Constant gain $H(\omega) = k$ (consider both cases $k > 0$ and $k < 0$)
 - b. Derivative factor $H(\omega) = j\omega\tau$. What is the slope of the Lm plot in terms of dB/dec (decibel per 10-fold frequency increase).
 - c. Integral factor $H(\omega) = 1/j\omega\tau$. What is the slope of the Lm plot?
 - d. First order factor in numerator $H(\omega) = 1 + j\omega\tau$. First give the general expressions of $LmH(\omega)$ and $\angle H(\omega)$. Then consider the following three special cases:
 - * $\omega\tau = 1$, i.e., $\omega = 1/\tau$
 - * $\omega\tau \ll 1$, find the asymptote of both Lm and phase plots
 - * $\omega\tau \gg 1$, find the asymptote of both Lm and phase plots. what is the slope of the Lm plot?
 Sketch the complete plots by combining the three cases.
 - e. First order factor in denominator $H(\omega) = 1/(1 + j\omega\tau)$
6. Construct an analytic signal based on (a) $x(t) = \cos(\omega_0 t)$ and (b) $y(t) = \sin(\omega_0 t)$. Verify that the negative half of the spectrum of the constructed analytic signal is zero.
7. Implement AM modulation and demodulation as discussed in Example 5.4
 - a. Create a triangular spectrum as shown in the top panel of Fig.5.10 with the highest frequency f_m and obtain the time signal by inverse Fourier transform.
 - b. Carry out AM modulation (Eq.5.78) of the signal with a carrier frequency $f_c > 2f_m$, display the spectrum of the resulting signal in both time and frequency domains.

- c. Carry out AM demodulation (Eq.5.80, display the spectrum of the resulting signal in both time and frequency domains.
 - d. Carry out an ideal low-pass filtering to remove all frequencies higher than f_m . display the spectrum of the resulting signal in both time and frequency domains.
 - e. Replace the artificial signal above by a real sound signal, and repeat the steps above. You may need to low-pass filter the signal to make sure $f_m < f_c/2$. Listen to the original signal and the reconstructed signal to convince yourself it is perfectly reconstructed.
8. Implement single sideband (SSB) modulation discussed in Example 5.10 by following the diagram shown in Fig.5.22.
 - a. Use the same artificial signal $x(t)$ with a triangular spectrum and obtain its Hilbert transform $\hat{x}(t)$. (in Matlab, the analytic version of a given signal vector x can be obtained by function `hilbert(x)`, whose imaginary part `imag(hilbert(x))` is the Hilbert transform of x .) Display both $x(t)$ and $\hat{x}(t)$ and their spectra.
 - b. Use $x(t)$ and $\hat{x}(t)$ to amplitude modulate respectively $\cos(2\pi f_c t)$ and $\sin(2\pi f_c t)$ with $f_c > 2f_m$. Display the resulting signals and their spectra.
 - c. Find the difference as given in Eq.5.106, display the signal and its spectrum to verify that it has only one sideband.
 - d. Repeat the steps above with the artificial signal replaced by a real sound signal. You may need to low-pass filter the signal first to make sure $f_m < f_c/2$.
 9. Implement image filtering and compression as discussed in Examples 5.7 and 5.8.
 - a. Carry out 2-D DFT of an image of your choice and display its spectrum first in terms of the real X_r and imaginary X_j parts, and then its magnitude and phase. The spectral information can be displayed as 3-D plots as well as 2-D images. Note that a nonlinear mapping (such as $y = x^{0.3}$) may be needed in order to see most of the frequency components.
 - b. Carry out ideal low-pass and high-pass filtering of the image and display the filters as well as the image after filtering in both spatial and spatial frequency domains.
 - c. Repeat the step above with the ideal filters replaced by the corresponding Butterworth filters.
 - d. Carry out directional filtering as shown in Fig.5.20.
 - e. Carry out image compression as shown in Fig.5.21 by suppressing to zero all frequency components lower than a certain threshold. Obtain the percentage of such suppressed frequency components, and the percentage of lost energy (in terms of signal value squared). (Note that this exercise only serves to illustrate the basic idea of image compression but it is not how image compression is practically done, where those components suppressed need to be recorded as well.)

10. The m-file QPSK_OFDMTxRx.m simulates a baseband OFDM system for transmitting an ASCII file such as a text message using Quadrature Phase Shift Keying (QPSK) to represent the bit stream.
 - a. Create an ASCII text file and using QPSK_OFDMTxRx.m experiment with different FFT and cyclic prefix lengths. Also try slightly mismatching the channel impulse responses at the transmitter and receiver.
 - b. Write your own Matlab function $x=d2x(d)$ that uses the IDFT to produce the sequence \bar{x} given the sequence \bar{d} .
 - c. Write your own Matlab function $d=x2d(x)$ that uses the DFT to produce the sequence \bar{d} given the sequence \bar{x} .
 - d. Test your functions using the sequence $\bar{d} = (1, 2, 3, 4, 5, 6, 7, 8)$.

6 The Laplace and z-Transforms

The Laplace and z-transforms are respectively the natural generalization of the continuous and discrete-time Fourier transforms, and both find a wide variety of applications in many fields of science and engineering in general, and in signal processing and system analysis/design in particular. Due to some of its most favorable properties, such as the conversion of ordinary differential and difference equations into easily solvable algebraic equations, a problem presented in time domain can be much more conveniently tackled in the transform domain.

6.1 The Laplace Transform

6.1.1 From Fourier Transform to Laplace Transform

The Laplace transform of a signal $x(t)$ can be considered as the generalization of the continuous-time Fourier transform (CTFT) of the signal:

$$\mathcal{F}[x(t)] = \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt = X(j\omega) \quad (6.1)$$

Here we adopt the notation $X(j\omega)$ for the CTFT spectrum, instead of $X(f)$ or $X(\omega)$ used previously, for some reason which will become clear later. The transform above is based on the underlying assumption that the signal $x(t)$ is square integrable so that the integral converges and the spectrum $X(j\omega)$ exists. However, this assumption is not valid for signals such as $x(t) = t$, $x(t) = x^2$, and $x(t) = e^{at}$, all of which are not square integrable as they grow without a bound when $|t| \rightarrow \infty$. In such cases, we could still consider the Fourier transform of a modified version of the signal $x'(t) = x(t)e^{-\sigma t}$, where $e^{-\sigma t}$ is an exponential factor with a real parameter σ , which can force the given signal $x(t)$ to decay exponentially for properly chosen value of σ (either positive or negative). For example, $x(t) = e^{at}u(t)$ ($a > 0$) does not converge when $t \rightarrow \infty$, therefore its Fourier spectrum does not exist. However, if we choose $\sigma > a$, the modified version $x'(t) = x(t)e^{-\sigma t} = e^{-(\sigma-a)t}u(t)$ will converge as $t \rightarrow \infty$.

In general, the Fourier transform of the modified signal is:

$$\mathcal{F}[x'(t)] = \mathcal{F}[x(t)e^{-\sigma t}] = \int_{-\infty}^{\infty} x(t)e^{-(\sigma+j\omega)t} dt = \int_{-\infty}^{\infty} x(t)e^{-st} dt \quad (6.2)$$

where we have defined a complex variable $s = \sigma + j\omega$. If the integral above converges, it results in a complex function $X(s)$, called the *bilateral Laplace transform* of $x(t)$, formally defined as:

$$X(s) = \mathcal{L}[x(t)] = \mathcal{F}[x(t)e^{-\sigma t}] = \int_{-\infty}^{\infty} x(t)\phi(t, s)dt = \int_{-\infty}^{\infty} x(t)e^{-st}dt \quad (6.3)$$

Same as the continuous-time Fourier transform, the Laplace transform can also be considered as an integral transform with a kernel function:

$$\phi(t, s) = e^{-st} = e^{-(\sigma+j\omega)t} = e^{-\sigma t}e^{-j\omega t} \quad (6.4)$$

which is a modified version of the kernel function $\phi(t, f) = e^{j2\pi ft}$ for the Fourier transform. However, different from the parameter f for frequency in the Fourier kernel function, the parameter $s = \sigma + j\omega$ in the Laplace kernel is complex with both real and imaginary parts $Re[s] = \sigma$ and $Im[s] = \omega$, and the transform $X(s)$, a complex function, is defined in a 2-D complex plane, called s-plane, with a Cartesian coordinates of σ for the real (horizontal) axis and $j\omega$ for the imaginary (vertical) axis.

The Laplace transform $X(s)$ exists only inside a certain region of the s-plane, called the *region of convergence (ROC)*, composed of all s values that guarantee the convergence of the integral in Eq. 6.3. Due to the introduction of the exponential decay factor $e^{-\sigma t}$, we can properly choose the parameter σ so that the Laplace transform can be applied to a broader class of signals than the Fourier transform.

If the imaginary axis $s = j\omega$ (corresponding to $Re[s] = \sigma = 0$) is inside the ROC, then we can evaluate the 2-D function $X(s)$ along the imaginary axis from $\omega = -\infty$ to $\omega = \infty$ to obtain the Fourier transform $X(j\omega)$ of $x(t)$. In other words, the 1-D Fourier spectrum of the signal is the cross section of the 2-D function $X(s) = X(\sigma + j\omega)$ along the imaginary axis $s = j\omega$, if it is inside the ROC, i.e., the CTFT is just a special case of the Laplace transform when $\sigma = 0$ and $s = j\omega$:

$$\mathcal{F}[x(t)] = \mathcal{L}[x(t)]|_{s=j\omega} = X(s)|_{s=j\omega} = X(j\omega) \quad (6.5)$$

This is the reason why the Fourier spectrum can also be denoted by $X(j\omega)$.

Given the Laplace transform $X(s) = \mathcal{L}[x(t)]$, the time signal $x(t)$ can be obtained by the inverse Laplace transform, which can be derived from the corresponding Fourier transform:

$$\mathcal{L}[x(t)] = X(s) = X(\sigma + j\omega) = \mathcal{F}[x(t)e^{-\sigma t}] \quad (6.6)$$

Taking the inverse Fourier transform of the above, we get

$$x(t)e^{-\sigma t} = \mathcal{F}^{-1}[X(\sigma + j\omega)] = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\sigma + j\omega)e^{j\omega t}d\omega \quad (6.7)$$

Multiplying both sides by $e^{\sigma t}$, we get:

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\sigma + j\omega) e^{(\sigma+j\omega)t} d\omega \quad (6.8)$$

To further represent this inverse transform in terms of s (instead of ω), we note

$$ds = d(\sigma + j\omega) = j d\omega, \quad i.e., \quad d\omega = ds/j \quad (6.9)$$

The integral over $-\infty < \omega < \infty$ with respect to ω corresponds to the integral with respect to s over $\sigma - j\infty < s < \sigma + j\infty$:

$$x(t) = \mathcal{L}^{-1}[X(s)] = \frac{1}{j2\pi} \int_{\sigma-j\infty}^{\sigma+j\infty} X(s) e^{st} ds \quad (6.10)$$

Now we get the forward and inverse Laplace transform pair:

$$\begin{aligned} X(s) &= \mathcal{L}[x(t)] = \int_{-\infty}^{\infty} x(t) e^{-st} dt \\ x(t) &= \mathcal{L}^{-1}[X(s)] = \frac{1}{j2\pi} \int_{\sigma-j\infty}^{\sigma+j\infty} X(s) e^{st} ds \end{aligned} \quad (6.11)$$

which can also be more concisely represented as:

$$x(t) \xleftrightarrow{\mathcal{L}} X(s) \quad (6.12)$$

In practice, we hardly need to carry out the integral in the inverse transform with respect to the complex variable s , as the Laplace transform pairs of most signals of interest can be obtained in some other ways and made available in table form.

In many applications the Laplace transform is a rational function as a ratio of two polynomials:

$$X(s) = \frac{N(s)}{D(s)} = \frac{\sum_{k=0}^M b_k s^k}{\sum_{k=0}^N a_k s^k} = \frac{b_M}{a_N} \frac{\prod_{k=1}^M (s - z_k)}{\prod_{k=1}^N (s - p_k)} \quad (6.13)$$

The last equal sign in Eq.6.13 is due to the fundamental theorem of algebra, stating that an N th order polynomial has N roots (some of which may be repeated with multiplicity greater than 1). Here the roots z_k , ($k = 1, 2, \dots, m$) of the numerator polynomial of order M are called the *zeros* of $X(s)$, and the roots p_k , ($k = 1, 2, \dots, n$) of the denominator polynomial of order N are called the *poles* of $X(s)$, i.e.,

$$X(z_k) = 0, \quad \text{and} \quad X(p_k) = \infty \quad (6.14)$$

The locations of the zeros and poles of $X(s)$ in the s -plane is of great importance as they characterize some most essential properties of a signal $x(t)$, such as whether it is right or left-sided, whether it grows or decays over time, as to be discussed later.

Moreover, if $N > M$, then $X(\infty) = 0$, i.e., $s = \infty$ is a zero. On the other hand, if $M > N$, then $X(\infty) = \infty$, i.e., $s = \infty$ is a pole. In general, we always assume

$M < N$, as otherwise we can carry out a long division to expand $X(s)$ into multiple terms so that $M < N$ is true for each fraction. For example,

$$X(s) = \frac{s^2 - 3s + 1}{s - 2} = s - 1 - \frac{1}{s - 2} \quad (6.15)$$

Now Eq. 6.13 can be converted into a sum of N terms by the method of partial fraction expansion:

$$X(s) = \frac{b_M}{a_N} \frac{\prod_{k=1}^M (s - z_k)}{\prod_{k=1}^N (s - p_k)} = \frac{b_M}{a_N} \sum_{k=1}^N \frac{c_k}{s - p_k} \quad (6.16)$$

6.1.2 The Region of Convergence

A Laplace transform $X(s) = \mathcal{L}[x(t)]$ always needs to be associated with the corresponding ROC, without which the inverse transform $x(t) = \mathcal{L}^{-1}[X(s)]$ cannot be meaningfully carried out. This point can be best illustrated in the following example.

Example 6.1:

1. A right-sided signal $x(t) = e^{-at}u(t)$ (a is a real constant):

$$X(s) = \int_0^\infty e^{-at} e^{-st} dt = \int_0^\infty e^{-at} e^{-(\sigma+j\omega)t} dt = \int_0^\infty e^{-(a+\sigma)t} e^{-j\omega t} dt \quad (6.17)$$

For this integral to converge, it is necessary to have $a + \sigma > 0$, i.e. the ROC is $\text{Re}[s] = \sigma > -a$, inside which the above becomes:

$$X(s) = \frac{1}{-(a + \sigma + j\omega)} e^{-(a+\sigma+j\omega)t} \Big|_0^\infty = \frac{1}{(\sigma + a) + j\omega} = \frac{1}{s + a} \quad (6.18)$$

In particular, if $a = 0$, $x(t) = u(t)$ and we have

$$U(s) = \mathcal{L}[u(t)] = \frac{1}{s}, \quad \sigma > 0 \quad (6.19)$$

If we let $\sigma \rightarrow 0$, then $U(s)$ is evaluated along the imaginary axis $s = j\omega$ and becomes $U(j\omega) = 1/j\omega$, which is seemingly the Fourier transform of $u(t)$. However this result is actually invalid, as $\sigma = 0$ is not inside the ROC $R[s] = \sigma > 0$. Comparing this result with the real Fourier transform of $u(t)$ in Eq.3.72:

$$\mathcal{F}[u(t)] = \frac{1}{2}\delta(f) + \frac{1}{j\omega} \quad (6.20)$$

we see that an extra term $\delta(f)/2$ in the Fourier spectrum which reflects the fact that the integral is only marginally convergent when $s = j\omega$.

2. A left-sided signal $x(t) = -e^{-at}u(-t)$:

$$X(s) = - \int_{-\infty}^0 e^{-at} e^{-st} dt = - \int_{-\infty}^0 e^{-(a+\sigma+j\omega)t} dt \quad (6.21)$$

where a is a real constant. For this integral to converge, it is necessary that $a + \sigma < 0$, i.e., the ROC is $\text{Re}[s] = \sigma < -a$, inside which the above becomes:

$$X(s) = \frac{1}{a + \sigma + j\omega} e^{-(a+\sigma+j\omega)t} \Big|_{-\infty}^0 = \frac{1}{a + \sigma + j\omega} = \frac{1}{s + a}, \quad \sigma < -a \quad (6.22)$$

When $a = 0$, $x(t) = -u(-t)$ we have

$$\mathcal{L}[-u(-t)] = \frac{1}{s}, \quad \sigma < 0 \quad (6.23)$$

We see that the Laplace transforms of two different signals $e^{-at}u(t)$ and $-e^{-at}u(-t)$ are identical, but their corresponding ROCs are different.

Based on the examples above we summarize a set of properties of the ROC:

- If a signal $x(t)$ of finite duration is absolutely integrable then its transform $X(s)$ exists for any s , i.e., its ROC is the entire s-plane.
- The ROC does not contain any poles at which $X(s) = \infty$.
- Two different signals may have identical transform but different ROCs. The inverse transform can be carried out only if an associated ROC is also specified.
- Only the real part $\text{Re}[s] = \sigma$ of s determines the convergence of the integral in the Laplace transform and thereby the ROC. The imaginary part $\text{Im}[s]$ has no effect on the convergence. Consequently the ROC is always bounded by two vertical lines parallel to the imaginary axis $s = j\omega$, corresponding to two poles p_1 and p_2 with $\text{Re}[p_1] < \text{Re}[p_2]$. It is possible that $\text{Re}[p_1] = -\infty$ and/or $\text{Re}[p_2] = \infty$.
- The ROC of a right-sided signal is the right-sided half plane to the right of the rightmost pole; The ROC of the transform of a left-sided signal is a left-sided half plane to the left of the leftmost pole. If a signal is two-sided, its ROC is the intersection of the two ROCs corresponding to its two one-sided parts, which can be either a vertical strip or an empty set.
- The Fourier transform $X(j\omega)$ of a signal $x(t)$ exists if the ROC of the corresponding Laplace transform $X(s)$ contains the imaginary axis $\text{Re}[s] = 0$, i.e., $s = j\omega$.

6.1.3 Properties of the Laplace Transform

The Laplace transform has a set of properties most of which are in parallel with those of the Fourier transform. The proofs of most of these properties are omitted as they are similar to that of their counterparts in the Fourier transform. However, here we need to pay special attention to the ROCs. Here we always assume:

$$\mathcal{L}[x(t)] = X(s), \quad \mathcal{L}[y(t)] = Y(s) \quad (6.24)$$

with ROCs R_x and R_y , respectively.

- **Linearity**

$$\mathcal{L}[ax(t) + by(t)] = aX(s) + bY(s), \quad ROC \supseteq (R_x \cap R_y) \quad (6.25)$$

It is obvious that the ROC of the linear combination of $x(t)$ and $y(t)$ should be the intersection $R_x \cap R_y$ of their individual ROCs in which both $X(s)$ and $Y(s)$ exist. However, note that in some cases the ROC of the linear combination may be larger than $R_x \cap R_y$. For example, $\mathcal{L}[u(t)] = 1/s$ and $\mathcal{L}[u(t - \tau)] = e^{-s\tau}/s$ have the same ROC $Re[s] > 0$, but their difference $u(t) - u(t - \tau)$ has finite duration and the corresponding ROC is the entire s-plane. Also when *zero-pole cancellation* occurs the ROC of the linear combination may also be larger than $R_x \cap R_y$. For example, let

$$X(s) = \mathcal{L}[x(t)] = \frac{1}{s+1}, \quad Re[s] > -1 \quad (6.26)$$

and

$$Y(s) = \mathcal{L}[y(t)] = \frac{1}{(s+1)(s+2)}, \quad Re[s] > -1 \quad (6.27)$$

then

$$\begin{aligned} \mathcal{L}[x(t) - y(t)] &= \frac{1}{s+1} - \frac{1}{(s+1)(s+2)} = \frac{s+1}{(s+1)(s+2)} = \frac{1}{s+2} \\ Re[s] &> -2 \end{aligned} \quad (6.28)$$

- **Time-shift**

$$\mathcal{L}[x(t - t_0)] = e^{-t_0 s} X(s), \quad ROC = R_x \quad (6.29)$$

- **Time reversal**

$$\mathcal{L}[x(-t)] = X(-s), \quad ROC = -R_x \quad (6.30)$$

- **s-Domain shift**

$$\mathcal{L}[e^{-s_0 t} x(t)] = X(s + s_0), \quad ROC = R_x + Re[s_0] \quad (6.31)$$

Note that the ROC is shifted by s_0 , i.e., it is shifted vertically by $Im[s_0]$ (with no effect on ROC) and horizontally by $Re[s_0]$.

- **Time scaling**

$$\mathcal{L}[x(at)] = \frac{1}{|a|} X\left(\frac{s}{a}\right), \quad ROC = \frac{R_x}{a} \quad (6.32)$$

Note that the ROC is horizontally scaled by $1/a$, which could be either positive ($a > 0$) or negative ($a < 0$) in which case both the function $x(t)$ and the ROC of its Laplace transform are horizontally flipped.

- **Conjugation**

$$\mathcal{L}[x^*(t)] = X^*(s^*), \quad ROC = R_x \quad (6.33)$$

- **Convolution**

$$\mathcal{L}[x(t) * y(t)] = X(s)Y(s), \quad ROC \supseteq (R_x \cap R_y) \quad (6.34)$$

Note that the ROC of the convolution could be larger than the intersection of R_x and R_y , due to the possible pole-zero cancellation caused by the convolution, similar to the linearity property. For example, assume

$$X(s) = \mathcal{L}[x(t)] = \frac{s+1}{s+2}, \quad Re[s] > -2 \quad (6.35)$$

$$Y(s) = \mathcal{L}[y(t)] = \frac{s+2}{s+1}, \quad Re[s] > -1 \quad (6.36)$$

then

$$\mathcal{L}[x(t) * y(t)] = X(s)Y(s) = 1 \quad (6.37)$$

with an ROC of the entire s-plane.

- **Differentiation in time domain**

$$\mathcal{L}\left[\frac{d}{dt}x(t)\right] = sX(s), \quad ROC \supseteq R_x \quad (6.38)$$

This is an important property based on which the Laplace transform finds a lot of applications in system analysis and design. This property can be proven by differentiating the inverse Laplace transform:

$$\frac{d}{dt}x(t) = \frac{1}{j2\pi} \int_{\sigma-j\infty}^{\sigma+j\infty} X(s) \frac{d}{dt}e^{st} ds = \frac{1}{j2\pi} \int_{\sigma-j\infty}^{\sigma+j\infty} sX(s)e^{st} ds \quad (6.39)$$

Again, multiplying $X(s)$ by s may cause pole-zero cancellation and therefore the resulting ROC may be larger than R_x . For example, let $x(t) = u(t)$ and $X(s) = \mathcal{L}[u(t)] = 1/s$ with ROC $Re[s] > 0$, then we have $\mathcal{L}[dx(t)/dt] = \mathcal{L}[\delta(t)] = sX(s) = 1$, but its ROC is the entire s-plane. Repeating this property we get:

$$\mathcal{L}\left[\frac{d^n}{dt^n}x(t)\right] = s^n X(s) \quad (6.40)$$

In particular, when $x(t) = \delta(t)$, we have

$$\mathcal{L}\left[\frac{d^n}{dt^n}\delta(t)\right] = s^n, \quad ROC = \text{entire s-plane} \quad (6.41)$$

- **Differentiation in s-Domain**

$$\mathcal{L}[tx(t)] = -\frac{d}{ds}X(s), \quad ROC = R_x \quad (6.42)$$

This can be proven by differentiating the Laplace transform:

$$\frac{d}{ds}X(s) = \int_{-\infty}^{\infty} x(t) \frac{d}{ds}e^{-st} dt = \int_{-\infty}^{\infty} (-t)x(t)e^{-st} dt \quad (6.43)$$

Repeat this process we get

$$\mathcal{L}[t^n x(t)] = (-1)^n \frac{d^n}{ds^n} X(s), \quad ROC = R_x \quad (6.44)$$

- **Integration in time domain**

$$\mathcal{L}\left[\int_{-\infty}^t x(\tau) d\tau\right] = \frac{X(s)}{s}, \quad ROC \supseteq (R_x \cap \{Re[s] > 0\}) \quad (6.45)$$

This can be proven by realizing that

$$x(t) * u(t) = \int_{-\infty}^{\infty} x(\tau) u(t - \tau) d\tau = \int_{-\infty}^t x(\tau) d\tau \quad (6.46)$$

and therefore by convolution property we have

$$\mathcal{L}[x(t) * u(t)] = X(s) \frac{1}{s} \quad (6.47)$$

As the ROC of $\mathcal{L}[u(t)] = 1/s$ is the right half plane $Re[s] > 0$, the ROC of $X(s)/s$ is the intersection $R_x \cap \{Re[s] > 0\}$, except when pole-zero cancellation occurs. For example, when $x(t) = d\delta(t)/dt$ with $X(s) = s$, $\mathcal{L}[\int_{-\infty}^t x(\tau) d\tau] = s/s = 1$ with the ROC being the entire s-plane.

6.1.4 Laplace Transform of Typical Signals

- $\delta(t), \delta(t - \tau)$

$$\mathcal{L}[\delta(t)] = \int_{-\infty}^{\infty} \delta(t) e^{-st} dt = e^0 = 1, \quad \text{ROC: entire } s \text{ plane} \quad (6.48)$$

Moreover, due to time-shift property, we have

$$\mathcal{L}[\delta(t - \tau)] = e^{-s\tau}, \quad \text{ROC: entire } s \text{ plane} \quad (6.49)$$

As the Laplace integration converges for any s , the ROC is the entire s-plane.

- $u(t), t u(t), t^n u(t)$

Due to the property of time domain integration, we have

$$\mathcal{L}[u(t)] = \mathcal{L}\left[\int_{-\infty}^t \delta(\tau) d\tau\right] = \frac{1}{s}, \quad Re[s] > 0 \quad (6.50)$$

Applying the s-domain differentiation property to the above, we have

$$\mathcal{L}[tu(t)] = -\frac{d}{ds}\left[\frac{1}{s}\right] = \frac{1}{s^2}, \quad Re[s] > 0 \quad (6.51)$$

and in general

$$\mathcal{L}[t^n u(t)] = \frac{n!}{s^{n+1}}, \quad Re[s] > 0 \quad (6.52)$$

- $e^{-at} u(t), te^{-at} u(t)$

Applying the s-domain shifting property to

$$\mathcal{L}[u(t)] = \frac{1}{s}, \quad \text{Re}[s] > 0 \quad (6.53)$$

we have

$$\mathcal{L}[e^{-at}u(t)] = \frac{1}{s+a}, \quad \text{Re}[s] > -a \quad (6.54)$$

Applying the same property to

$$\mathcal{L}[t^n u(t)] = \frac{n!}{s^{n+1}}, \quad \text{Re}[s] > 0 \quad (6.55)$$

we have

$$\mathcal{L}[t^n e^{-at}u(t)] = \frac{n!}{(s+a)^{n+1}}, \quad \text{Re}[s] > -a \quad (6.56)$$

- $e^{-j\omega_0 t}u(t), \sin(\omega_0 t)u(t), \cos(\omega_0 t)u(t)$

Letting $a = \pm j\omega_0$ in

$$\mathcal{L}[e^{-at}u(t)] = \frac{1}{s+a}, \quad \text{Re}[s] > -\text{Re}[a] \quad (6.57)$$

we get

$$\mathcal{L}[e^{-j\omega_0 t}u(t)] = \frac{1}{s+j\omega_0} \quad \text{and} \quad \mathcal{L}[e^{j\omega_0 t}u(t)] = \frac{1}{s-j\omega_0} \quad \text{Re}[s] > 0 \quad (6.58)$$

and therefore

$$\mathcal{L}[\cos(\omega_0 t)u(t)] = \frac{1}{2}\mathcal{L}[e^{j\omega_0 t} + e^{-j\omega_0 t}] = \frac{1}{2}\left[\frac{1}{s-j\omega_0} + \frac{1}{s+j\omega_0}\right] = \frac{s}{s^2 + \omega_0^2} \quad (6.59)$$

and

$$\mathcal{L}[\sin(\omega_0 t)u(t)] = \frac{1}{2j}\mathcal{L}[e^{j\omega_0 t} - e^{-j\omega_0 t}] = \frac{1}{2j}\left[\frac{1}{s-j\omega_0} - \frac{1}{s+j\omega_0}\right] = \frac{\omega_0}{s^2 + \omega_0^2} \quad (6.60)$$

- $t \cos(\omega_0 t)u(t), t \sin(\omega_0 t)u(t)$

Letting $a = \pm j\omega_0$ in

$$\mathcal{L}[te^{-at}u(t)] = \frac{1}{(s+a)^2}, \quad \text{Re}[s] > -a \quad (6.61)$$

we get

$$\mathcal{L}[te^{-j\omega_0 t}u(t)] = \frac{1}{(s+j\omega_0)^2}, \quad \mathcal{L}[te^{j\omega_0 t}u(t)] = \frac{1}{(s-j\omega_0)^2}, \quad \text{Re}[s] > -a \quad (6.62)$$

Based on these we have:

$$\begin{aligned} \mathcal{L}[t \cos(\omega_0 t)u(t)] &= \frac{1}{2}\mathcal{L}[t(e^{j\omega_0 t} + e^{-j\omega_0 t})] = \frac{1}{2}\left[\frac{1}{(s-j\omega_0)^2} + \frac{1}{(s+j\omega_0)^2}\right] \\ &= \frac{s^2 - \omega_0^2}{(s^2 + \omega_0^2)^2} \end{aligned} \quad (6.63)$$

and

$$\begin{aligned}\mathcal{L}[t \sin(\omega_0 t)u(t)] &= \frac{1}{2j} \mathcal{L}[t (e^{j\omega_0 t} - e^{-j\omega_0 t})] = \frac{1}{2j} \left[\frac{1}{(s - j\omega_0)^2} - \frac{1}{(s + j\omega_0)^2} \right] \\ &= \frac{2s\omega_0}{(s^2 + \omega_0^2)^2}\end{aligned}\quad (6.64)$$

- $e^{-at} \cos(\omega_0 t)u(t), e^{-at} \sin(\omega_0 t)u(t)$

Applying s-domain shifting property to

$$\mathcal{L}[\cos(\omega_0 t)u(t)] = \frac{s}{s^2 + \omega_0^2}, \quad \text{and} \quad \mathcal{L}[\sin(\omega_0 t)u(t)] = \frac{\omega_0}{s^2 + \omega_0^2} \quad (6.65)$$

we get, respectively

$$\mathcal{L}[e^{-at} \cos(\omega_0 t)u(t)] = \frac{s+a}{(s+a)^2 + \omega_0^2} \quad (6.66)$$

and

$$\mathcal{L}[e^{-at} \sin(\omega_0 t)u(t)] = \frac{\omega_0}{(s+a)^2 + \omega_0^2} \quad (6.67)$$

6.1.5 Analysis of Continuous LTI Systems by Laplace Transform

The Laplace transform is a convenient tool for the analysis and design of continuous LTI systems $y(t) = \mathcal{O}[x(t)]$ whose output $y(t)$ is the convolution of the input $x(t)$ and its impulse response function $h(t)$:

$$y(t) = \mathcal{O}[x(t)] = h(t) * x(t) = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau \quad (6.68)$$

In particular, if the input is an impulse $x(t) = \delta(t)$, then the output is the impulse response function $y(t) = \mathcal{O}[\delta(t)] = h(t) * \delta(t) = h(t)$. Also if the input is a complex exponential $x(t) = e^{st} = e^{\sigma+j\omega}$, then the output can be found to be

$$y(t) = \mathcal{O}[e^{st}] = \int_{-\infty}^{\infty} h(\tau)e^{s(t-\tau)}d\tau = e^{st} \int_{-\infty}^{\infty} h(\tau)e^{-s\tau}d\tau = H(s)e^{st} \quad (6.69)$$

where $H(s)$ is the *transfer function* of the system, first defined in Eq.1.89 in Chapter 1, which is actually the Laplace transform of the impulse response $h(t)$ of the system:

$$H(s) = \mathcal{L}[h(t)] = \int_{-\infty}^{\infty} h(t)e^{-st}dt \quad (6.70)$$

Eq.6.69 is the eigenequation of *any* continuous LTI system, where the transfer function $H(s)$ is the eigenvalue, and the complex exponential input $x(t) = e^{st}$ is the corresponding eigenfunction. In particular, if we let $\sigma = 0$, then $s = j\omega$ and the transfer function $H(s)$ becomes the Fourier transform of the impulse response $h(t)$ of the system:

$$H(s)|_{s=j\omega} = H(j\omega) = \int_{-\infty}^{\infty} h(t)e^{-j\omega t}dt = \mathcal{F}[h(t)] \quad (6.71)$$

This is the frequency response function of the LTI system first defined in Eq.5.3 of Chapter 3. Various properties and behaviors such as the stability and filtering effects of a continuous LTI system can be qualitatively characterized based on the locations of the zeros and poles of its transfer function $H(s) = \mathcal{L}[h(t)]$ due to the properties of the ROC of the Laplace transform.

Also, due to its convolution property of the Laplace transform, the convolution in Eq.6.68 can be converted to a multiplication in s-domain:

$$y(t) = h(t) * x(t) \xrightarrow{\mathcal{L}} Y(s) = H(s)X(s) \quad (6.72)$$

Based on this relationship the transfer function $H(s)$ can also be found in s-domain as the ratio $H(s) = Y(s)/X(s)$ of the output $Y(s)$ and input $X(s)$, which is can also be used as the definition of the transfer function of an LTI system. The ROC and poles of the transfer function $H(s)$ of an LTI system dictate the behaviors of system, such as its causality and stability.

- **Stability**

Also as discussed in Chapter 1, an LTI system is stable if to any bounded input $|x(t)| < B$ its response $y(t)$ is also bounded for all t , and its impulse response function $h(t)$ needs to be absolutely integrable (Eq.1.100):

$$\int_{-\infty}^{\infty} |h(\tau)| d\tau < \infty \quad (6.73)$$

i.e., the frequency response function $\mathcal{F}[h(t)] = H(j\omega) = H(s)|_{s=j\omega}$ exists. In other words, an LTI system is stable if and only if the ROC of its transfer function $H(s)$ includes the imaginary axis $s = j\omega$.

- **Causality**

As discussed in Chapter 1, an LTI system is causal if its impulse response $h(t)$ is a consequence of the impulse input $\delta(t)$, i.e., $h(t)$ comes after $\delta(t)$:

$$h(t) = h(t)u(t) = \begin{cases} h(t) & t \geq 0 \\ 0 & t < 0 \end{cases} \quad (6.74)$$

and its output is (Eq.1.101):

$$y(t) = \int_{-\infty}^{\infty} h(\tau)x(t - \tau)d\tau = \int_0^{\infty} h(\tau)x(t - \tau)d\tau \quad (6.75)$$

The ROC of $H(s)$ is a right sided half plane. In particular, when $H(s)$ is rational, the system is causal if and only if its ROC is the right half plane to the right of the rightmost pole, and the order of numerator is no greater than that of the denominator so that $s = \infty$ is not a pole ($H(\infty)$ exists).

Combining the two properties above, we see that a causal LTI system with a rational transfer function $H(s)$ is stable if and only if all poles of $H(s)$ are in the left half of the s-plane, i.e., the real parts of all poles p_k are negative: $Re[p_k] < 0$.

One type of continuous LTI systems can be characterized by a linear constant-coefficient differential equation (LCCDE):

$$\sum_{k=0}^N a_k \frac{d^k}{dt^k} y(t) = \sum_{k=0}^M b_k \frac{d^k}{dt^k} x(t) \quad (6.76)$$

Taking the Laplace transform on both sides of this equation, we get an algebraic equation in s domain:

$$Y(s) \left[\sum_{k=0}^N a_k s^k \right] = X(s) \left[\sum_{k=0}^M b_k s^k \right] \quad (6.77)$$

The transfer function of such a system is rational:

$$H(s) = \frac{Y(s)}{X(s)} = \frac{\sum_{k=0}^M b_k s^k}{\sum_{k=0}^N a_k s^k} = \frac{b_M}{a_N} \frac{\prod_{k=0}^M (s - z_k)}{\prod_{k=0}^N (s - p_k)} \quad (6.78)$$

where z_k , ($k = 1, 2, \dots, M$) and p_k , ($k = 1, 2, \dots, N$) are the zeros and poles of $H(s)$, respectively. For simplicity and without loss of generality, we will assume $N > M$ and $b_M/a_N = 1$ below.

The output $Y(s)$ of the LTI system can be represented as

$$Y(s) = H(s)X(s) = \left(\sum_{k=0}^M b_k s^k \right) \frac{1}{\sum_{k=0}^N a_k s^k} X(s) = \left(\sum_{k=0}^M b_k s^k \right) W(s) \quad (6.79)$$

or in time domain:

$$y(t) = \sum_{k=0}^M b_k \frac{d^k w(t)}{dt^k} \quad (6.80)$$

where we have defined $W(s) = X(s)/(\sum_{k=0}^N a_k s^k)$ as an intermediate variable, or in time domain:

$$\sum_{k=0}^N a_k \frac{d^k w(t)}{dt^k} = x(t), \quad \text{or} \quad a_N \frac{d^N w(t)}{dt^N} = x(t) - \sum_{k=0}^{N-1} a_k w^{(k)}(t) \quad (6.81)$$

Without loss of generality, we assume $a_N = 1$, and the LTI system can now be represented as a *block diagram* as shown in Fig.6.1 (for $M = 2$ and $N = 3$).

To find the impulse response $h[n]$ we first convert $H(z)$ to a summation by partial fraction expansion:

$$H(s) = \frac{\sum_{k=0}^M b_k s^k}{\sum_{k=0}^N a_k s^k} = \sum_{k=1}^N \frac{c_k}{s - p_k} \quad (6.82)$$

(assume no repeated poles) and then carry out the inverse transform (the LTI system in Eq.6.76 is causal) to get:

$$h(t) = \mathcal{L}^{-1}[H(s)] = \sum_{k=1}^N \mathcal{L}^{-1} \left[\frac{c_k}{s - p_k} \right] = \sum_{k=1}^N c_k e^{p_k t} u(t) \quad (6.83)$$

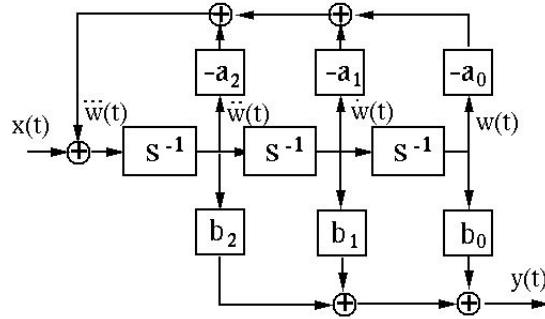


Figure 6.1 Block diagram of a continuous LTI system described by a LCCDE

The output $y(t)$ of the LTI system can be found by solving the differential equation in Eq. 6.76. Alternatively, it can also be found by the convolution $y(t) = h(t) * x(t)$, or the inverse Laplace transform:

$$y(t) = \mathcal{L}^{-1}[Y(s)] = \mathcal{L}^{-1}[H(s)X(s)] \quad (6.84)$$

As the LCCDE in Eq. 6.76 is an LTI system, it can also be solved in the following two steps. First, we assume the input on the right-hand is simply $x(t)$ and find the corresponding output $y(t)$. Then the response to the true input $\sum_k b_k d^k x(t)/dt^k$ can be found to be $\sum_k b_k d^k y(t)/dt^k$.

Note that the output $y(t)$ obtained this way is only the particular solution due to input $x(t)$, but the homogeneous solution due to non-zero initial conditions is not represented by the bilateral Laplace transform. This problem will be addressed by the unilateral Laplace transform to be discussed later, which takes the initial conditions into consideration.

According to the fundamental theorem of algebra, if all coefficients a_k of the denominator polynomial of $H(s)$ are real, then its roots p_k 's are either real or complex conjugate pairs, corresponding to the following system behaviors in time domain:

- If at least one pole $\text{Re}[p_k] > 0$ is on the right half s-plane then the corresponding term $c_k e^{p_k t} u(t)$ grows exponentially without bounds, and the system is unstable.
- If all poles $\text{Re}[p_k] < 0$ ($1 < k < N$) are on the left half s-plane, i.e., all terms in the summation of $h(t)$ above decay to zero exponentially, then $h(t)$ is absolutely integrable and the system is stable.
- Any pair of complex conjugate poles $p_{1,2} = \sigma \pm j\omega$ corresponds to a sinusoid of frequency ω :

$$e^{p_1 t} + e^{p_2 t} = e^\sigma [e^{j\omega t} + e^{-j\omega t}] = \frac{1}{2} e^\sigma [\cos(\omega t) + j\sin(\omega t)] \quad (6.85)$$

This term either decays if $\sigma < 0$ or grows if $\sigma > 0$ exponentially. In the latter case, the system is unstable.

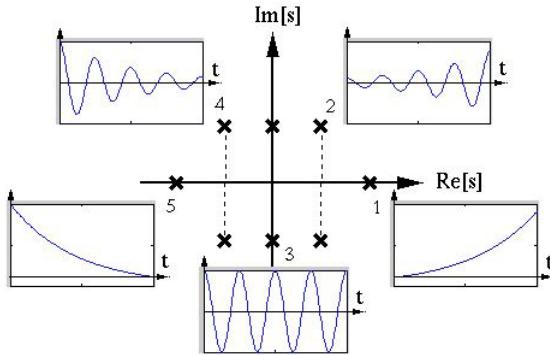


Figure 6.2 Different pole locations of $H(s)$ and the corresponding waveforms of $h(t)$

- If $0 < \text{Re}[p_2] \ll \text{Re}[p_1]$, then $e^{p_1 t}$ grows much more rapidly than $e^{p_2 t}$, i.e., the behavior of an unstable system is dominated by the rightmost pole on the right half s-plane. On the other hand, if $\text{Re}[p_2] \ll \text{Re}[p_1] < 0$, then $e^{p_1 t} = e^{-|p_1|t}$ decays much more slowly than $e^{p_2 t}$, i.e., the behavior of a stable system is also dominated by the rightmost pole on the left half s-plane. Based on this observation, the behavior of a high order system with a large number of poles can be approximated based only on its most dominant poles.

These different pole locations in s-plane and the corresponding waveforms in time domain are further illustrated in Fig.6.2 and summarized in the table below:

	Pole locations in s-plane	Waveforms in time domain
1	single real pole: $p > 0$	exponential growth: $h(t) = e^{pt}$
2	complex conjugate poles: $p_{1,2} = \sigma \pm j\omega$ ($\sigma > 0$)	exponentially growing sinusoid: $h(t) = \cos(\omega t)e^{\sigma t}$
3	complex conjugate poles: $p_{1,2} = \pm j\omega$	sinusoid: $h(t) = \cos(\omega t)$
4	complex conjugate poles: $p_{1,2} = \sigma \pm j\omega$ ($\sigma < 0$)	exponentially decaying sinusoid: $h(t) = \cos(\omega t)e^{- \sigma t}$
5	single real pole: $p < 0$	exponential decay: $h(t) = e^{- p t}$

An LTI system can be considered as a filter characterized by the magnitude and phase of its frequency response function $H(j\omega) = H(s)|_{s=j\omega}$:

$$\begin{aligned} |H(j\omega)| &= \frac{\prod_{k=1}^M |j\omega - z_k|}{\prod_{k=1}^N |j\omega - p_k|} = \frac{\prod_{k=1}^M |\mathbf{u}_k|}{\prod_{k=1}^N |\mathbf{v}_k|} \\ \angle H(j\omega) &= \frac{\sum_{k=1}^M \angle(j\omega - z_k)}{\sum_{k=1}^N \angle(j\omega - p_k)} = \frac{\sum_{k=1}^M \angle \mathbf{u}_k}{\sum_{k=1}^N \angle \mathbf{v}_k} \end{aligned} \quad (6.86)$$

where each factor $\mathbf{u}_k = j\omega - z_k$ or $\mathbf{v}_k = j\omega - p_k$ is a vector in s-plane that connects a point $j\omega$ on the imaginary axis and one of the zeros or poles. The filtering effects of the system are therefore dictated by the zero and pole locations on the s-plane and can be qualitatively determined by observing how $|H(j\omega)|$ and $\angle H(j\omega)$ change when frequency ω increase along the imaginary axis from 0 toward ∞ .

Example 6.2: The input to an LTI is

$$x(t) = e^{-3t}u(t) \quad (6.87)$$

and the output is

$$y(t) = h(t) * x(t) = (e^{-t} - e^{-2t})u(t) \quad (6.88)$$

We want to identify the system by finding $h(t)$ and $H(s)$. In s-domain, input and output signals are

$$X(s) = \frac{1}{s+3} \quad \text{Re}[s] > -3 \quad (6.89)$$

and

$$Y(s) = H(s)X(s) = \frac{1}{s+1} - \frac{1}{s+2} = \frac{1}{(s+1)(s+2)} \quad \text{Re}[s] > -1 \quad (6.90)$$

The transfer function can therefore be obtained

$$H(s) = \frac{Y(s)}{X(s)} = \frac{s+3}{(s+1)(s+2)} = \frac{s+3}{s^2+3s+2} \quad (6.91)$$

This system $H(s)$ has two poles $p_1 = -1$ and $p_2 = -2$ and therefore three possible ROCs: $\text{Re}[s] < -2$, $-2 < \text{Re}[s] < -1$ and $\text{Re}[s] > -1$ corresponding to left-sided (anti-causal), two-sided and right-sided (causal) system, respectively. To determine which of these ROCs the system has, recall that the ROC of a convolution $Y(s) = H(s) * X(s)$ should be no less than the intersection of the ROCs of $H(s)$ and $X(s)$, i.e., the ROC of $H(s)$ must be $\text{Re}[s] > -1$, i.e., the system is causal and stable. The inverse Laplace transform of $Y(s) = H(s)X(s)$ is the LCCDE of the system:

$$\frac{d^2}{dt^2}y(t) + 3\frac{d}{dt}y(t) + 2y(t) = \frac{d}{dt}x(t) + 3x(t) \quad (6.92)$$

In the following, we will consider two specific systems $H(s) = N(s)/D(s)$ where $D(s)$ is either a first order ($n = 1$) or a second order ($n = 2$) polynomial.

6.1.6 First Order System

In the transfer function $H(s)$ of a first order LTI system, the denominator $D(s)$ is a first order polynomial of order $N = 1$, and $H(s)$ is conventionally written in the following *canonic form*:

$$H(s) = \frac{N(s)}{D(s)} = \frac{1}{s - p} = \frac{1}{s + 1/\tau} \quad (6.93)$$

where τ is the system parameter called the *time constant*, and $p = -1/\tau$ is the pole of $H(s)$. In practice, $\tau > 0$ is always positive and the pole $p = -1/\tau < 0$ is on the left side of the s-plane, i.e., the system is stable.

Here we reconsider the RC circuit in Example 5.1 in Chapter 3 to illustrate the essential properties of the first order system. The input is the voltage $x(t) = v_{in}(t)$ applied across R and C in series, and the output can be either the voltage $v_C(t)$ across C or the voltage $v_R(t)$ across R . First, we let the output be $y(t) = v_C(t)$, the system can be described by a differential equation:

$$RC\dot{y}(t) + y(t) = x(t), \quad \text{i.e.,} \quad \dot{y}(t) + \frac{1}{\tau}y(t) = \frac{1}{\tau}x(t) \quad (6.94)$$

where $\tau = RC$ is the time constant of the system. Now we solve this LCCDE by taking the Laplace transform on both sides of this equation to get:

$$\left[s + \frac{1}{\tau} \right] Y(s) = \frac{1}{\tau} X(s), \quad \text{i.e.,} \quad H_C(s) = \frac{Y(s)}{X(s)} = \frac{1/\tau}{s + 1/\tau} \quad (6.95)$$

Given $H_C(s)$, we can also get $H_R(s)$ when $v_R(t)$ is treated as output based on Kirchhoff's voltage law $\delta(t) = h_C(t) + h_R(t)$:

$$H_R(s) = 1 - H_C(s) = 1 - \frac{1/\tau}{s + 1/\tau} = \frac{s}{s + 1/\tau} \quad (6.96)$$

We now consider both the impulse and step responses as well as the filtering effects of this first order system.

1. Impulse response function:

Taking the inverse Laplace transform on both sides of Eqs.6.95 and 6.96, we get:

$$h_C(t) = \mathcal{L}^{-1}[H_C(s)] = \frac{1}{\tau}e^{-t/\tau}u(t) \quad (6.97)$$

and

$$h_R(t) = \mathcal{L}^{-1}[H_R(s)] = \delta(t) - \frac{1}{\tau}e^{-t/\tau}u(t) \quad (6.98)$$

2. Step response:

The step response of this systems to input $x(t) = u(t)$ or $X(s) = 1/s$ can also be found in s-domain as:

$$Y(s) = H_C(s)X(s) = \frac{1/\tau}{s(s + 1/\tau)} = \frac{1}{s} - \frac{1}{s + 1/\tau} \quad (6.99)$$

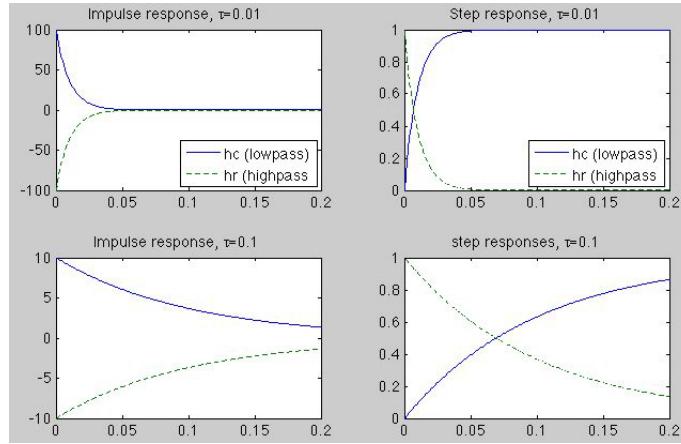


Figure 6.3 Impulse (left) and step (right) responses of first order systems

Taking inverse transform we get the step response:

$$y(t) = v_C(t) = (1 - e^{-t/\tau})u(t) \quad (6.100)$$

The step response of the system when the voltage $v_R(t)$ across R is treated as output can be obtained based on Kirchhoff's voltage law:

$$v_R(t) = u(t) - v_C(t) = u(t) - (1 - e^{-t/\tau})u(t) = e^{-t/\tau}u(t) \quad (6.101)$$

The impulse and step response functions for the two first order systems are shown in Fig.6.3.

3. First order systems as filters:

The filtering effects of the first order system are characterized by the magnitudes and phases of their frequency response functions $H(j\omega) = H(s)|_{s=j\omega}$:

$$\begin{aligned} |H_C(j\omega)| &= \left| \frac{1/\tau}{j\omega + 1/\tau} \right| = \frac{1}{\sqrt{(\omega\tau)^2 + 1}} \\ \angle H_C(j\omega) &= -\angle(j\omega + 1/\tau) = -\tan^{-1} \omega\tau \end{aligned} \quad (6.102)$$

and

$$\begin{aligned} |H_R(j\omega)| &= \left| \frac{j\omega}{j\omega + 1/\tau} \right| = \frac{\omega\tau}{\sqrt{(\omega\tau)^2 + 1}} \\ \angle H_R(j\omega) &= \angle(j\omega\tau) - \angle(j\omega\tau + 1) = \frac{\pi}{2} - \tan^{-1}(\omega\tau) \end{aligned} \quad (6.103)$$

Both the linear and Bode plots of the two systems are given in Fig.6.4, where the magnitudes of the two frequency response functions are plotted for $\tau = 0.01$ (top) and $\tau = 0.1$ (bottom), and in both linear scale (left) and Bode plots for their magnitudes (middle) and phases (right) are also plotted. We see that H_C and H_R attenuate high and low frequencies, and are therefore correspondingly low and high-pass filters, respectively.

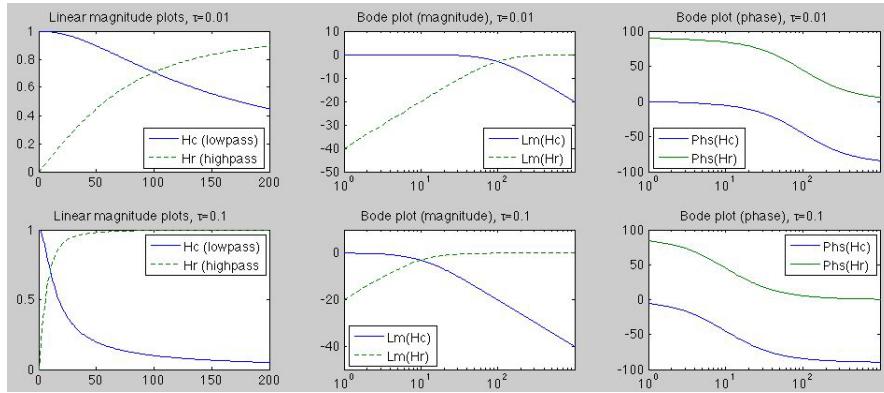


Figure 6.4 Filtering effects of first order systems

The *bandwidth* $\Delta\omega$ of the low-pass filter $H_C(j\omega)$ is defined as the interval between zero frequency at which the output power reaches its peak value and the *cutoff frequency* ω_c at which the output power is half of the peak power. As the output power is proportional to $|H_C(j\omega)|^2$ and $H_C(0) = 1$, we have

$$\frac{|H_C(j\omega_c)|^2}{|H_C(0)|^2} = \frac{1}{(\omega_c\tau)^2 + 1} = \frac{1}{2} \quad (6.104)$$

Solving for ω_c , we get the cutoff frequency $\omega_c = 1/\tau$, at which $|H_C(j\omega_c)| = 1/\sqrt{2} = 0.707$ and $Lm H(j\omega_c) = 20 \log_{10} 0.707 \approx -3 dB$.

The filtering effects of a system can be qualitatively determined based on the locations of the zeros and poles of the transfer function $H(s)$ of the system. For each point $j\omega$ along the imaginary axis representing a frequency, we define two vectors connecting $j\omega$ to the zero $s_z = 0$ of $H_R(s)$ and the common pole of both $H_C(s)$ and $H_R(s)$, respectively, as shown in Fig.6.5:

$$\mathbf{u} = j\omega, \quad \mathbf{v} = j\omega + 1/\tau \quad (6.105)$$

Now the magnitudes of $H_C(j\omega)$ and $H_R(j\omega)$ can be expressed as:

$$|H_C(j\omega)| = 1/\tau|\mathbf{v}|, \quad |H_R(j\omega)| = |\mathbf{u}|/|\mathbf{v}| \quad (6.106)$$

Based on the following two extreme cases:

- When $\omega = 0$, $|\mathbf{u}| = 0$ and $|\mathbf{v}| = 1/\tau$, we have $H_C(0) = 1$ and $H_R(0) = 0$;
- When $\omega = \infty$, $|\mathbf{v}| = |\mathbf{u}| = \infty$, we have $H_C(j\infty) = 0$ and $H_R(j\infty) = 1$.

We see that indeed $H_C(j\omega)$ and $H_R(j\omega)$ are low and high-pass filters, respectively.

6.1.7 Second Order System

In the transfer function $H(s)$ of a second order LTI system, the denominator polynomial $D(s)$ is a second order polynomial of order $N = 2$, and $H(s)$ is con-

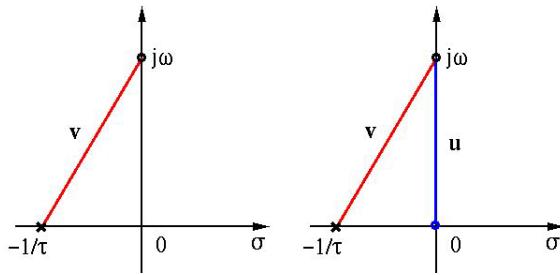


Figure 6.5 Qualitative determination of filtering behavior of first order systems

ventionally written in the following *canonic form*:

$$H(s) = \frac{N(s)}{D(s)} = \frac{N(s)}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{N(s)}{(s - p_1)(s - p_2)} \quad (6.107)$$

where ω_n and ζ in $D(s)$ are the two system parameters called *natural frequency*, which is always positive, and *damping coefficient* respectively. The two poles p_1 and p_2 of $H(s)$ are the two roots of the denominator quadratic function $D(s) = s^2 + 2\zeta\omega_n s + \omega_n^2$:

$$\begin{cases} p_1 = (-\zeta + \sqrt{\zeta^2 - 1})\omega_n = (-\zeta + j\sqrt{1 - \zeta^2})\omega_n \\ p_2 = (-\zeta - \sqrt{\zeta^2 - 1})\omega_n = (-\zeta - j\sqrt{1 - \zeta^2})\omega_n \end{cases} \quad (6.108)$$

We also have following relations:

$$p_1 p_2 = \omega_n^2, \quad p_1 + p_2 = -2\zeta\omega_n, \quad p_1 - p_2 = 2j\omega_n\sqrt{1 - \zeta^2} = 2j\omega_d \quad (6.109)$$

where

$$\omega_d = \omega_n\sqrt{1 - \zeta^2} < \omega_n \quad (6.110)$$

is called the *damped natural frequency*.

If $|\zeta| \geq 1$, both poles are real, otherwise they form a complex conjugate pair located on a circle in the s-plane with radius ω_n :

$$p_{1,2} = (-\zeta \pm j\sqrt{1 - \zeta^2})\omega_n = -\omega_n e^{\mp j\phi} \quad (6.111)$$

where, as shown in Fig.6.6

$$\phi = \tan^{-1} \left(\frac{\sqrt{1 - \zeta^2}}{\zeta} \right) \quad (6.112)$$

and

$$\sin \phi = \sqrt{1 - \zeta^2}, \quad \cos \phi = \zeta, \quad \tan \phi = \sqrt{1 - \zeta^2}/\zeta \quad (6.113)$$

As also shown in Fig.6.6, the positions of the poles on the circle are determined by the angle ϕ . When the value of ζ increases from $-\infty$ to ∞ , the pole locations change along the *root locus* in the s-plane, as shown in Fig.6.6 from which we see that each of the two poles follows its own root locus when ζ moves from $-\infty$ to ∞ :

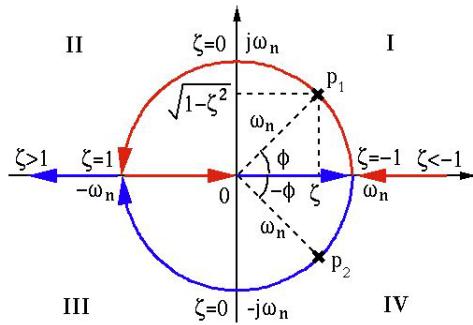


Figure 6.6 Root locus of the poles of a second order system

- Locus of p_1 : $\infty \Rightarrow \omega_n \Rightarrow j\omega_n \Rightarrow -\omega_n \Rightarrow 0$
- Locus of p_2 : $0 \Rightarrow \omega_n \Rightarrow -j\omega_n \Rightarrow -\omega_n \Rightarrow -\infty$

The root locus is further summarized in the table below:

ζ	p_1, p_2	comments on poles
$\zeta = -\infty$	$\infty, 0$	
$-\infty < \zeta < -1$	$(-\zeta \pm \sqrt{\zeta^2 - 1})\omega_n$	real, $0 < p_2 < p_1$
$\zeta = -1$	ω_n	real, repeated, $0 < p_1 = p_2 = \omega_n$
$-1 < \zeta < 0$	$(-\zeta \pm j\sqrt{1 - \zeta^2})\omega_n$	conjugate pair in quadrants I, VI
$\zeta = 0$	$\pm j\omega_n$	imaginary pair
$0 < \zeta < 1$	$(-\zeta \pm j\sqrt{1 - \zeta^2})\omega_n$	conjugate pair in quadrants II, III
$\zeta = 1$	$-\omega_n$	real, repeated $p_1 = p_2 = -\omega_n < 0$
$1 < \zeta < \infty$	$(-\zeta \pm \sqrt{\zeta^2 - 1})\omega_n$	real, $p_2 < p_1 < 0$
$\zeta = \infty$	$0, -\infty$	

We see that only when $\zeta > 0$, will the two poles $p_{1,2}$ be in the left half of the s-plane and the system is stable. When $\zeta = 0$, the poles are on the imaginary axis and system is marginally stable, and when $\zeta < 0$ the poles are on the right half plane and the system is unstable.

The behavior of a second order system in terms of its impulse response function $h(t)$ is determined by the two system parameters ω_n and ζ , which are directly associated with the locations of the poles of the transfer function $H(s)$. In the following, we show how $h(t)$ can be determined by inverse Laplace transform of $H(s)$, based on the given pole locations in the s-plane. Here we assume $N(s) = 1$ so that the transfer function is:

$$H(s) = \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{1}{(s - p_1)(s - p_2)} \quad (6.114)$$

If $\zeta = \pm 1$, we have

$$H(s) = \frac{1}{s^2 \pm 2\omega_n s + \omega_n^2} = \frac{1}{(s \pm \omega_n)^2} = \frac{1}{(s - p)^2} \quad (6.115)$$

where $p = \pm\omega_n$ is the repeated pole of $H(s)$, then we have:

$$h(t) = \mathcal{L}^{-1}[H(s)] = t e^{\pm\omega_n t} u(t) \quad (6.116)$$

If $|\zeta| \neq 1$, then $p_1 \neq p_2$, and $H(s)$ can be written the following by partial fraction expansion:

$$H(s) = \frac{1}{p_1 - p_2} \left[\frac{1}{s - p_1} - \frac{1}{s - p_2} \right] \quad (6.117)$$

and the impulse response can be found by inverse Laplace transform:

$$h(t) = \mathcal{L}^{-1}[H(s)] = \frac{1}{p_1 - p_2} [e^{p_1 t} - e^{p_2 t}] u(t) = C [e^{p_1 t} - e^{p_2 t}] u(t) \quad (6.118)$$

where

$$C = \frac{1}{p_1 - p_2} = \frac{1}{2\omega_n \sqrt{\zeta^2 - 1}} = \frac{1}{2j\omega_n \sqrt{1 - \zeta^2}} \quad (6.119)$$

In the following we consider specifically each of the cases listed in Table 6.1.7 to see how $h(t)$ given in Eq.6.118 varies when the value of ζ changes from $-\infty$ to ∞ .

- $-\infty < \zeta < -1$, $0 < p_2 < p_1$, both poles are on the real axis on the right side of the s-plane, and both terms $e^{p_1 t}$ and $e^{p_2 t}$ grow exponentially as $t \rightarrow \infty$, so does their difference, i.e., the system is unstable:

$$h(t) = C [e^{p_1 t} - e^{p_2 t}] u(t), \quad (p_1 > p_2) \quad (6.120)$$

- $\zeta = -1$, $p_1 = p_2 = -\zeta\omega_n = \omega_n$ are repeated poles still on the right side of the s-plane. We have:

$$h(t) = t e^{\omega_n t} u(t) \quad (6.121)$$

which grows without bound when $t \rightarrow \infty$, the system is unstable.

- $-1 < \zeta < 0$, the two poles form a conjugate pair in quadrants I and IV, respectively:

$$p_{1,2} = (-\zeta \pm j\sqrt{1 - \zeta^2})\omega_n = -\omega_n\zeta \pm j\omega_d \quad (6.122)$$

Now we have:

$$h(t) = \frac{1}{2j\omega_n \sqrt{1 - \zeta^2}} e^{-\zeta\omega_n t} [e^{j\omega_d t} - e^{-j\omega_d t}] u(t) = \frac{e^{-\zeta\omega_n t}}{\omega_d} \sin(\omega_d t) u(t) \quad (6.123)$$

As $\zeta < 0$ and therefore $-\zeta\omega_n t > 0$, $h(t)$ is an exponentially growing sinusoid, the system is still unstable.

- $\zeta = 0$, $p_{1,2} = \pm j\omega_n$ are on the imaginary axis, and the system is marginally stable:

$$h(t) = \frac{1}{2j\omega_n} [e^{j\omega_n t} - e^{-j\omega_n t}] u(t) = \frac{1}{\omega_n} \sin(\omega_n t) u(t) \quad (6.124)$$

In particular, when the frequency of the input $x(t) = e^{j\omega_n t}$ is the same as the system's natural frequency ω_n , the output can be found to be (Eq.6.69):

$$y(t) = H(s)|_{s=j\omega_n} e^{j\omega_n t} = \frac{1}{s^2 + \omega_n^2}|_{s=j\omega_n} e^{j\omega_n t} = \frac{e^{j\omega_n t}}{\omega_n^2 - \omega_n^2} = \infty \quad (6.125)$$

The response of the system becomes infinity, i.e., *resonance* occurs.

- $0 < \zeta < 1$, the two poles form a complex conjugate pair in quadrants II and III, respectively. Similar to the case when $-1 < \zeta < 0$, we have the same expression for $h(t)$:

$$h(t) = \frac{e^{-\zeta\omega_n t}}{\omega_n \sqrt{1-\zeta^2}} \sin(\omega_d t) u(t) = \frac{e^{-\zeta\omega_n t}}{\omega_d} \sin(\omega_d t) u(t) \quad (6.126)$$

As $\zeta > 0$, p_1 and p_2 are on the left half plane, and the impulse response $h(t)$ is an exponentially decaying sinusoid with frequency ω_d , the system is *underdamped* and stable.

- $\zeta = 1$, $p_1 = p_2 = -\zeta\omega_n = -\omega_n < 0$ are two repeated poles on the left side, the system is *critically damped* and stable.

$$h(t) = t e^{-\omega_n t} u(t) \quad (6.127)$$

- $1 < \zeta < \infty$, $p_2 < p_1 < 0$, both poles are on the real axis on the left of the s-plane, the impulse response is the difference of two exponentially decaying functions:

$$h(t) = C(e^{p_1 t} - e^{p_2 t}) u(t) = C(e^{-|p_1|t} - e^{-|p_2|t}) u(t), \quad (|p_1| < |p_2|) \quad (6.128)$$

which decays to zero in time. The system is *overdamped* and stable.

All seven cases considered above are summarized in the table below, as a continuation of Table 6.1.7.

ζ	$H(s)$	$h(t) = C(e^{p_1 t} - e^{p_2 t})$	Comments
$\zeta < -1$		$C(e^{p_1 t} - e^{p_2 t}) u(t)$	exponential growth
$\zeta = -1$	$1/(s - \omega_n)^2$	$t e^{\omega_n t} u(t)$	exponential growth
$-1 < \zeta < 0$		$\frac{e^{-\zeta\omega_n t}}{\omega_d} \sin(\omega_d t) u(t)$	exponentially growing sinusoid
$\zeta = 0$	$1/(s^2 + \omega_n^2)$	$\frac{1}{\omega_n} \sin(\omega_n t) u(t)$	sinusoid
$0 < \zeta < 1$		$\frac{e^{-\zeta\omega_n t}}{\omega_d} \sin(\omega_d t) u(t)$	exponentially decaying sinusoid
$\zeta = 1$	$1/(s + \omega_n)^2$	$t e^{-\omega_n t} u(t)$	critically damped
$\zeta > 1$		$C(e^{- p_1 t} - e^{- p_2 t}) u(t)$	exponential decay

These different impulse response functions $h(t)$ corresponding to different values of ζ are plotted in Fig.6.7. Note in particular the following two cases:

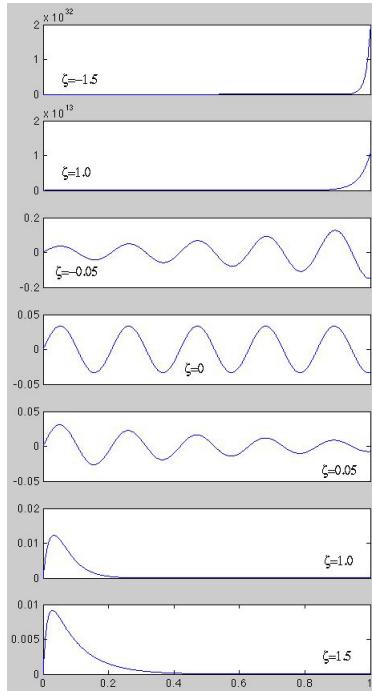


Figure 6.7 Impulse response of 2nd order system for different ζ

- $\zeta \ll -1$, we have $0 < p_2 \ll p_1$ and

$$h(t) = C(e^{p_1 t} - e^{p_2 t})u(t) \approx Ce^{p_1 t} \quad (6.129)$$

i.e., p_1 which is farther away from the origin dominates the system behavior.

- $\zeta \gg 1$, we have $p_2 \ll p_1 < 0$ and

$$h(t) = C(e^{-|p_1|t} - e^{-|p_2|t})u(t) \approx Ce^{-|p_1|t} \quad (6.130)$$

i.e., p_1 which is closer to the origin dominates the system behavior.

In either case, when the non-dominant pole can be neglected, the behavior of the second order system can be approximated by a first order system with a single pole $p = -1/\tau$.

As a typical example of the second order system, consider a circuit composed of a resistor R , a capacitor C and an inductor L connected in series as shown in Fig.6.8. An input voltage $v(t)$ is applied to the series combination of the three elements and the output is $v_L(t)$, $v_R(t)$, or $v_C(t)$, the voltage across one of the three elements. The system is described by the following differential equation in time domain:

$$v(t) = v_L(t) + v_R(t) + v_C(t) = L \frac{d}{dt} i(t) + R i(t) + \frac{1}{C} \int_{-\infty}^t i(\tau) d\tau \quad (6.131)$$

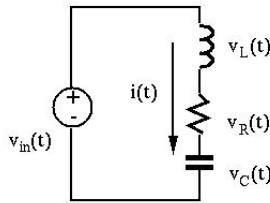


Figure 6.8 A 2nd order RCL series circuit

Taking Laplace transform on both sides, we get an algebraic equation in s-domain:

$$\begin{aligned} X(s) &= V_L(s) + V_R(s) + V_C(s) = \left[sL + R + \frac{1}{sC} \right] I(s) \\ &= [Z_L + Z_R + Z_C]I(s) = Z(s)I(s) \end{aligned}$$

where

$$Z_L(s) = \frac{V_L(s)}{I(s)} = sL, \quad Z_R = \frac{V_R(s)}{I(s)} = R, \quad Z_C(s) = \frac{V_C(s)}{I(s)} = 1/sC \quad (6.132)$$

are the *impedances* of the circuit elements L , R and C , respectively, defined as the ratio between the voltage across and current through each of the components in s-domain, similar to the resistive $R = v(t)/i(t)$ of a resistor R defined by Ohm's law as the ratio between the voltage and current in time domain. The total impedance $Z(s)$ of the three elements in series is the sum of the individual impedances:

$$Z(s) = \frac{V(s)}{I(s)} = sL + R + \frac{1}{sC} = Z_L + Z_R + Z_C \quad (6.133)$$

	capacitor C	resistor R	inductor L
time domain	$v_C(t) = \int i(t)dt/C$	$v_R(t) = Ri(t)$	$v_L(t) = Li'(t)$
s-domain	$V_C(s) = I(s)/Cs$	$V_R(s) = RI(s)$	$V_L(s) = I(s)sL$
impedance $Z(s) = V(s)/I(s)$	$1/sC$	R	sL

The transfer function $H(s)$, the ratio between the voltage across one of the three elements (V_L , V_R , or V_C) and input voltages $V(s)$, can be found by treating the series circuit as a voltage divider:

- Output is voltage across the capacitor $v_C(t)$

$$H_C(s) = \frac{V_C(s)}{V(s)} = \frac{Z_C(s)}{Z(s)} = \frac{1/sC}{Ls + R + 1/sC} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (6.134)$$

- Output is voltage across the resistor $v_R(t)$

$$H_R(s) = \frac{V_R(s)}{V(s)} = \frac{Z_R(s)}{Z(s)} = \frac{R}{Ls + R + 1/sC} = \frac{2\zeta\omega_n s}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (6.135)$$

- Output is voltage across the inductor $v_L(t)$

$$H_L(s) = \frac{V_L(s)}{V(s)} = \frac{Z_L(s)}{Z(s)} = \frac{sL}{Ls + R + 1/sC} = \frac{s^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (6.136)$$

Here we have converted the denominator $D(s)$ into the canonical second order form:

$$D(s) = s^2 + (R/L)s + (1/LC) = s^2 + 2\zeta\omega_n s + \omega_n^2 = (s - p_1)(s - p_2) \quad (6.137)$$

where the damping coefficient ζ and natural frequency ω_n are defined as:

$$\zeta = \frac{R}{2}\sqrt{\frac{C}{L}} > 0, \quad \omega_n = \frac{1}{\sqrt{LC}} > 0 \quad (6.138)$$

If we assume $0 < \zeta < 1$ then the two poles are:

$$p_{1,2} = (-\zeta \pm j\sqrt{1 - \zeta^2})\omega_n = -\omega_n e^{\mp j\phi} \quad (6.139)$$

where $\phi = \tan^{-1}(\sqrt{1 - \zeta^2}/\zeta)$ as defined in Eq.6.112.

In the following, we further consider some important characteristics of the second order systems in both time and frequency domains.

1. **Impulse response function:** When voltage $v_C(t)$ across C is treated as the output, the impulse response $h_C(t)$ can be found by inverse transform of Eq.6.134:

$$\begin{aligned} h_C(t) &= \mathcal{L}^{-1}[H_C(s)] = \mathcal{L}^{-1}\left[\frac{\omega_n^2}{(s - p_1)(s - p_2)}\right] \\ &= \frac{\omega_n e^{-\zeta\omega_n t}}{\sqrt{1 - \zeta^2}} \sin(\omega_n t) u(t) \end{aligned} \quad (6.140)$$

This is based on the assumption $0 < \zeta < 1$ (Eq.6.126 multiplied by ω_n^2). Alternatively, when the voltage across R or L is treated as the output, the corresponding impulse response $h_R(t)$ or $h_L(t)$ can also be found by inverse transform of Eqs.6.135 or 6.136. The derivation of these impulse responses $h_L(t)$ and $h_R(t)$ are left as homework problems, but their waveforms are plotted together with $h_C(t)$ obtained above in Fig.6.9 ($\zeta = 0.05$), from which we see that the three responses do add up to the step input $\delta(t) = h_C(t) + h_L(t) + h_R(t)$, i.e., Kirchhoff's voltage law holds.

2. **Step response:**

When $V_C(s)$ across C is treated as the output, in s-domain the step response to a step input $U(s) = \mathcal{L}[u(t)] = 1/s$ is:

$$\begin{aligned} Y_C(s) &= H_C(s)U(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \frac{1}{s} \\ &= \frac{1}{s} + \frac{p_2}{p_1 - p_2} \frac{1}{s - p_1} - \frac{p_1}{p_1 - p_2} \frac{1}{s - p_2} \end{aligned} \quad (6.141)$$

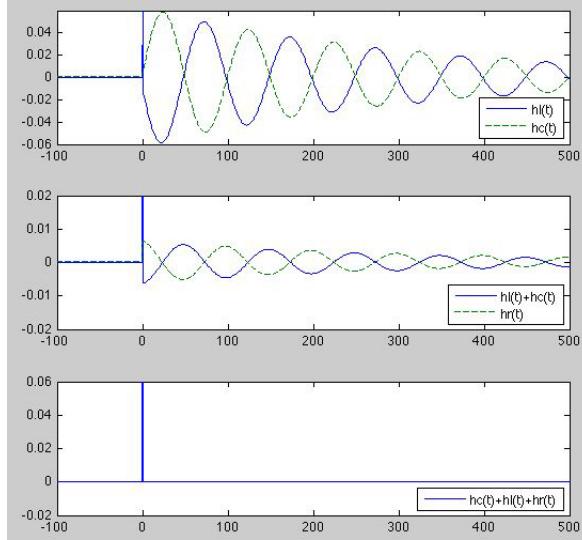


Figure 6.9 Impulse responses by R , L and C of an RCL system

Top: impulse responses $h_L(t)$ (solid curve) and $h_C(t)$ (dashed curve); Middle: their sum $h_L(t) + h_C(t)$ (solid curve) and impulse response $h_R(t)$ (dashed curve); Bottom: the sum of all three: $\delta(t) = h_L(t) + h_R(t) + h_C(t)$.

and the step response in time domain can be obtained by inverse transform:

$$\begin{aligned}
 y_C(t) &= \mathcal{L}^{-1}[Y(s)] = \left[1 + \frac{1}{p_1 - p_2} (p_2 e^{p_1 t} - p_1 e^{p_2 t}) \right] u(t) \\
 &= \left[1 - \frac{\omega_n}{p_1 - p_2} (e^{j\phi} e^{(-\zeta\omega_n + j\omega_d)t} - e^{-j\phi} e^{(-\zeta\omega_n - j\omega_d)t}) \right] u(t) \\
 &= \left[1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1 - \zeta^2}} \sin(\omega_d t + \phi) \right] u(t)
 \end{aligned} \tag{6.142}$$

This step response function is plotted in Fig.6.10 for different ζ values.

Alternatively, the voltage across R or L can also be treated as the output of the 2nd order system, and we can find the system's step response in both s and time domains for these cases. The derivation of these step responses $y_L(t)$ and $y_R(t)$ are left as homework problems, but their waveforms are plotted together with $y_C(t)$ in Fig.6.11 ($\zeta = 0.05$), from which we see that the three responses do add up to the step input $u(t) = y_C(t) + y_L(t) + y_R(t)$, i.e., Kirchhoff's voltage law holds.

3. Second order systems as filters:

The filtering effects of the three second order systems are characterized by the magnitudes and phases of their frequency response functions $H_C(j\omega)$, $H_R(j\omega)$, and $H_L(j\omega)$, as plotted in Fig.6.12, based on the assumed parameters $\omega_n = 2\pi 1000$ and $\zeta = 0.1$ (top) and $\zeta = 1/\sqrt{2} = 0.707$ (bottom). We see that when $\zeta = 0.1 < 0.707$, both $H_C(j\omega)$ and $H_L(j\omega)$ behave like a bandpass filter

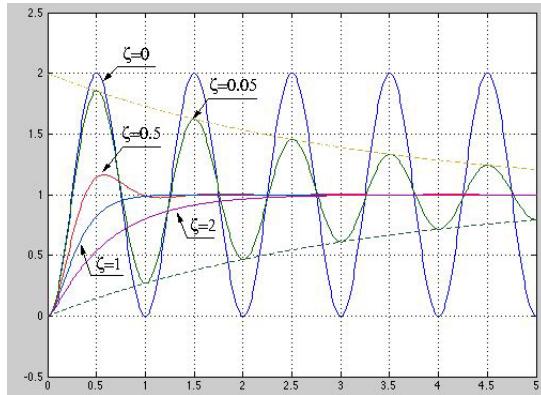


Figure 6.10 Step response of 2nd order system for different ζ

Step responses corresponding to five different values of ζ : 0, 0.05, 0.5, 1, and 2. The envelop of the step response for $\zeta = 0.05$ is also plotted to show the exponential decay of the sinusoid.

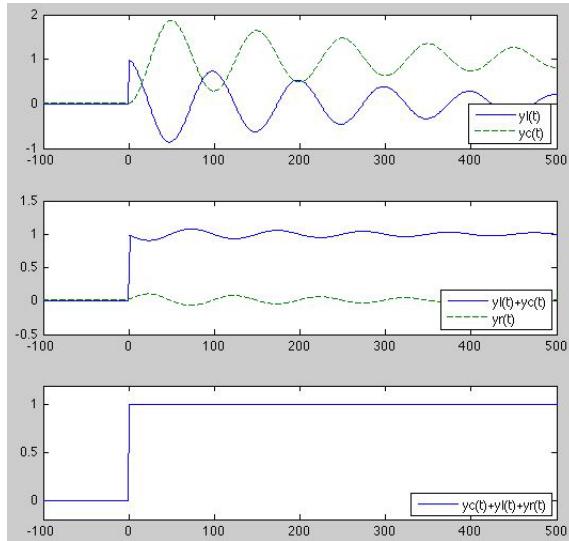


Figure 6.11 Step responses by R , L as well as C of an RCL system

Top: step responses $y_L(t)$ (solid curve) and $y_C(t)$ (dashed curve); Middle: their sum $y_L(t) + y_C(t)$ (solid curve) and step responses $y_R(t)$ (dashed curve); Bottom: the sum of all three: $y_C(t) + y_L(t) + y_R(t) = u(t)$.

similar to $H_R(j\omega)$ (top row), but when $\zeta \geq 0.707$, they behave as low-pass and high-pass filters without any peak (bottom row), respectively.

The filtering effects of the three systems can be qualitatively estimated based on the location of the zeros and poles of their corresponding transfer functions. We first define three vectors connecting an arbitrary point $j\omega$ on the imaginary

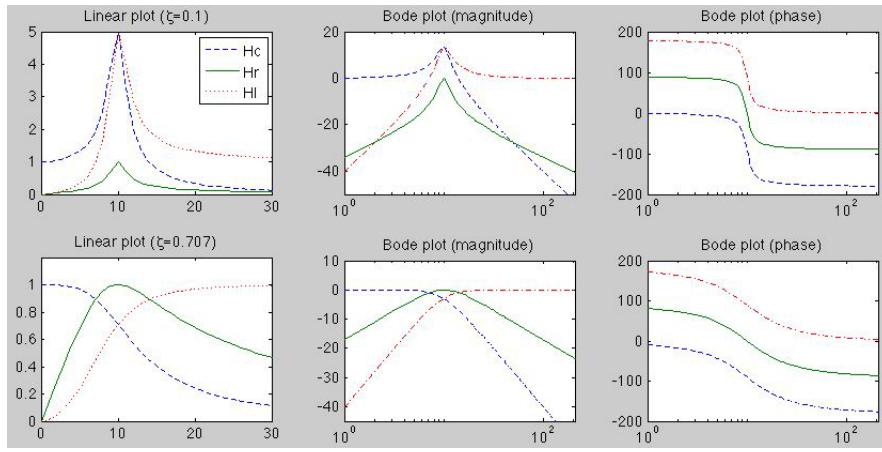


Figure 6.12 Frequency response functions $H_C(j\omega)$, $H_R(j\omega)$, and $H_L(\omega)$

Top: $\zeta = 0.1$, Bottom: $\zeta = 0.707$; Left: linear magnitude plots, Middle and Right: Bode log-magnitude and phase plots.

axis to the origin and each of the poles:

$$\mathbf{u} = j\omega, \quad \mathbf{v}_1 = j\omega - p_1, \quad \mathbf{v}_2 = j\omega - p_2 \quad (6.143)$$

and then observe how each of the three frequency response functions changes when ω increase from 0 toward ∞ , as illustrated in Fig.6.13:

- $H_C(s) = \omega_n^2/D(s)$ with two poles but no zero:

$$|H_C(j\omega)| = \frac{\omega_n^2}{|j\omega - p_1||j\omega - p_2|} = \frac{\omega_n^2}{|\mathbf{v}_1||\mathbf{v}_2|} \quad (6.144)$$

which is some constant when $\omega = 0$ but approaches 0 when $\omega \rightarrow \infty$ causing both $|\mathbf{v}_1| \rightarrow \infty$ and $|\mathbf{v}_2| \rightarrow \infty$, i.e., the system is a low-pass filter.

- $H_R(s) = 2\zeta\omega_n s/D(s)$ with two poles and one zero:

$$|H_R(j\omega)| = \frac{2\zeta\omega_n |j\omega|}{|j\omega - p_1||j\omega - p_2|} = \frac{2\zeta\omega_n |\mathbf{u}|}{|\mathbf{v}_1||\mathbf{v}_2|} \quad (6.145)$$

which is zero when $\omega = 0$ or $\omega \rightarrow \infty$, but greater than 0 when $0 < \omega < \infty$, i.e., the system is a band-pass filter.

- $H_L(s) = s^2/D(s)$ with two poles and two repeated zeros (corresponding to two vectors $\mathbf{u}_1 = \mathbf{u}_2 = \mathbf{u}$):

$$|H_L(j\omega)| = \frac{|j\omega|^2}{|j\omega - p_1||j\omega - p_2|} = \frac{|\mathbf{u}|^2}{|\mathbf{v}_1||\mathbf{v}_2|} \quad (6.146)$$

which is zero when $\omega = 0$, but approaches constant 1 when $\omega \rightarrow \infty$, i.e., the system is a high-pass filter.

4. Peak frequency of second order filters:

The *peak frequency* ω_p of a filter $H(j\omega)$ is the frequency at which $|H(j\omega_p)| = |H_{max}|$ is maximized. To simplify the algebra, we first define a variable $u =$

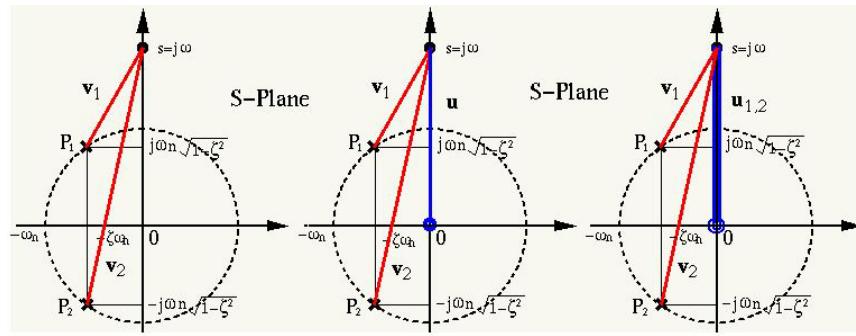


Figure 6.13 Graphic determination of filtering behavior of second order systems

$(\omega/\omega_n)^2$ (frequency ω normalized by ω_n) so that the squared magnitudes of the frequency response functions can be expressed as:

$$\begin{aligned} |H_C(j\omega)|^2 &= \left| \frac{\omega_n^2}{(j\omega)^2 + 2\zeta\omega_n j\omega + \omega_n^2} \right|^2 = \frac{1}{(u-1)^2 + 4\zeta^2 u} \\ |H_R(j\omega)|^2 &= \left| \frac{2\zeta\omega_n j\omega}{(j\omega)^2 + 2\zeta\omega_n j\omega + \omega_n^2} \right|^2 = \frac{4\zeta^2 u}{(u-1)^2 + 4\zeta^2 u} \\ |H_L(j\omega)|^2 &= \left| \frac{(j\omega)^2}{(j\omega)^2 + 2\zeta\omega_n j\omega + \omega_n^2} \right|^2 = \frac{u^2}{(u-1)^2 + 4\zeta^2 u} \end{aligned} \quad (6.147)$$

To find the value u_p at which each of these functions is maximized, we take derivative of each of the functions with respect to u , set the results to zero, and then solve the resulting equations to get:

$$\begin{cases} u_{p_C} = 1 - 2\zeta^2 \\ u_{p_R} = 1 \\ u_{p_L} = 1/(1 - 2\zeta^2) \end{cases} \quad \text{i.e.} \quad \begin{cases} \omega_{p_C} = \omega_n \sqrt{u_{p_C}} = \omega_n \sqrt{1 - 2\zeta^2} \\ \omega_{p_R} = \omega_n \sqrt{u_{p_R}} = \omega_n \\ \omega_{p_L} = \omega_n \sqrt{u_{p_L}} = \omega_n / \sqrt{1 - 2\zeta^2} \end{cases} \quad (6.148)$$

We see that the three peak frequencies are different:

$$\omega_{C_p} \leq \omega_{R_p} \leq \omega_{L_p} \quad (6.149)$$

Substituting these peak frequencies into Eq.6.147, we get the peak values of the three filters:

$$\begin{aligned} |H_{max_R}| &= |H(j\omega_{p_R})| = 1 \\ |H_{max_C}| &= |H(j\omega_{p_C})| = |H_{max_L}| = |H(j\omega_{p_L})| = \frac{1}{2\zeta\sqrt{1-\zeta^2}} \end{aligned} \quad (6.150)$$

Also note that for the peak frequencies ω_{P_c} and ω_{P_L} given in Eq.6.148 to be real, the following has to be satisfied:

$$1 - 2\zeta^2 > 0, \quad \text{i.e.,} \quad \zeta < 1/\sqrt{2} = 0.707 \quad (6.151)$$

Otherwise these peak frequencies do not exist, and $H_C(j\omega)$ becomes a low-pass filter that reaches its maximum of 1 at $\omega = 0$, and $|H_L(j\omega)|$ becomes a high-pass filter that reaches its maximum of 1 at $\omega = \infty$, as shown in Fig.6.12.

5. Bandwidth of second order bandpass filter:

The bandwidth $\Delta\omega = \omega_1 - \omega_2$ of a bandpass filter $H(j\omega)$ is defined as the interval between two *cutoff frequencies* ω_1 and ω_2 at which the output power is half of that at the peak frequency ω_p :

$$|H(j\omega_1)|^2 = |H(j\omega_2)|^2 = \frac{1}{2}|H(j\omega_p)|^2 = \frac{1}{2}|H_{max}|^2 \quad (6.152)$$

Specifically, for the bandpass filter $H_R(j\omega)$, $H_{max}(0) = 1$, and at the two cutoff frequencies we have:

$$\frac{|H_R(j\omega)|^2}{|H_{max,R}|^2} = |H_R(j\omega)|^2 = \frac{4\zeta^2 u}{(u-1)^2 + 4\zeta^2 u} = \frac{1}{2} \quad (6.153)$$

Solving this quadratic equation we get two solutions:

$$u_{1,2} = 1 + 2\zeta^2 \pm 2\zeta\sqrt{1 + \zeta^2} \quad (6.154)$$

and the corresponding cutoff frequencies are:

$$\omega_{1,2} = \omega_n \sqrt{1 + 2\zeta^2 \pm 2\zeta\sqrt{1 + \zeta^2}} \quad (6.155)$$

and the bandwidth is:

$$\Delta\omega_R = \omega_1 - \omega_2 = 2\zeta\omega_n \quad (6.156)$$

Based on this result, the denominator of the second order transfer function can also be written as:

$$D(s) = s^2 + \Delta\omega_n s + \omega_n^2 \quad (6.157)$$

6.1.8 The Unilateral Laplace Transform

When applied to solving LCCDEs, the bilateral Laplace transform considered so far can only find the particular solutions, but not the homogeneous solution due to non-zero initial conditions, which are not taken into consideration. This problem can be overcome by the *unilateral* or one-sided Laplace transform, which can solve a given LCCDE to find the homogeneous as well as the particular solution.

The unilateral Laplace transform of a given signal $x(t)$ is defined as

$$\mathcal{U}L[x(t)] = \int_{-\infty}^{\infty} x(t)u(t)e^{-st}dt = \int_0^{\infty} x(t)e^{-st}dt \quad (6.158)$$

When the unilateral Laplace transform is applied to a signal $x(t)$, it is always assumed that the signal starts at time $t = 0$, i.e., $x(t) = 0$ for all $t < 0$. When it is applied to the impulse response function $h(t)$ of an LTI system to find the transfer function $H(s) = \mathcal{U}L[h(t)]$, it is always assumed that its impulse response

$h(t) = 0$ for $t < 0$, i.e., the system is causal. In either case, all poles have to be on the left half s-plane, i.e., the ROC is always in the right half s-plane. Obviously, if $x(t) = x(t)u(t)$, its unilateral and bilateral Laplace transforms are identical. Otherwise the two Laplace transforms are different.

The unilateral Laplace Transform shares all of the properties of the bilateral Laplace transform, although some may be expressed in different forms. Here we will not repeat all the properties except the following, which are most relevant to solving the LCCDE of an LTI system.

- **Time derivative:**

$$\mathcal{U}L\left[\frac{d}{dt}x(t)\right] = sX(s) - x(0) \quad (6.159)$$

Proof:

$$\begin{aligned} \mathcal{U}L\left[\frac{d}{dt}x(t)\right] &= \int_0^\infty \left[\frac{d}{dt}x(t)\right] e^{-st} dt = \int_0^\infty e^{-st} d[x(t)] \\ &= x(t)e^{-st}\Big|_0^\infty - \int_0^\infty x(t)d(e^{-st}) = -x(0) + s \int_0^\infty x(t)e^{-st} dt = sX(s) - x(0) \end{aligned} \quad (6.160)$$

We can further get the transform of the 2nd derivative of $x(t)$:

$$\mathcal{U}L\left[\frac{d^2}{dt^2}x(t)\right] = s \mathcal{U}L\left[\frac{d}{dt}x(t)\right] - \dot{x}(0) = s^2X(s) - sx(0) - \dot{x}(0) \quad (6.161)$$

and in general we have:

$$\mathcal{U}L[x^{(n)}(t)] = s^n X(s) - \sum_{k=0}^{n-1} s^k x^{(n-1-k)}(0) \quad (6.162)$$

- **The initial value theorem:**

If a right-sided signal $x(t)$ containing no impulse or higher order singularities at $t = 0$, its initial value $x(0^+)$ ($t \rightarrow 0$ from $t > 0$) can be found to be:

$$x(0^+) = \lim_{t \rightarrow 0} x(t) = \lim_{s \rightarrow \infty} sX(s) \quad (6.163)$$

Proof: At the limit $s \rightarrow 0$, Eq.6.160 becomes:

$$\lim_{s \rightarrow 0} \int_0^\infty \frac{d}{dt}x(t)e^{-st} dt = \int_0^\infty dx(t) = x(\infty) - x(0) = \lim_{s \rightarrow 0} [sX(s) - x(0)] \quad (6.164)$$

i.e.,

$$\lim_{s \rightarrow 0} sX(s) = x(\infty) \quad (6.165)$$

- **The final value theorem:**

If a right-sided signal $x(t)$ approaches a finite value $x(\infty)$ as $t \rightarrow \infty$, it can be found to be:

$$x(\infty) = \lim_{t \rightarrow \infty} x(t) = \lim_{s \rightarrow 0} sX(s) \quad (6.166)$$

Proof: At the limit $s \rightarrow \infty$, Eq.6.160 becomes:

$$\lim_{s \rightarrow \infty} \int_0^\infty \frac{d}{dt} x(t) e^{-st} dt = 0 = \lim_{s \rightarrow \infty} [sX(s) - x(0)] \quad (6.167)$$

i.e.,

$$\lim_{s \rightarrow \infty} sX(s) = x(0) \quad (6.168)$$

Due to these properties, the unilateral Laplace transform is a useful tool for solving LCCDEs with non-zero initial conditions.

Example 6.3: We consider Example 5.1 in Chapter 5 one more time, where the LCCDE of the first order system is:

$$\tau \dot{y}(t) + y = x \quad (6.169)$$

and $y(0) = y_0$ is the initial condition. Taking the unilateral Laplace transform on both sides, we get:

$$\tau[sY(s) - y_0] + Y(s) = X(s), \quad \text{i.e.} \quad Y(s) = \frac{X(s)}{s\tau + 1} + \frac{\tau y_0}{s\tau + 1} \quad (6.170)$$

Consider the following two inputs:

- When $x(t) = \delta(t)$, $X(s) = 1$ and the output is:

$$Y(s) = \frac{1}{s\tau + 1} + \frac{\tau y_0}{s\tau + 1} \quad (6.171)$$

Taking inverse transform we get:

$$y(t) = \left(\frac{1}{\tau} + y_0 \right) e^{-t/\tau} u(t) = \frac{1}{\tau} e^{-t/\tau} u(t) + y_0 e^{-t/\tau} u(t) \quad (6.172)$$

This first term is the particular solution representing the discharge of the capacitor charge $Q = \frac{1}{RC} \int \delta(t) dt = u(t)/\tau$ instantly charged by the input voltage $x(t) = \delta(t)$, and the second term is the homogeneous solution representing the discharge of the initial voltage y_0 . Comparing the result above to Eq.6.97, we see that that the bilateral Laplace transform fails to find the homogeneous solution.

- When $x(t) = u(t)$, $X(s) = 1/s$ and the output is:

$$Y(s) = \frac{1}{s} \frac{1}{s\tau + 1} + \frac{\tau y_0}{s\tau + 1} = \frac{1}{s} - \frac{\tau}{s\tau + 1} + \frac{\tau y_0}{s\tau + 1} \quad (6.173)$$

Taking inverse transform we get:

$$y(t) = [1 + (y_0 - 1)e^{-t/\tau}] u(t) = (1 - e^{-t/\tau}) u(t) + y_0 e^{-t/\tau} u(t) \quad (6.174)$$

The first term is the particular solution representing the charge of the capacitor C by the input $x(t) = u(t)$, while the second is the homogeneous solution representing the discharge of the initial voltage $y(0) = y_0$. Comparing the

result above to Eq.6.100, we see that that the bilateral Laplace transform fails to find the homogeneous solution.

All these results are consistent with Example 5.1. Note that the bilateral Laplace transform fails to find the homogeneous solutions.

Example 6.4: Solve the following 2nd-order LCCDE:

$$\frac{d^2}{dt^2}y(t) + 3\frac{d}{dt}y(t) + 2y(t) = x(t) = \alpha u(t) \quad (6.175)$$

with initial conditions

$$y(0) = \beta, \quad \dot{y}(0) = \gamma \quad (6.176)$$

Applying the unilateral Laplace transform to the LCCDE we get:

$$\begin{aligned} s^2Y(s) - \beta s - \gamma + 3sY(s) - 3\beta + 2Y(s) \\ = (s^2 + 3s + 2)Y(s) - \beta s - \gamma - 3\beta = \alpha/s \end{aligned} \quad (6.177)$$

Solving for $Y(s)$ we get:

$$Y(s) = \frac{\alpha}{s(s+1)(s+2)} + \frac{\beta(s+3)}{(s+1)(s+2)} + \frac{\gamma}{(s+1)(s+2)} = Y_p(s) + Y_h(s) \quad (6.178)$$

This is the general solution of the LCCDE which is composed of two parts:

- **The homogeneous (zero-input) solution:** due to the non-zero initial conditions $\beta \neq 0$ and $\gamma \neq 0$ with zero input $\alpha = 0$:

$$Y_h(s) = \frac{\beta(s+3)}{(s+1)(s+2)} + \frac{\gamma}{(s+1)(s+2)} \quad (6.179)$$

- **The particular (zero-state) solution:** due to the non-zero input $\alpha \neq 0$ but with zero initial conditions $\beta = \gamma = 0$:

$$Y_p(s) = \frac{\alpha}{s(s+1)(s+2)} \quad (6.180)$$

Given specific values $\alpha = 2$, $\beta = 3$ and $\gamma = -5$ and using the method of partial fraction expansion, we can write $Y(s)$ as:

$$\begin{aligned} Y(s) &= Y_p(s) + Y_h(s) = \left[\frac{2}{s(s+1)(s+2)} + \frac{3(s+3)}{(s+1)(s+2)} \right] - \frac{5}{(s+1)(s+2)} \\ &= \left[\frac{1}{s} - \frac{2}{s+1} + \frac{1}{s+2} \right] + \left[\frac{1}{s+1} + \frac{2}{s+2} \right] \end{aligned}$$

Taking the inverse transform on both sides we get the solution in time domain solution:

$$\begin{aligned} y_p(t) &= \mathcal{U}L^{-1}[Y_p(s)] = \mathcal{U}L^{-1}\left[\frac{1}{s} - \frac{2}{s+1} + \frac{1}{s+2}\right] = [1 - 2e^{-t} + e^{-2t}]u(t) \\ y_h(t) &= \mathcal{U}L^{-1}[Y_h(s)] = \mathcal{U}L^{-1}\left[\frac{1}{s+1} + \frac{2}{s+2}\right] = [e^{-t} + 2e^{-2t}]u(t) \end{aligned}$$

and

$$y(t) = y_h(t) + y_p(t) = [1 - e^{-t} + 3e^{-2t}]u(t) \quad (6.181)$$

If bilateral Laplace transform is applied to the same LCCDE, we get

$$s^2Y(s) + 3sY(s) + 2Y(s) = (s^2 + 3s + 2)Y(s) = \frac{\alpha}{s} = \frac{2}{s} \quad (6.182)$$

Solving this for $Y(s)$ and taking inverse transform, we get:

$$Y(s) = \frac{2}{s(s+1)(s+2)}, \quad y(t) = [e^{-t} + 2e^{-2t}]u(t) \quad (6.183)$$

This is the particular solution above with zero initial conditions. From this we see that bilateral Laplace transform can only solve an LCCDE system of zero initial conditions. When the initial conditions of the system are not all zero, unilateral Laplace transform has to be used.

6.2 The z-Transform

Similar to the Laplace transform, the z-transform is also a powerful tool widely used in many fields, especially in digital signal processing and discrete system analysis/design. Much of the discussion below is in parallel with that for the Laplace transform, with the only essential difference that all signals and systems considered here are discrete in time.

6.2.1 From Discrete Time Fourier Transform to z-Transform

The z-transform of a discrete signal $x[n]$ can be considered as the generalization of the discrete-time Fourier transform (DTFT) of the signal:

$$\mathcal{F}[x[n]] = \sum_{n=-\infty}^{\infty} x[n]e^{-jn\omega} = X(e^{j\omega}) \quad (6.184)$$

Here we adopt the notation $X(e^{j\omega})$ for the DTFT spectrum, instead of $X(f)$ or $X(\omega)$ used previously, for some reason which will become clear later. The transform above is based on the underlying assumption that the signal $x[n]$ is

square summable so that the summation converges and $X(e^{j\omega})$ exists. However, this assumption is not true for signals such as $x[n] = n$, $x[n] = n^2$, and $x[n] = e^{an}$, all of which are not square summable as they grow without a bound when $|n| \rightarrow \infty$. In such cases, we could still consider the Fourier transform of a modified version of the signal $x'[n] = x[n]e^{-\sigma n}$, where $e^{-\sigma n}$ is an exponential factor with a real parameter σ , which can force the given signal $x[n]$ to decay exponentially for some properly chosen value of σ (either positive or negative). For example, $x[n] = e^{an}u[n]$ ($a > 0$) does not converge when $n \rightarrow \infty$, therefore its Fourier spectrum does not exist. However, if we choose $\sigma > a$, the modified version $x'[n] = x[n]e^{-\sigma} = e^{-(\sigma-a)n}u[n]$ will converge as $n \rightarrow \infty$.

In general, the Fourier transform of the modified signal is:

$$\mathcal{F}[x'[n]] = \mathcal{F}[x[n]e^{-\sigma n}] = \sum_{n=-\infty}^{\infty} x[n]e^{-n(\sigma+j\omega)} = \sum_{n=-\infty}^{\infty} x[n]z^{-n} \quad (6.185)$$

where we have defined a complex variable

$$z = e^s = e^{\sigma+j\omega} = e^{\sigma}e^{j\omega} = |z|\angle z \quad (6.186)$$

which can be represented most conveniently in polar form in terms of its magnitude $|z| = e^{\sigma}$ and angle $\angle z = \omega$. If the summation above converges, it results in a complex function $X(z)$, which is called the *bilateral z-transform* of $x[n]$, formally defined as:

$$X(z) = \mathcal{Z}[x[n]] = \mathcal{F}[x[n]e^{-\sigma n}] = \sum_{n=-\infty}^{\infty} x[n]z^{-n} \quad (6.187)$$

Here $X(z)$ is a function defined over a 2-D complex z-plane typically represented in polar coordinates of $|z|$ and $\angle z$. Similar to the Laplace transform, here the z-transform $X(z)$ exists only inside the corresponding region of convergence (ROC) in the z-plane, composed of all z values that guarantee the convergence of the summation in Eq. 6.187. Due to the introduction of the exponential decay factor $e^{-\sigma n}$, we can properly choose the parameter σ so that the z-transform can be applied to a broader class of signals than the Fourier transform.

If the unit circle $|z| = e^{\sigma} = 1$ (when $\sigma = 0$ and $s = j\omega$) is inside the ROC, we can evaluate the 2-D function $X(z)$ along the unit circle with respect to $z = e^{j\omega}$ from $\omega = 0$ to $\omega = 2\pi$ to obtain the Fourier transform of $x[n]$. We see that the 1-D Fourier spectrum $X(e^{j\omega})$ of the discrete signal $x[n]$ is simply the cross section of the 2D function $X(z) = X(|z|e^{j\omega})$ along the unit circle $z = e^{j\omega}$, which is obviously periodic with period 2π . In other words, the discrete-time Fourier transform is just a special case of the z-transform when $\sigma = 0$ and $z = e^{j\omega}$:

$$\mathcal{F}[x[n]] = \mathcal{Z}[x[n]]|_{z=e^{j\omega}} = X(z)|_{z=e^{j\omega}} = X(e^{j\omega}) \quad (6.188)$$

This is the reason why sometimes the discrete-time Fourier spectrum is also denoted by $X(e^{j\omega})$.

Given the z-transform $X(z) = \mathcal{Z}[x[n]]$, the time signal $x[n]$ can be found by the inverse z-transform, which can be derived from the corresponding Fourier

transform of discrete signals:

$$\mathcal{Z}[x[n]] = X(z) = X(e^{\sigma+j\omega}) = \mathcal{F}[x[n]e^{-\sigma n}] \quad (6.189)$$

Taking the inverse Fourier transform of the above, we get

$$x[n]e^{-m\sigma} = \mathcal{F}^{-1}[X(e^{\sigma+j\omega})] = \frac{1}{2\pi} \int_0^{2\pi} X(e^{\sigma+j\omega}) e^{jn\omega} d\omega \quad (6.190)$$

Multiplying both sides by $e^{n\sigma}$, we get:

$$x[n] = \frac{1}{2\pi} \int_0^{2\pi} X(e^{\sigma+j\omega}) e^{(\sigma+j\omega)n} d\omega \quad (6.191)$$

To further represent the inverse z-transform in terms of z (instead of ω), we note

$$dz = d(e^{\sigma+j\omega}) = e^\sigma j e^{j\omega} d\omega = jz d\omega, \quad \text{i.e.,} \quad d\omega = z^{-1} dz / j \quad (6.192)$$

The integral of the inverse transform with respect to ω from 0 to 2π becomes an integral with respect to z along a circle of radius e^σ :

$$x[n] = \frac{1}{2\pi} \oint X(z) z^n z^{-1} dz / j = \frac{1}{2\pi j} \oint X(z) z^{n-1} dz \quad (6.193)$$

Now we get the forward and inverse z-transform pair:

$$\begin{aligned} X(z) &= \mathcal{Z}[x[n]] = \sum_{n=-\infty}^{\infty} x[n] z^{-n} \\ x[n] &= \mathcal{Z}^{-1}[X(z)] = \frac{1}{2\pi j} \oint X(z) z^{n-1} dz \end{aligned} \quad (6.194)$$

which can also be more concisely represented as

$$x[n] \xrightarrow{\mathcal{Z}} X(z) \quad (6.195)$$

In practice, we hardly need to carry out the integral in the inverse transform with respect to the complex variable z , as the z-transform pairs of most of the signals of interest can be obtained in some other ways and made available in table form.

As shown in Eq.6.186, the z-transform is related to the Laplace transform by an analytic function $z = e^s$ which maps a complex variable s in the s-plane to another complex variable z in the z-plane and vice versa. This function is called a *conformal mapping* as it preserves the angle formed by any two curves through each point in the complex plane. For example, a vertical line $Re[s] = \sigma_0$ in the s-plane is mapped to a circle $|z| = e^{\sigma_0}$ centered at the origin in the z-plane, a horizontal line $Im[s] = j\omega_0$ in the s-plane is mapped to a ray $\angle z = \omega_0$ in the z-plane from the origin in the direction determined by angle ω_0 , and the right angle formed by the pair of vertical and horizontal lines in the s-plane is mapped to the right angle formed by the circle and ray in the z-plane, i.e., the right angle is preserved by the mapping $z = e^s$.

The following three mapping pairs are of particular interest:

- The imaginary axis $\text{Re}[s] = \sigma = 0$ in the s-plane is mapped to the unit circle $|z| = e^\sigma = 1$ in the z-plane. In particular, the origin $s = \sigma + j\omega = 0$ of the s-plane is mapped to $z = e^s = e^0 = 1$ on the real axis in the z-plane;
- The vertical line corresponding to $\text{Re}[s] = \sigma = -\infty$ in the s-plane is mapped to the origin $|z| = e^\sigma = 0$ in the z-plane;
- The vertical line corresponding to $\text{Re}[s] = \sigma = \infty$ in the s-plane is mapped to a circle with infinite radius $|z| = e^\sigma = \infty$ in the z-plane.

Note that the continuous-time Fourier spectrum $X(j\omega) = \mathcal{F}[x(t)]$ is a non-periodic function defined over the entire imaginary axis $s = j\omega$ of the s-plane in the infinite range $-\infty < \omega < \infty$. But when the signal $x(t)$ is sampled to become a discrete signal $x[n]$, the corresponding discrete-time Fourier spectrum $X(e^{j\omega}) = \mathcal{F}[x[n]]$ becomes a periodic function over a finite range $0 \leq \omega < 2\pi$ around the unit circle $z = e^{j\omega}$ in the z-plane. These results are of course consistent with those obtained in the previous chapters.

In many applications the z-transform takes the form of a rational function as a ratio of two polynomials:

$$X(z) = \frac{\sum_{k=0}^M b_k z^k}{\sum_{k=0}^N a_k z^k} = \frac{b_M}{a_N} \frac{\prod_{k=1}^M (z - z_k)}{\prod_{k=1}^N (z - p_k)} \quad (6.196)$$

Here the roots z_k , ($k = 1, 2, \dots, m$) of the numerator polynomial of order M are the zeros of $X(z)$, and the roots p_k , ($k = 1, 2, \dots, n$) of the denominator polynomial of order N are the poles of $X(z)$. Some of these roots may be repeated. Moreover, if $N > M$, then $X(\infty) = 0$, i.e., $z = \infty$ is a zero. On the other hand, if $M > N$, then $X(\infty) = \infty$, i.e., $z = \infty$ is a pole. In general, we always assume $M < N$, as otherwise we can carry out a long division to expand $X(z)$ into multiple terms so that $M < N$ is true for each fraction. The locations of the zeros and poles of $X(z)$ characterize some essential properties of a signal $x[n]$.

6.2.2 Region of Convergence

Same as in the Laplace transform, the region of convergence plays an important role in the z-transform. Here we consider z-transform of a set of signals which are in parallel with those in Example 6.1 of the Laplace transform:

Example 6.5:

1. A right-sided discrete signal $x[n] = a^{-n}u[n]$:

$$X(z) = \sum_{n=-\infty}^{\infty} x[n]z^{-n} = \sum_{n=0}^{\infty} (az)^{-n} \quad (6.197)$$

where a is a real constant. This summation is a geometric series which does not converge unless $|(az)^{-1}| < 1$, i.e., the region of convergence (ROC) can

be specified as $|z| > 1/|a|$, which is the entire region outside the circle with radius $|z| = 1/|a|$. Now the z-transform above can be further written as:

$$X(z) = \sum_{n=0}^{\infty} (az^{-1})^n = \frac{1}{1 - (az)^{-1}}, \quad \text{if } |z| > 1/|a| \quad (6.198)$$

Specially when $a = 1$, we have $x[n] = u[n]$ and

$$U(z) = \mathcal{Z}[u[n]] = \frac{1}{1 - z^{-1}}, \quad \text{if } |z| > 1 \quad (6.199)$$

If we let $\text{Re}[s] = \sigma \rightarrow 0$, i.e., $|z| = 1$, $U(z)$ will be evaluated along the unit circle $z = e^{j\omega}$ and become $\mathcal{Z}[u[n]] = 1/(1 - e^{-j\omega})$, which is seemingly the Fourier spectrum of $u[n]$. However this result is actually invalid, as $|z| = 1$ is not inside the ROC $|z| > 1$. Comparing this result with the real Fourier transform of $u[n]$ in Eq.4.24:

$$\mathcal{F}[u[n]] = \frac{1}{1 - e^{-j2\pi f}} + \frac{1}{2} \sum_{k=-\infty}^{\infty} \delta(f - k) \quad (6.200)$$

we see that an extra term $\sum_{k=-\infty}^{\infty} \delta(f - k)/2$ in the Fourier spectrum which reflects the fact that the summation is only marginally convergent when $|z| = 1$.

2. A left-sided signal $x[n] = -a^{-n}u[-n - 1]$:

$$X(z) = - \sum_{n=-\infty}^{\infty} a^{-n}u[-n - 1]z^{-n} = - \sum_{n=-\infty}^{-1} (az)^{-m} = 1 - \sum_{n=0}^{\infty} (az)^n \quad (6.201)$$

We see that only when $|az| < 1$, i.e., z is inside the ROC $|z| < 1/|a|$, will this summation converge and $X(z)$ exist:

$$X(z) = 1 - \frac{1}{1 - az} = \frac{1}{1 - (az)^{-1}}, \quad \text{if } |z| < 1/|a| \quad (6.202)$$

Based on the examples above we summarize a set of properties of the ROC:

- If a signal $x[n]$ of finite duration is absolutely summable then its transform $X(z)$ exists for any z , i.e., its ROC is the entire z -plane.
- The ROC does not contain any poles because by definition $X(z)$ does not exist at any pole.
- Two different signals may have identical transform but different ROCs. The inverse transform can be carried out only if an associated ROC is also specified.
- Only the magnitude $|z| = e^{\sigma}$ of z determines the convergence of the summation in the z-transform and thereby the ROC. The angle $\angle z$ has no effect on the convergence. Consequently the ROC is always bounded by two concentric circles centered at the origin corresponding to two poles p_1 and p_2 with $|p_1| < |p_2|$. It is possible that $|p_1| = 0$ and/or $|p_2| = \infty$.

- The ROC of a right-sided signal is outside the outermost pole; The ROC of a left-sided signal is inside the innermost pole. If a signal is two-sided, its ROC is the intersection of the two ROCs corresponding to its two one-sided parts, which can be either a ring between two circles or an empty set.
- The Fourier transform $X(e^{j\omega})$ of a signal $x[n]$ exists if the ROC of the corresponding z-transform $X(z)$ contains the unit circle $|z| = 1$, i.e., $z = e^{j\omega}$.

The zeros and poles of $X(z) = \mathcal{Z}[x[n]]$ dictate the ROC and thereby the most essential properties of the corresponding signal $x[n]$, such as whether it is right or left-sided, whether it grows or decays over time.

Example 6.6: Find the time signal corresponding to the following z-transform:

$$X(z) = \frac{1}{(1 - \frac{1}{3}z^{-1})(1 - 2z^{-1})} = -\frac{1/5}{1 - \frac{1}{3}z^{-1}} + \frac{6/5}{1 - 2z^{-1}} \quad (6.203)$$

This function has two poles: $p_1 = 1/3$ and $p_2 = 2$. Now consider three possible ROCs corresponding to three different time signals:

- $|z| > 2$: The ROC is outside the outermost pole $p_2 = 2$, both terms of $X(z)$ correspond to right-sided time functions:

$$x[n] = -\frac{1}{5}(\frac{1}{3})^n u[n] + \frac{1}{5}(\frac{1}{3})^n u[n] \quad (6.204)$$

- $|z| < 1/3$: The ROC is inside the innermost pole $p_1 = 1/3$, both terms of $X(z)$ correspond to left-sided time functions:

$$x[n] = \frac{1}{5}(\frac{1}{3})^n u[-n-1] - \frac{1}{5}(\frac{1}{3})^n u[-n-1] \quad (6.205)$$

- $1/3 < |z| < 2$: The ROC is a ring between the two poles, the two terms correspond to two different types of functions, one right-sided while the other left-sided:

$$x[n] = -\frac{1}{5}(\frac{1}{3})^n u[n] - \frac{1}{5}(\frac{1}{3})^n u[-n-1] \quad (6.206)$$

In particular, note that only the last ROC includes the circle $|z| = 1$ and the corresponding time function $x[n]$ has a discrete Fourier transform. Fourier transform of the other two functions do not exist.

6.2.3 Properties of the z-Transform

The z-transform has a set of properties many of which are in parallel with those of the discrete-time Fourier transform. The proofs of such properties are therefore omitted as they are similar to that of their counterparts in the Fourier transform. However, here we need to pay special attention to the ROCs. In the following,

we always assume:

$$\mathcal{Z}[x[n]] = X(z), \quad \mathcal{Z}[y[n]] = Y(z) \quad (6.207)$$

with R_x and R_y as their corresponding ROCs. If a property can be easily derived from the definition, the proof is not provided.

- **Linearity**

$$\mathcal{Z}[ax[n] + by[n]] = aX(z) + bY(z) \quad ROC \supseteq (R_x \cap R_y) \quad (6.208)$$

Similar to the case of the Laplace transform, the ROC of the linear combination of $x[n]$ and $y[n]$ may be larger than the intersection of their individual ROCs $R_x \cap R_y$, due to reasons such as zero-pole cancellation.

- **Time-shift**

$$\mathcal{Z}[x[n - n_0]] = z^{-n_0} X(z), \quad ROC = R_x \quad (6.209)$$

Time delay is a very important and useful operation that delays a signal $x[n]$ by one time unit to become $x[n - 1]$. This operation is easily realized in z-domain by a multiplication with z^{-1} , which can be readily used as a delay unit.

- **Time reversal**

$$\mathcal{Z}[x[-n]] = X(z^{-1}), \quad ROC = 1/R_x \quad (6.210)$$

Proof:

$$\mathcal{Z}[x[-n]] = \sum_{n=-\infty}^{\infty} x[-n]z^{-n} = \sum_{n'=-\infty}^{\infty} x[n'](z^{-1})^{-n'} = X(z^{-1}) \quad (6.211)$$

- **Modulation**

$$\mathcal{Z}[(-1)^n x[n]] = X(-z) \quad (6.212)$$

Here modulation means every other sample of the signal is negated.

Proof:

$$\mathcal{Z}[(-1)^n x[n]] = \sum_{n=-\infty}^{\infty} x[n](-1)^n z^{-n} = \sum_{n=-\infty}^{\infty} x[n](-z)^{-n} = X(-z) \quad (6.213)$$

- **Down-sampling**

$$\mathcal{Z}[x_{(2)}[n]] = \frac{1}{2}[X(z^{1/2}) + X(-z^{1/2})] \quad (6.214)$$

Here the down-sampled version $x_{(2)}[n]$ of a signal $x[n]$ is composed of all the even terms of the signal with all odd terms dropped, i.e., $x_{(2)}[n] = x[2n]$.

Proof:

$$\begin{aligned}
 \mathcal{Z}[x_{(2)}[n]] &= \sum_{n=-\infty}^{\infty} x[2n]z^{-n} = \sum_{m=\dots,-2,0,2,\dots} x[m](z^{1/2})^{-m} \\
 &= \frac{1}{2} \left[\sum_{m=-\infty}^{\infty} x[m](z^{1/2})^{-m} + \sum_{m=-\infty}^{\infty} x[m](-z^{1/2})^{-m} \right] \\
 &= \frac{1}{2} [X(z^{1/2}) + X(-z^{1/2})]
 \end{aligned} \tag{6.215}$$

where we have assumed $m = 2n$. The third equal sign is due to the fact that the two terms are the same when m is even but their sum is zero when m is odd.

- **Up-sampling**

$$\mathcal{Z}[x^{(k)}[n]] = X(z^k) \tag{6.216}$$

Here $x^{(k)}[n]$ is defined as:

$$x^{(k)}[n] = \begin{cases} x[n/k] & \text{if } n \text{ is a multiple of } k \\ 0 & \text{else} \end{cases} \tag{6.217}$$

i.e. $x^{(k)}[n]$ is obtained by inserting $k - 1$ zeros between every two consecutive samples of $x[n]$.

Proof:

$$\mathcal{Z}[x^{(k)}[n]] = \sum_{n=-\infty}^{\infty} x[n/k]z^{-n} = \sum_{m=-\infty}^{\infty} x[m]z^{-km} = X(z^k) \tag{6.218}$$

Note that the change of the summation index from n to $m = n/k$ has no effect as the terms skipped are all zeros.

Combining the down and up-sampling above, we see that if a signal $x[n]$ with $X(z) = \mathcal{Z}[x[n]]$ is first down-sampled and then up-sampled, its z-transform is:

$$\mathcal{Z}[(x_{(2)})^{(2)}[n]] = \frac{1}{2} [X(z) + X(-z)] \tag{6.219}$$

However, also note that if the signal is first up and then down-sampled, it remains the same: $(x^{(2)})_{(2)}[n] = x[n]$.

- **Convolution**

$$\mathcal{Z}[x[n] * y[n]] = X(z)Y(z), \quad ROC \supseteq (R_x \cap R_y) \tag{6.220}$$

The ROC of the convolution could be larger than the intersection of R_x and R_y , due to the possible pole-zero cancellation caused by the convolution.

- **Autocorrelation**

$$\mathcal{Z} \left[\sum_k x[k]x[k-n] \right] = X(z)X(z^{-1}) \tag{6.221}$$

Proof:

The autocorrelation of a signal $x[n]$ is the convolution of the signal with its time reversed version. Applying the properties of time reversal and convolution, the above can be proven.

- **Time difference**

$$\mathcal{Z}[x[n] - x[n-1]] = (1 - z^{-1})X(z), \quad ROC = R_x \quad (6.222)$$

Proof:

$$\mathcal{Z}[x[n] - x[n-1]] = X(z) - z^{-1}X(z) = (1 - z^{-1})X(z) \quad (6.223)$$

Note that due to the additional zero $z = 1$ and pole $z = 0$, the resulting ROC is the same as R_x except the possible deletion of $z = 0$ caused by the added pole and/or addition of $z = 1$ caused by the added zero which may cancel an existing pole.

- **Time accumulation**

$$\mathcal{Z}\left[\sum_{k=-\infty}^n x[k]\right] = \frac{1}{1 - z^{-1}}X(z) \quad (6.224)$$

Proof: First we realize that the accumulation of $x[n]$ can be written as its convolution with $u[n]$:

$$u[n] * x[n] = \sum_{k=-\infty}^{\infty} u[n-k]x[k] = \sum_{k=-\infty}^n x[k] \quad (6.225)$$

Applying the convolution property, we get

$$\mathcal{Z}\left[\sum_{k=-\infty}^n x[k]\right] = \mathcal{Z}[u[n] * x[n]] = \frac{1}{1 - z^{-1}}X(z) \quad (6.226)$$

as $\mathcal{Z}[u[n]] = 1/(1 - z^{-1})$.

- **Scaling in Z-domain**

$$\mathcal{Z}[z_0^n x[n]] = X\left(\frac{z}{z_0}\right), \quad ROC = |z_0|R_x \quad (6.227)$$

Proof:

$$\mathcal{Z}[z_0^n x[n]] = \sum_{n=-\infty}^{\infty} x[n] \left(\frac{z}{z_0}\right)^{-1} = X\left(\frac{z}{z_0}\right) \quad (6.228)$$

In particular, if $z_0 = e^{j\omega_0}$, the above becomes

$$\mathcal{Z}[e^{jn\omega_0} x[n]] = X(e^{-j\omega_0} z), \quad ROC = R_x \quad (6.229)$$

The multiplication by $e^{-j\omega_0}$ to z corresponds to a rotation by angle ω_0 in the z-plane, i.e., a frequency shift by ω_0 . The rotation is either clockwise ($\omega_0 > 0$) or counter clockwise ($\omega_0 < 0$) corresponding to, respectively, either a left-shift or a right shift in s-domain. The property is essentially the same as the frequency shifting property of discrete Fourier transform.

- **Conjugation**

$$\mathcal{Z}[x^*[n]] = X^*(z^*), \quad ROC = R_x \quad (6.230)$$

Proof: Complex conjugate of the z-transform of $x[n]$ is

$$X^*(z) = \left[\sum_{n=-\infty}^{\infty} x[n]z^{-n} \right]^* = \sum_{n=-\infty}^{\infty} x^*[n](z^*)^{-n} \quad (6.231)$$

Replacing z by z^* , we get the desired result.

- **Differentiation in z-Domain**

$$\mathcal{Z}[nx[n]] = -\frac{d}{dz}X(z), \quad ROC = R_x \quad (6.232)$$

Proof:

$$\frac{d}{dz}X(z) = \sum_{n=-\infty}^{\infty} x[n]\frac{d}{dz}(z^{-n}) = \sum_{n=-\infty}^{\infty} (-n)x[n]z^{-n-1} \quad (6.233)$$

i.e.,

$$\mathcal{Z}[nx[n]] = -z\frac{d}{dz}X(z) \quad (6.234)$$

Example 6.7: Consider the following examples:

- The z-transform of a modulated, time-reversed and shifted signal $(-1)^n x[k-n]$ is

$$\begin{aligned} \mathcal{Z}[(-1)^n x[k-n]] &= \sum_{n=-\infty}^{\infty} (-1)^n x[k-n]z^{-n} = \sum_{n=-\infty}^{\infty} x[k-n](-z)^{-n} \\ &= \sum_{m=-\infty}^{\infty} x[m](-z)^{m-k} = (-z)^{-k} \sum_{m=-\infty}^{\infty} x[m](-z^{-1})^{-m} = (-z)^{-k} X(-z^{-1}) \end{aligned} \quad (6.235)$$

where $m = k - n$.

- The z-transform of a signal $x[n]$ first down-sampled then up-sampled is

$$X'(z) = \frac{1}{2}[X(z) + X(-z)] \quad (6.236)$$

which can be obtained by applying the properties of down-sampling and up-sampling in Eqs.6.214 and 6.216. To verify this result, we apply the property of modulation in Eq.6.212 to the second term and get:

$$\begin{aligned} x'[n] &= \mathcal{Z}^{-1}[X'(z)] = \frac{1}{2}[\mathcal{Z}^{-1}[X(z)] + \mathcal{Z}^{-1}[X(-z)]] \\ &= \frac{1}{2}[x[n] + (-1)^n x[n]] = \begin{cases} x[n] & \text{even } n \\ 0 & \text{odd } n \end{cases} \end{aligned} \quad (6.237)$$

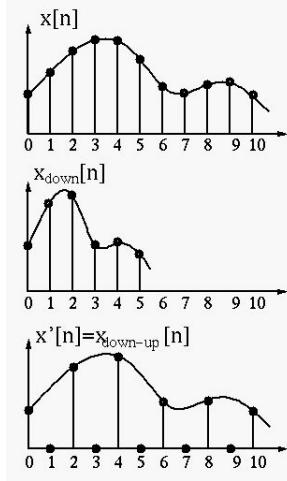


Figure 6.14 Down and up sampling

- Taking derivative of the right side of

$$\mathcal{Z}[a^n u[n]] = \frac{1}{1 - az^{-1}}, \quad |z| > a \quad (6.238)$$

we get

$$\frac{d}{dz} \left[\frac{1}{1 - az^{-1}} \right] = \frac{-az^{-2}}{(1 - az^{-1})^2} \quad (6.239)$$

Due to the property of differentiation in z-domain, we have

$$\mathcal{Z}[na^n u[n]] = \frac{az^{-1}}{1 - az^{-1}}, \quad |z| > a \quad (6.240)$$

Note that for a different ROC $|z| < a$, we have

$$\mathcal{Z}[-na^n u[-n - 1]] = \frac{az^{-1}}{1 - az^{-1}}, \quad |z| < a \quad (6.241)$$

6.2.4 z-Transform of Typical Signals

- $\delta[n]$, $\delta[n - m]$

$$\mathcal{Z}[\delta[n]] = \sum_{n=-\infty}^{\infty} \delta[n]z^{-n} = 1, \quad \text{for all } z \quad (6.242)$$

Due to the time-shift property, we also have

$$\mathcal{Z}[\delta[n - m]] = z^{-m}, \quad \text{for all } z \quad (6.243)$$

- $u[n]$, $a^n u[n]$, $na^n u[n]$

$$\mathcal{Z}[u[n]] = \sum_{n=0}^{\infty} z^{-n} = \frac{1}{1-z^{-1}}, \quad |z| > 1 \quad (6.244)$$

Due to the scaling in z-domain property, we have

$$\mathcal{Z}[a^n u[n]] = \frac{1}{1-(z/a)^{-1}} = \frac{1}{1-az^{-1}}, \quad |z| > a \quad (6.245)$$

Applying the property of differentiation in z-Domain to the above, we have

$$\mathcal{Z}[n a^n u[n]] = -z \frac{d}{dz} \left[\frac{1}{1-az^{-1}} \right] = -z \frac{-az^{-2}}{(1-az^{-1})^2} = \frac{az^{-1}}{(1-az^{-1})^2}, \quad |z| > a \quad (6.246)$$

- $e^{\pm j n \omega_0} u[n], \cos[n \omega_0] u[n], \sin[n \omega_0] u[n]$

Applying the scaling in z-domain property to $\mathcal{Z}[u[n]] = 1/(1-z^{-1})$, we have

$$\mathcal{Z}[e^{j m \omega_0} u[n]] = \frac{1}{1-(e^{j \omega_0} z)^{-1}} = \frac{1}{1-e^{-j \omega_0} z^{-1}}, \quad |z| > 1 \quad (6.247)$$

and similarly, we have

$$\mathcal{Z}[e^{-j m \omega_0} u[n]] = \frac{1}{1-e^{j \omega_0} z^{-1}}, \quad |z| > 1 \quad (6.248)$$

Moreover, we have

$$\begin{aligned} \mathcal{Z}[\cos(n \omega_0) u[n]] &= \mathcal{Z}\left[\frac{e^{j n \omega_0} + e^{-j n \omega_0}}{2} u[n]\right] \\ &= \frac{1}{2} \left[\frac{1}{1-e^{j \omega_0} z^{-1}} + \frac{1}{1-e^{-j \omega_0} z^{-1}} \right] = \frac{2 - (e^{j \omega_0} + e^{-j \omega_0}) z^{-1}}{2[1-(e^{j \omega_0} + e^{-j \omega_0}) z^{-1} + z^{-2}]} \\ &= \frac{1 - \cos \omega_0 z^{-1}}{1 - 2 \cos \omega_0 z^{-1} + z^{-2}} \quad |z| > 1 \end{aligned} \quad (6.249)$$

Similarly we have

$$\mathcal{Z}[\sin(n \omega_0) u[n]] = \frac{\sin \omega_0 z^{-1}}{1 - 2 \cos \omega_0 z^{-1} + z^{-2}}, \quad |z| > 1 \quad (6.250)$$

- $r^n \cos(n \omega_0) u[n], r^n \sin(n \omega_0) u[n]$

Applying the z-domain scaling property to the above, we have

$$\mathcal{Z}[r^n \cos(n \omega_0) u[n]] = \frac{1 - r \cos \omega_0 z^{-1}}{1 - 2r \cos \omega_0 z^{-1} + r^2 z^{-2}}, \quad |z| > r \quad (6.251)$$

and

$$\mathcal{Z}[r^n \sin(n \omega_0) u[n]] = \frac{r \sin \omega_0 z^{-1}}{1 - 2r \cos \omega_0 z^{-1} + r^2 z^{-2}}, \quad |z| > r \quad (6.252)$$

6.2.5 Analysis of Discrete LTI Systems by z-Transform

The z-transform is a convenient tool for the analysis and design of discrete LTI systems $y[n] = \mathcal{O}[x[n]]$ whose output $y[n]$ is the convolution of the input $x[n]$

and its impulse response function $h[n]$:

$$y[n] = \mathcal{O}[x[n]] = h[n] * x[n] = \sum_{m=-\infty}^{\infty} h[m]x[n-m] \quad (6.253)$$

In particular, if the input is an impulse $x[n] = \delta[n]$, then the output is the impulse response function $y[n] = \mathcal{O}[\delta[n]] = h[n] * \delta[n] = h[n]$. Also if the input is a complex exponential $x[n] = e^{sn} = z^n$ ($z = e^s$), then the output is:

$$y[n] = \mathcal{O}[z^n] = \sum_{m=-\infty}^{\infty} h[m]z^{n-m} = z^n \sum_{m=-\infty}^{\infty} h[m]z^{-m} = H(z)z^n \quad (6.254)$$

where $H(z)$ is the *transfer function* of the discrete system, first defined in Eq.1.110 in Chapter 1, which is actually the z-transform of the impulse response $h[n]$ of the system:

$$H(z) = \mathcal{Z}[h[n]] = \sum_{n=-\infty}^{\infty} h[n]z^{-n} \quad (6.255)$$

Eq.6.254 is the eigenequation of *any* discrete LTI system, where the transfer function $H(z)$ is the eigenvalue, and the complex exponential input $x[n] = e^{sn} = z^n$ is the corresponding eigenfunction. In particular, if we let $\sigma = 0$, i.e., $z = e^{j\omega}$, then the transfer function $H(z)$ becomes the discrete-time Fourier transform of the impulse response $h[n]$ of the system:

$$H(z)|_{s=j\omega} = H(e^{j\omega}) = \sum_{n=-\infty}^{\infty} h[n]e^{-j\omega n} = \mathcal{F}[h[n]] \quad (6.256)$$

This is the frequency response function of the discrete LTI system first defined in Eq.5.10 of Chapter 3. Various properties and behaviors such as the stability and filtering effects of a discrete LTI system can be qualitatively characterized based on the locations of the zeros and poles of the zeros and poles of its transfer function $H(z) = \mathcal{Z}[h[n]]$ due to the properties of the ROC of the z-transform.

Also, due to its convolution property of the z-transform, the convolution in Eq.6.253 can be converted to a multiplication in z-domain:

$$y[n] = h[n] * x[n] \xrightarrow{\mathcal{Z}} Y(z) = H(z)X(z) \quad (6.257)$$

Based on this relationship the transfer function $H(z)$ can also be found in z-domain as the ratio $H(z) = Y(z)/X(z)$ of the output $Y(z)$ and input $X(z)$. The ROC and poles of the transfer function $H(s)$ of an LTI system dictate the behaviors of system, such as its causality and stability.

- **Stability**

Also as discussed in Chapter 1, a discrete LTI system is stable if to any bounded input $|x[n]| < B$ its response $y[n]$ is also bounded for all n , and its

impulse response function $h[n]$ needs to be absolutely summable (Eq.1.120):

$$\sum_{n=-\infty}^{\infty} |h[n]| < \infty \quad (6.258)$$

i.e., the frequency response function $\mathcal{F}[h[n]] = H(e^{j\omega}) = H(z)|_{z=e^{j\omega}}$ exists. In other words, an LTI system is stable if and only if the ROC of its transfer function $H(z)$ includes the unit circle $|z| = 1$.

- **Causality**

A discrete LTI system is causal if its impulse response $h[n]$ is a consequence of the impulse input $\delta[n]$, i.e., $h[n]$ comes after $\delta[n]$:

$$h[n] = h[n]u[n] = \begin{cases} h[n] & n \geq 0 \\ 0 & n < 0 \end{cases} \quad (6.259)$$

and its output is (Eq.1.121):

$$y[n] = \sum_{m=-\infty}^{\infty} h[m]x[n-m] = \sum_{m=0}^{\infty} h[m]x[n-m] \quad (6.260)$$

The ROC of $H(z)$ is the exterior of a circle. In particular, when $H(z)$ is rational, the system is causal if and only if its ROC is the exterior of a circle outside the outermost pole, and the order of numerator is no greater than that of the denominator so that $z = \infty$ is not a pole ($H(\infty)$ exists).

Combining the two properties above, we see that a causal LTI system with a rational transfer function $H(z)$ is stable if and only if all poles of $H(z)$ are inside the unit circle of the z-plane, i.e., the magnitudes of all poles are smaller than 1: $|p_k| < 1$.

One type of discrete LTI system can be characterized by a linear constant-coefficient difference equation (LCCDE):

$$\sum_{k=0}^N a_k y[n-k] = \sum_{k=0}^M b_k x[n-k] \quad (6.261)$$

Taking the z-transform of this equation, we get an algebraic equation in z domain:

$$Y(z) \left[\sum_{k=0}^N a_k z^{-k} \right] = X(z) \left[\sum_{k=0}^M b_k z^{-k} \right] \quad (6.262)$$

The transfer function of such a system is rational:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^M b_k z^{-k}}{\sum_{k=0}^N a_k z^{-k}} = c \frac{\prod_{k=1}^M (z - z_{0_k})}{\prod_{k=1}^N (z - z_{0_k})} \quad (6.263)$$

where z_k , ($k = 1, 2, \dots, M$) and p_k , ($k = 1, 2, \dots, N$) are the zeros and poles of $H(z)$, respectively. For simplicity and without loss of generality, we will assume $N > M$ and $c = 1$ below.

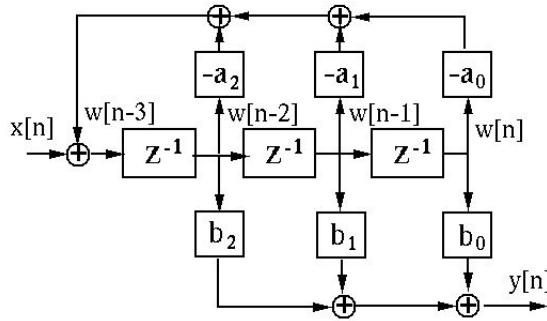


Figure 6.15 Block diagram of a discrete LTI system described by a LCCDE

The output $Y(z)$ of the LTI system can be represented as

$$Y(z) = H(z)X(z) = \left(\sum_{k=0}^M b_k z^{-k} \right) \frac{1}{\sum_{k=0}^N a_k z^{-k}} X(z) = \left(\sum_{k=0}^M b_k z^{-k} \right) W(z) \quad (6.264)$$

or in time domain:

$$y[n] = \sum_{k=0}^M b_k w[n-k] \quad (6.265)$$

where we have defined $W(z) = X(z)/(\sum_{k=0}^N a_k z^{-k})$ as an intermediate variable, or in time domain:

$$\sum_{k=0}^N a_k w[n-k] = x[n], \quad \text{or} \quad a_N w[n-N] = x[n] - \sum_{k=0}^{N-1} a_k w[n-k] \quad (6.266)$$

Without loss of generality, we assume $a_N = 1$, and the LTI system can now be represented as a *block diagram* as shown in Fig.6.15 (for $M = 2$ and $N = 3$).

To find the impulse response $h[n]$ we first convert $H(z)$ to a summation by partial fraction expansion:

$$H(z) = \frac{\prod_{k=1}^M (z - z_{0k})}{\prod_{k=1}^N (z - z_{0k})} = \sum_{k=1}^N \frac{c_k}{1 - p_k z^{-1}} \quad (6.267)$$

(assume no repeated poles) and then carry out the inverse transform (the LTI system in Eq.6.261 is causal) to get:

$$h[n] = \mathcal{Z}^{-1}[H(z)] = \sum_{k=1}^N \mathcal{Z}^{-1} \left[\frac{c_k}{1 - p_k z^{-1}} \right] = \sum_{k=1}^N c_k p_k^n u[n] \quad (6.268)$$

The output $y[n]$ of the LTI system can be found by solving the difference equation in Eq.6.261. Alternatively, it can also be found by the convolution $y[n] = h[n] * x[n]$, or the inverse z-transform:

$$y[n] = \mathcal{Z}^{-1}[Y(z)] = \mathcal{Z}^{-1}[H(z)X(z)] \quad (6.269)$$

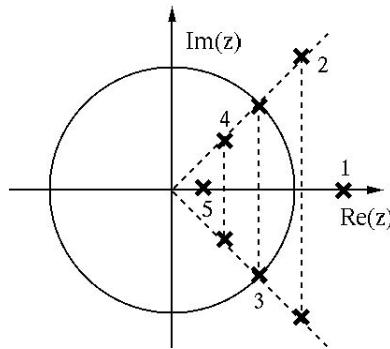


Figure 6.16 Different pole locations of $H(z)$

As the LCCDE in Eq.6.261 is an LTI system, it can also be solved in the following two steps. First, we assume the input on the right-hand is simply $x[n]$ and find the corresponding output $y[n]$. Then the response to the true input $\sum_k b_k x[n - k]$ can be found to be $\sum_k b_k y[n - k]$.

Note that the output $y[n]$ obtained this way is only the particular solution due to input $x[n]$, but the homogeneous solution due to any non-zero initial conditions is not represented by the bilateral Laplace transform. This problem will be addressed by the unilateral z-transform to be discussed later, which takes the initial conditions into consideration.

Same as in the case of a continuous LTI system, here the behavior of a discrete LTI system in terms of stability and oscillation is also dictated by the pole locations in the z-plane. The poles are either real or form complex conjugate pairs, either inside or outside the unit circle, as shown in Fig. 6.16, where the numbered pole locations correspond to those in the s-plane for the continuous case as shown in Fig.6.2 with similar waveforms in time domain.

A discrete LTI system can be treated as a filter, called a *digital filter*. Depending on the specific form of the LCCDE (Eq.6.261) that describes the filter, it belongs to either of the following two types.

- *Finite impulse response (FIR) filters:*

In Eq.6.261, if specially $a_0 = 1$ and $a_k = 0$ for all $k > 0$, then the impulse response of the system becomes:

$$h[n] = \sum_{k=0}^M b_k \delta[n - k], \quad (n = 0, \dots, M) \quad (6.270)$$

As $h[n]$ has only a finite number of non-zero terms, it is absolutely summable, and the transfer function

$$H(z) = \sum_{n=0}^M h[n] = \sum_{n=0}^M b_n z^{-n} \quad (6.271)$$

does not have any poles, i.e., an FIR filter is always stable. In particular, if $b_k = 1/(M+1)$, this system becomes a discrete moving average filter, i.e., the output $y[n]$ is the average of the last $M+1$ inputs.

- *Infinite impulse response (IIR) filters:*

Any LTI system described by Eq.6.261 without the special condition $a_k = 0$ ($k > 0$) is an IIR filter as there are in general an infinite number of terms in its impulse response in Eq.6.268. As discussed previously, an IIR filter is stable if all of its poles are inside the unit circle. For example, consider this simple LTI system:

$$y[n] - ay[n-1] = x[n] \quad (6.272)$$

with impulse response $h[n] = a^n u[n]$. This system is stable only if $|a| < 1$ and its transfer function is $H(z) = 1/(1 - az^{-1})$. As the impulse response $h[n]$ has infinite non-zero terms ($n = 0, 1, \dots$), this is an IIR filter.

Example 6.8: The input and output of an LTI system are related by

$$y[n] - \frac{1}{2}y[n-1] = x[n] + \frac{1}{3}x[n-1] \quad (6.273)$$

Note that without further information such as the initial condition, this equation does not uniquely specify $y[n]$ when $x[n]$ is given. Taking z-transform of this equation and using the time-shift property, we get

$$Y(z) - \frac{1}{2}z^{-1}Y(z) = X(z) + \frac{1}{3}z^{-1}X(z) \quad (6.274)$$

and the transfer function can be obtained

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 + \frac{1}{3}z^{-1}}{1 - \frac{1}{2}z^{-1}} = \frac{1}{1 - \frac{1}{2}z^{-1}}(1 + \frac{1}{3}z^{-1}) \quad (6.275)$$

Note that the causality and stability of the system is not provided by this equation, unless the ROC of this $H(z)$ is specified. Consider these two possible ROCs:

- If ROC is $|z| > 1/2$, it is outside the pole $z_p = 1/2$ and includes the unit circle. The system is causal and stable:

$$h[n] = (\frac{1}{2})^n u[n] + \frac{1}{3}(\frac{1}{2})^{n-1} u[n-1] \quad (6.276)$$

- If ROC is $|z| < 1/2$, it is inside the pole $z_p = 1/2$ and does not include the unit circle. The system is anti-causal and unstable:

$$h[n] = -(\frac{1}{2})^n u[-n-1] - \frac{1}{3}(\frac{1}{2})^{n-1} u[-n] \quad (6.277)$$

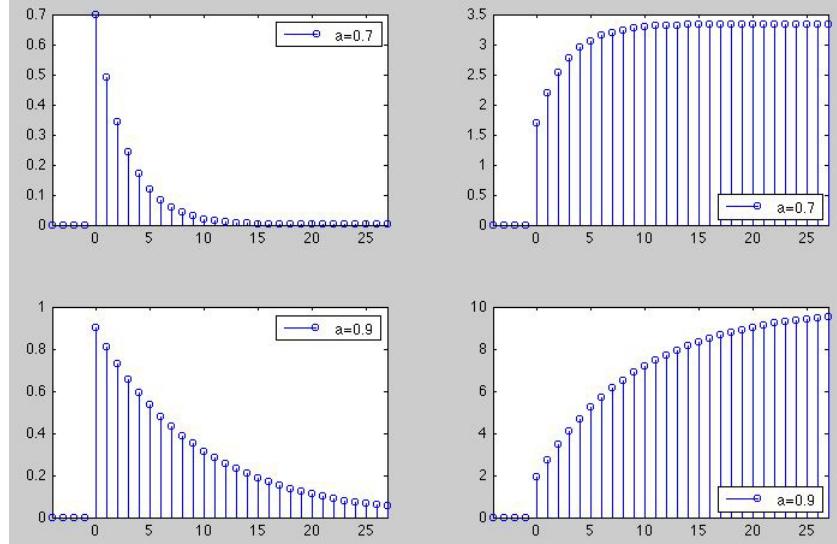


Figure 6.17 Impulse and Step Responses of 1st Order System

The impulse responses (left) and step responses (right) for $a = 0.7$ (top) and $a = 0.9$ (bottom).

6.2.6 First and Second Order Systems

As discussed previously in Example 5.2, a first order causal system is described by the following difference equation:

$$y[n] - ay[n-1] = x[n] \quad (6.278)$$

Its impulse response is $h[n] = a^n u[n]$ with $|a| < 1$ for the system to be stable. The transfer function of the system is:

$$H(z) = \mathcal{Z}[h[n]] = \sum_{n=0}^{\infty} a^n z^{-n} = \frac{1}{1 - az^{-1}} = \frac{z}{z - a} \quad (6.279)$$

Like τ for a continuous first order system, here the pole a is the only parameter needed to characterize a first order discrete system. Also as shown in Example 5.2, the system's step response is:

$$y[n] = h[n] * u[n] = \frac{1 - a^{n+1}}{1 - a} u[n] \quad (6.280)$$

The impulse and step responses of the first-order system are shown in Fig.6.17.

The canonical form of the difference equation for a second order system is

$$y[n] - 2r \cos \theta y[n-1] + r^2 = x[n] \quad (6.281)$$

Like ζ and ω_n for a continuous second order system, here r and θ are the two parameters needed to characterize a second order discrete system.

Taking the z-transform on both sides we get:

$$(1 - 2r \cos \theta z^{-1} + r^2 z^{-2})Y(z) = X(z) \quad (6.282)$$

and the transfer function is:

$$\begin{aligned} H(z) &= \frac{Y(z)}{X(z)} = \frac{1}{1 - 2r \cos \theta z^{-1} + r^2 z^{-2}} = \frac{1}{(1 - p_1 z^{-1})(1 - p_2 z^{-1})} \\ &= \frac{1}{(1 - re^{j\theta} z^{-1})(1 - re^{-j\theta} z^{-1})} = \frac{z^2}{(z - re^{j\theta})(z - re^{-j\theta})} \end{aligned} \quad (6.283)$$

where p_1 and p_2 are the two poles, the two solutions of the quadratic equation $z^2 - 2r \cos \theta z + r^2 = 0$:

$$p_{1,2} = r \cos \theta \pm jr \sin \theta = r(\cos \theta \pm j \sin \theta) = re^{\pm j\theta} \quad (6.284)$$

When θ is not 0 or π therefore $e^{j\theta} \neq e^{-j\theta}$, the two poles are different. We see that for the system to be stable, we must have $|p_{1,2}| = |r| < 1$ for both poles to be inside the unit circle.

To find the impulse response of the system by inverse z-transform, we first carry out partial fraction expansion:

$$H(z) = \frac{1}{(1 - re^{j\theta} z^{-1})(1 - re^{-j\theta} z^{-1})} = \frac{A}{1 - re^{j\theta} z^{-1}} + \frac{B}{1 - re^{-j\theta} z^{-1}} \quad (6.285)$$

to find

$$A = \frac{e^{j\theta}}{2j \sin \theta}, \quad B = \frac{-e^{-j\theta}}{2j \sin \theta} \quad (6.286)$$

Now we can get the impulse response:

$$\begin{aligned} h[n] &= \mathcal{Z}^{-1} \left[\frac{A}{1 - re^{j\theta} z^{-1}} + \frac{B}{1 - re^{-j\theta} z^{-1}} \right] = [A(re^{j\theta})^n + B(re^{-j\theta})^n] u[n] \\ &= r^n \frac{\sin((n+1)\theta)}{\sin \theta} u[n] \end{aligned} \quad (6.287)$$

This is the underdamped case of a discrete 2nd order system. We see that r and θ dictate the decay rate and oscillation frequency of the response, respectively, corresponding to ζ and ω_n in the impulse response $h(t)$ of a continuous system (Eq.6.126).

The step response of a discrete 2nd order system can be found as:

$$\begin{aligned} y[n] &= h[n] * u[n] = \sum_{m=0}^n h[m] = \left[A \sum_{m=0}^n (re^{j\theta})^m + B \sum_{m=0}^n (re^{-j\theta})^m \right] u[n] \\ &= \left[A \frac{1 - re^{j(n+1)\theta}}{1 - re^{j\theta}} - B \frac{1 - re^{-j(n+1)\theta}}{1 - re^{-j\theta}} \right] u[n] \\ &= \frac{\sin \theta - r^{n+1} \sin((n+2)\theta) + r^{n+2} \sin((n+1)\theta)}{\sin \theta(1 - 2r \cos \theta + r^2)} u[n] \end{aligned} \quad (6.288)$$

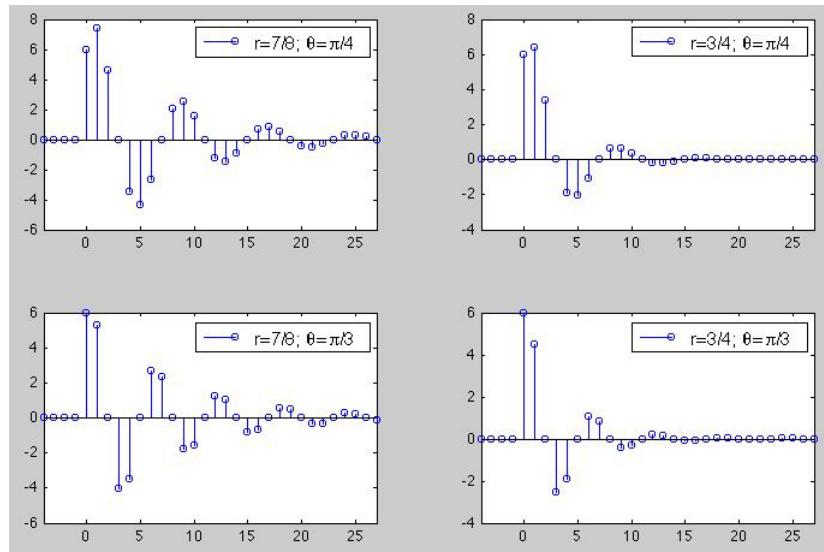


Figure 6.18 Impulse Response of 2nd Order System

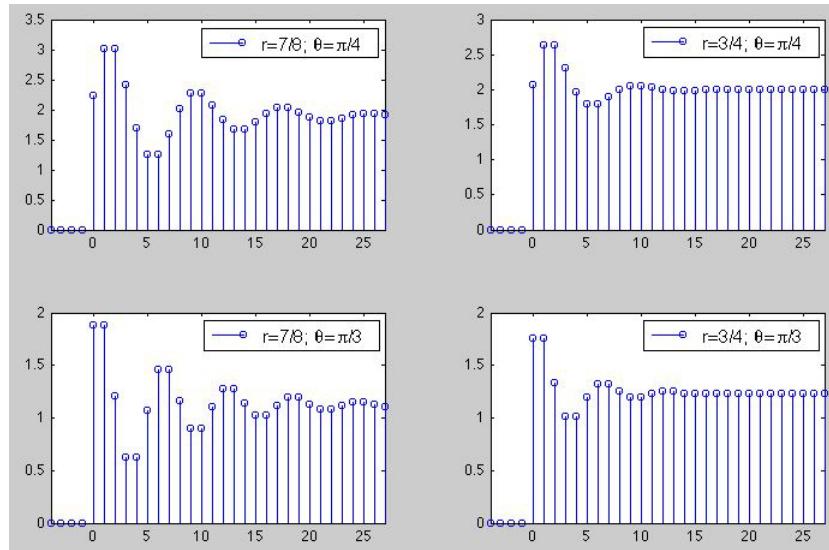


Figure 6.19 Step Response of 2nd Order System

In these examples for the first and second order systems, the numerator of $H(z)$ is always 1 corresponding to the assumed input $x[n]$ on the right-hand side of the LCCDE. However, as shown in Eq.6.263 for a general LTI system, the numerator is typically a polynomial corresponding to the input $\sum_k b_k x[n - k]$ on the right-hand side of Eq.6.261. Of course we could resolve the system following the steps above. However, as the LCCDE is an LTI system, once the response

$y[n]$ to an input $x[n]$ is known, we can find its response to $\sum_k b_k x[n - k]$ to be $\sum_k b_k y[n - k]$.

A discrete LTI system, such as the first and second order systems, can also be considered as a filter characterized by the magnitude and phase of its frequency response function $H(e^{j\omega}) = H(z) \Big|_{z=e^{j\omega}}$:

$$\begin{aligned} |H(e^{j\omega})| &= \frac{\prod_{k=1}^M |e^{j\omega} - z_k|}{\prod_{k=1}^N |e^{j\omega} - p_k|} = \frac{\prod_{k=1}^M |\mathbf{u}_k|}{\prod_{k=1}^N |\mathbf{v}_k|} \\ \angle H(e^{j\omega}) &= \frac{\sum_{k=1}^M \angle(e^{j\omega} - z_k)}{\sum_{k=1}^N \angle(e^{j\omega} - p_k)} = \frac{\sum_{k=1}^M \angle \mathbf{u}_k}{\sum_{k=1}^N \angle \mathbf{v}_k} \end{aligned} \quad (6.289)$$

where each factor $\mathbf{u}_k = e^{j\omega} - z_k$ or $\mathbf{v}_k = e^{j\omega} - p_k$ is a vector in z-plane that connects the point $e^{j\omega}$ on the unit circle and one of the zeros or poles. The filtering effects of the system are therefore dictated by the zero and pole locations on the z-plane and can be qualitatively determined by observing how $|H(e^{j\omega})|$ and $\angle H(e^{j\omega})$ change when frequency ω varies along the unit circle from $-\pi$ to π .

The frequency response function of the first order system in Eq. 6.279 is:

$$H(e^{j\omega}) = H(z) \Big|_{z=e^{j\omega}} = \frac{1}{1 - pe^{-j\omega}} = \frac{e^{j\omega}}{e^{j\omega} - p} = \frac{\mathbf{u}}{\mathbf{v}} \quad (6.290)$$

where $p = re^{j\theta}$ is the pole of the system and the zero is at the origin, and $\mathbf{u} = e^{j\omega}$ and $\mathbf{v} = e^{j\omega} - p = e^{j\omega} - re^{j\theta}$ are the two vectors connecting $e^{j\omega}$ to the zero and pole, respectively, as shown on the left in Fig.6.20. While the magnitude of \mathbf{u} is unity, the magnitude of \mathbf{v} varies as ω moves from $-\pi$ to π , and it reaches minimum when $\omega = 0$ and $e^{j\omega} = 1$. We can therefore qualitatively determine that the system is a low-pass filter with maximum magnitude at zero frequency, as shown in the top panel of Fig.6.21.

The frequency response function of the first order system in Eq. 6.283 is:

$$H(e^{j\omega}) = H(z) \Big|_{z=e^{j\omega}} = \frac{e^{j2\omega}}{(e^{j\omega} - p_1)(e^{j\omega} - p_2)} = \frac{\mathbf{u}_1 \mathbf{u}_2}{\mathbf{v}_1 \mathbf{v}_2} \quad (6.291)$$

where $p_{1,2} = re^{\pm j\theta}$ are the two poles and the double zeros are at the origin, and $\mathbf{u}_1 = \mathbf{u}_2 = e^{j\omega}$ and $\mathbf{v}_{1,2} = e^{j\omega} - p_{1,2} = e^{j\omega} - re^{\pm j\theta}$ are the vectors connecting $e^{j\omega}$ to the two zeros and two poles, respectively, as shown on the right in Fig.6.20. While the magnitude of $\mathbf{u}_1 = \mathbf{u}_2$ is unity, the magnitudes of \mathbf{v}_1 and \mathbf{v}_2 vary as ω moves from $-\pi$ to π , and they reach minimum when $\omega = \pm\theta$. We can therefore qualitatively determine that the system is a band-pass filter with center frequency of the passing band around $\omega = \theta$, as shown in the bottom panel of Fig.6.21.

6.2.7 The Unilateral z-Transform

Same as the bilateral Laplace transform, the bilateral z-transform does not take initial condition into consideration while solving difference equations, and this

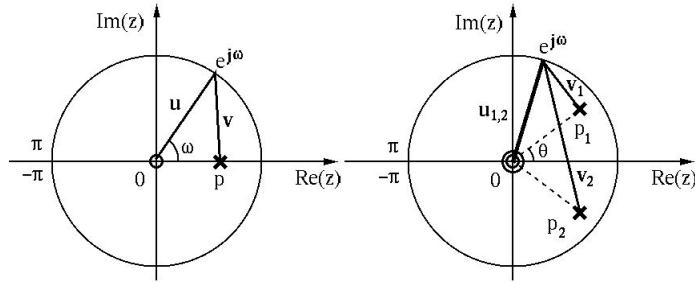


Figure 6.20 First (left) and second (right) order filters in z-plane

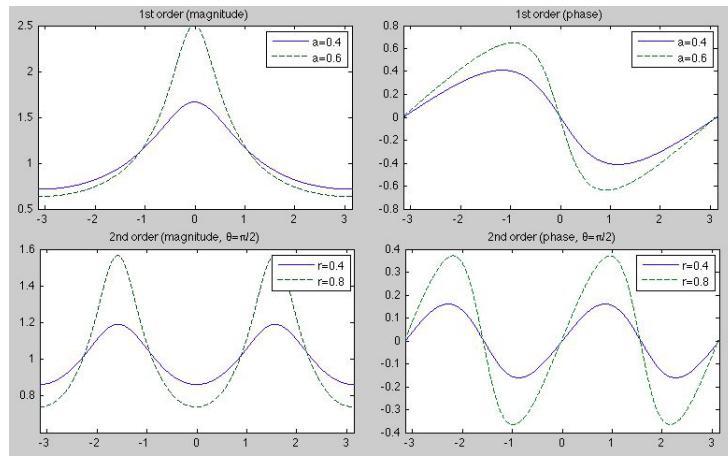


Figure 6.21 Filters of first (top) and second order (bottom)

The magnitudes of the filters are shown on the left and the phases are shown on the right.

problem can be resolved by the *unilateral z-transform* defined below:

$$\mathcal{UZ}[x[n]] = X(z) = \sum_{n=-\infty}^{\infty} x[n]u[n]z^{-n} = \sum_{n=0}^{\infty} x[n]z^{-n} \quad (6.292)$$

When the unilateral z-transform is applied to a signal $x[n]$, it is always assumed that the signal starts at time $n = 0$, i.e., $x[n] = 0$ for $n < 0$; when it is applied to the impulse response function of a LTI system to find the transfer function $H(z) = \mathcal{UZ}[h[n]]$, it is always assumed that the system is causal, i.e., $h[n] = 0$ for $n < 0$. In both cases, the ROC is always the exterior of a circle.

By definition, the unilateral z-transform of any signal $x[n] = x[n]u[n]$ is identical to its bilateral z-transform. However, when $x[n] \neq x[n]u[n]$, the two z-transforms are different. Some of the properties of the unilateral z-transform different from the bilateral z-transform are listed below.

- **Time advance**

$$\begin{aligned} \mathcal{UZ}[x[n+1]] &= \sum_{n=0}^{\infty} x[n+1]z^{-n} = z \sum_{m=1}^{\infty} x[m]z^{-m} \\ &= z \left[\sum_{m=0}^{\infty} x[m]z^{-m} - x[0] \right] = zX(z) - zx[0] \end{aligned} \quad (6.293)$$

where $m = n + 1$.

- **Time delay**

$$\begin{aligned} \mathcal{UZ}[x[n-1]] &= \sum_{n=0}^{\infty} x[n-1]z^{-n} = z^{-1} \sum_{m=-1}^{\infty} x[m]z^{-m} \\ &= z^{-1} \left[\sum_{m=0}^{\infty} x[m]z^{-m} + zx[-1] \right] = z^{-1}X(z) + x[-1] \end{aligned} \quad (6.294)$$

where $m = n - 1$. Similarly, we have

$$\begin{aligned} \mathcal{UZ}[x[n-2]] &= \sum_{n=0}^{\infty} x[n-2]z^{-n} = z^{-2} \sum_{m=-2}^{\infty} x[m]z^{-m} \\ &= z^{-2} \left[\sum_{n=0}^{\infty} x[n]z^{-n} + zx[-1] + z^2x[-2] \right] = z^{-2}X(z) + x[-1]z^{-1} + x[-2] \end{aligned} \quad (6.295)$$

where $m = n - 2$. In general, we have

$$\mathcal{UZ}[x[n-n_0]] = z^{-n_0}X(z) + \sum_{k=0}^{n_0-1} z^{-k}x[k-n_0] \quad (6.296)$$

- **Initial value theorem**

If $x[n] = x[n]u[n]$, i.e., $x[n] = 0$ for $n < 0$, then

$$x[0] = \lim_{z \rightarrow \infty} X(z) \quad (6.297)$$

Proof:

$$\lim_{z \rightarrow \infty} X(z) = \lim_{z \rightarrow \infty} \left[\sum_{n=0}^{\infty} x[n]z^{-n} \right] = x[0] \quad (6.298)$$

This is because all terms with $n > 0$ become zero as $z^{-n} = 1/z^n \rightarrow 0$ as $z \rightarrow \infty$.

- **Final value theorem**

If $x[n] = x[n]u[n]$, i.e., $x[n] = 0$ for $n < 0$, then

$$\lim_{n \rightarrow \infty} x[n] = \lim_{z \rightarrow 1} (1 - z^{-1})X(z) \quad (6.299)$$

Proof:

$$\mathcal{Z}[x[n] - x[n-1]] = \sum_{n=0}^{\infty} [x[n] - x[n-1]]z^{-n} = X(z) - X(z)z^{-1} \quad (6.300)$$

i.e.

$$(1 - z^{-1})X(z) = \lim_{m \rightarrow \infty} \sum_{n=0}^m [x[n] - x[n-1]]z^{-n} \quad (6.301)$$

Letting $z \rightarrow$, we get

$$\begin{aligned} \lim_{z \rightarrow 1} (1 - z^{-1})X(z) &= \lim_{m \rightarrow \infty} \sum_{n=0}^m [x[n] - x[n-1]] \\ &= \lim_{m \rightarrow \infty} \left[\sum_{n=0}^{m-1} [x[n] - x[n]] + x[m] - x[-1] \right] = \lim_{m \rightarrow \infty} x[m] \end{aligned}$$

Note that $x[-1] = 0$.

Due to the initial and final value theorems, the unilateral z-transform is a powerful tool for solving LCCDEs with non-zero initial conditions.

Example 6.9: Given a signal $x[n] = a^{-(n+1)}u[n+1]$, find both the bilateral and unilateral z-transforms. Note that this signal is right-sided starting at $n = -1$ (i.e., $x[n] \neq x[n]u[n]$). By definition, the bilateral z-transform of $x[n]$ is:

$$\begin{aligned} \mathcal{Z}[x[n]] &= \sum_{n=-1}^{\infty} a^{-(n+1)}z^{-n} = z + a^{-1} \sum_{n=0}^{\infty} a^{-n}z^{-n} \\ &= z + \frac{a^{-1}}{1 - (az)^{-1}} = \frac{z}{1 - (az)^{-1}} \end{aligned} \quad (6.302)$$

It was assumed that $|z| > a$. The unilateral z-transform of this signal is

$$\mathcal{UZ}[x[n]] = \sum_{n=0}^{\infty} a^{-(n+1)}z^{-n} = a^{-1} \sum_{n=0}^{\infty} (az)^{-n} = \frac{a^{-1}}{1 - (az)^{-1}} \quad (6.303)$$

If we assume zero initial condition $y[-1] = 0$,

Example 6.10: A system is described by this LCCDE

$$y[n] + 3y[n-1] = x[n] = \alpha u[n] \quad (6.304)$$

Taking unilateral z-transform of the DE, we get

$$Y(z) + 3Y(z)z^{-1} + 3y[-1] = X(z) = \frac{\alpha}{1 - z^{-1}} \quad (6.305)$$

- **The particular (zero-state) solution**

If the system is initially at rest, i.e., $y[-1] = 0$, the above equation can be solved for the output $Y(z)$ to get

$$Y(z) = H(z)X(z) = \frac{1}{1 + 3z^{-1}} \frac{\alpha}{1 - z^{-1}} = \frac{3\alpha/4}{1 + 3z^{-1}} + \frac{\alpha/4}{1 - z^{-1}} \quad (6.306)$$

where $H(z) = 1/(1 + 3z^{-1})$ is the system's transfer function. In time domain this is the particular (or zero-state) solution (caused by the input with zero initial condition):

$$y_p[n] = \alpha\left[\frac{1}{4} + \frac{3}{4}(-3)^n\right]u[n] \quad (6.307)$$

- **The homogeneous (zero-input) solution**

When the initial condition is nonzero

$$y[-1] = \beta \quad (6.308)$$

but the input is zero $x[n] = 0$, the z-transform of the difference equation becomes

$$Y(z) + 3Y(z)z^{-1} + 3\beta = 0 \quad (6.309)$$

Solving this for $Y(z)$ we get

$$Y(z) = \frac{-3\beta}{1 + 3z^{-1}} \quad (6.310)$$

In time domain, this is the homogeneous (or zero-input) solution (caused by the initial condition with zero input):

$$y_h[n] = -3\beta(-3)^n u[n] \quad (6.311)$$

When neither $y[-1]$ nor $x[n]$ is zero, we have

$$Y(z) + 3Y(z)z^{-1} + 3\beta = X(z) = \frac{\alpha}{1 - z^{-1}} \quad (6.312)$$

Solving this algebraic equation in z-domain for $Y(z)$ we get

$$Y(z) = \frac{\alpha}{(1 + 3z^{-1})(1 - z^{-1})} - \frac{3\beta}{1 + 3z^{-1}} \quad (6.313)$$

The first term is the particular solution caused by the input alone and the second term is the homogeneous solution caused by the initial condition alone. The $Y(z)$ can be further written as

$$Y(z) = \frac{1}{1 + 3z^{-1}}\left(\frac{3}{4}\alpha - 3\beta\right) + \frac{\alpha}{4}\frac{1}{1 - z^{-1}} \quad (6.314)$$

and in time domain, we have the general solution

$$y_g[n] = \left[\left(\frac{3}{4}\alpha - 3\beta\right)(-3)^n + \frac{\alpha}{4}\right]u[n] = y_h[n] + y_p[n] \quad (6.315)$$

which is the sum of both the homogeneous and particular solutions.

Note that bilateral z-transform can also be used to solve LCCDEs. However, as bilateral z-transform does not take initial condition into account, it is always implicitly assumed that the system is initially at rest. If this is not the case, unilateral z-transform has to be used.

6.3 Homework Problems

1. Find the Laplace transform and the corresponding ROC of the following signals.
 - a. $x(t) = [e^{-2t} + e^t \cos(3t)]u(t)$ (Write $X(s) = \mathcal{L}[x(t)]$ in the form of a rational function, a ratio of two polynomials.)
 - b. $x(t) = e^{-a|t|} = e^{-at}u(t) + e^{at}u(-t)$ (Consider both cases: (1) $a > 0$ and (2) $a < 0$.)
 - c. Another two-sided signal $x(t) = e^{-at}u(t) - e^{-bt}u(-t)$:
 - d. $x(t) = u(-1) - u(1)$
2. Given the following Laplace transform $X(s)$, find the time function $x(t)$ corresponding to each of the possible ROCs. In each case, decide if $x(t)$ is stable or not, if it is left-sided or right-sided.
 - a. $X(s) = \frac{s^2 - 3}{s + 2}$
 - b. $X(s) = \frac{1}{(s+1)(s+2)} = \frac{1}{s+1} - \frac{1}{s+2}$
3. Given the transfer functions $H_R(s)$ and $H_L(s)$ in Eqs.6.135 and 6.136 respectively, find the impulse responses $h_R(t)$ and $h_L(t)$. Assume $0 < \zeta < 1$. Check to confirm Kirchhoff's voltage law:

$$h_C(t) + h_R(t) + h_L(t) = \delta(t) \quad (6.316)$$

where $h_C(t)$ is given in Eq.6.140. Do this in two different ways:

4. In Eqs.6.141 and 6.142 we considered only the step response $Y_C(s)$ in s-domain and $y_C(t)$ in time domain when the voltage across C in an RCL system is treated as the output.
 - Find the step responses $Y_L(s)$ and $y_L(t)$ when the voltage across L is treated as the output.
 - Find the step responses $Y_R(s)$ and $y_R(t)$ when the voltage across R is treated as the output.
 - Verify your results by Kirchhoff's voltage law:

$$Y_C(s) + Y_R(s) + Y_L(s) = 1/s, \quad y_C(t) + y_R(t) + y_L(t) = u(t) \quad (6.317)$$

5. Consider an RC circuit with input voltage $v_{in}(t) = A \cos(\omega t)u(t)$ applied to the series combination of a resistor R and a capacitor C (representing a sinusoidal input and a switch which is closed at $t = 0$). The initial voltage on C for $t \leq 0$ to be $v_C(0)$. Use unilateral Laplace transform method to find the voltage $v_C(t)$ across C for $t > 0$.
6. Use Laplace transform method to find the response of a 2nd-order system to a sinusoidal input $x(t) = \cos(\omega_0 t)u(t)$:

$$\ddot{y}(t) + 2\zeta\omega_n\dot{y}(t) + \omega_n^2 y(t) = x(t) = \cos(\omega_0 t)u(t) \quad (6.318)$$

Assume zero initial conditions: $y(0) = \dot{y}(0) = 0$.

7. An LTI system is described by the following LCCDE:

$$\frac{d^2y(t)}{dt^2} + 3\frac{dy(t)}{dt} + 2y(t) = x(t) \quad (6.319)$$

- Find the particular solution $y_p(t)$ (with zero initial conditions) when the input is $x(t) = e^{-3t}u(t)$.
- Find the homogeneous solution $y_h(t)$ (with zero input $x(t) = 0$) with initial conditions:

$$y(0) = 1, \quad \dot{y}(0) = \left. \frac{dy(t)}{dt} \right|_{t=0} = -1 \quad (6.320)$$

- Find the complete solution $y(t) = y_p(t) + y_h(t)$.
- 8. Find the z-transform and the corresponding ROC of the following signals.

- a. $x[n] = 0$ for all n except $x[-1] = x[0] = x[1] = 1$

Solution:

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

As $z^{-n} = \infty$ when $z = \infty$ or $z = 0$, the ROC is the entire z plane excluding these two z values.

- b. $x[n] = b^{|n|}$. Consider both cases $b > 1$ and $b < 1$.

Solution:

$$x[n] = b^{|n|} = b^n u[n] + b^{-n} u[-n-1]$$

For the right-sided part:

$$\mathcal{Z}[b^n u[n]] = \sum_{n=0}^{\infty} b^n z^{-n} = \sum_{n=0}^{\infty} (bz^{-1})^n = \frac{1}{1 - bz^{-1}} \quad |z| > b$$

For the left-sided part:

$$\begin{aligned} \mathcal{Z}[b^{-n} u[-n-1]] &= \sum_{n=-\infty}^{-1} b^{-n} z^{-n} = \sum_{n=0}^{\infty} (bz)^n - 1 \\ &= \frac{1}{1 - bz} - 1 = \frac{-1}{1 - (bz)^{-1}} \quad |z| < 1/b \end{aligned}$$

The ROC for both parts combined is the intersection of the individual ROCs: $b < |z| < 1/b$. When $b < 1$, $x[n]$ decays on both sides as $n \rightarrow \infty$ and its ROC is a ring, and we have

$$\mathcal{Z}[b^{|n|}] = \frac{1}{1 - bz^{-1}} + \frac{-1}{1 - (bz)^{-1}}$$

But when $b > 1$, $x[n]$ grows on both sides and it is not absolutely summable, correspondingly its ROC is an empty set, i.e., its z-transform does not exist.

- c. Another two-sided signal $x[n] = a^{-n}u[n] - b^{-n}u[-n-1]$

Solution:

$$X(z) = \frac{1}{1 - (az)^{-1}} + \frac{1}{1 - (bz)^{-1}}, \quad |z| > a, \quad |z| < 1/b$$

provided both conditions $|z| > 1/a$ and $|z| < 1/b$ are satisfied, or $1/a < |z| < 1/b$. However, this is only possible if $a > b$. If $a < b$, $X(s)$ given above does not exist.

9. Given the following z-transform $X(z)$, find the corresponding discrete signals.
a.

$$X(z) = \frac{1 - \frac{1}{2}z^{-1}}{1 + 2z^{-1} - 3z^{-2}}, \quad |z| > 3 \quad (6.321)$$

b.

$$X(z) = \frac{1 - \frac{1}{2}z^{-1}}{1 + \frac{1}{2}z^{-1}}, \quad |z| > 1/2 \quad (6.322)$$

10. Given a discrete signal $x[n]$ as shown below:

n	...	-1	0	1	2	3	4	...
$x[n]$...	0	1	2	3	4	0	...

find the z-transforms of $y[n] = x_{(2)}[n]$ and then $z[n] = y^{(2)}[n]$ using the up and down-sampling properties, and compare them with $Y(z)$ and $Z(z)$ obtained directly from the definition of the z-transform.

11. Given the input $x[n]$ and the response $y[n]$ below, find the impulse response $h[n]$ of the LTI system, and decide if the system is causal and stable.

$$x[n] = \left(\frac{1}{5}\right)^n u[n], \quad y[n] = \left[3\left(\frac{1}{2}\right)^n - 2\left(\frac{1}{3}\right)^n\right] u[n] \quad (6.323)$$

12. Find the impulse response $h[n]$ and step response $y[n]$ of the following discrete LTI system:

$$y[n] - 2r \cos \theta y[n-1] + r^2 = x[n-1] \quad (6.324)$$

First, take the approach used in the text ($y[n] = \mathcal{Z}^{-1}[H(z)X(z)]$) to find the responses when the right-hand side is $x[n]$, and then confirm your results by time invariance: if $\mathcal{O}[x[n]] = y[n]$ then $\mathcal{O}[x[n-k]] = y[n-k]$.

13. An LTI system is described by the following LCCDE:

$$6y[n] + 5y[n-1] + y[n-2] = 2x[n-1] - x[n-2] \quad (6.325)$$

- a. Find the transfer function $H(z) = Y(z)/X(z)$ and impulse response $h[n]$;
- b. Obtain an inverse system $G(z) = 1/H(z)$, so that when the two systems are cascaded the output of $G(z)$ is same as the input to $H(z)$, i.e., $Z(z) = G(x)Y(z) = G(z)H(z)X(z) = X(z)$ or $z[n] = x[n]$;
- c. Find the impulse response $g[n]$ and the corresponding LCCDE of the inverse system in terms of input $y[n]$ and output $z[n]$. Show the inverse system is not causal.
- d. Introduce a unit delay z^{-1} in the system so that the resulting system $G'(z) = G(z)z^{-1}$ is causal, i.e., its output is same as the input $z[n] = x[n-1]$. Get its input response $g'[n]$ and give the corresponding LCCDE in terms of input $y[n]$ and output $z[n]$.

14. Design the following four types of filters by specifying the zero and pole positions of a rational transfer function $H(s)$. Use minimum number of zeros and poles (no more than 2 for each). For each of the four cases, determine the expression of the frequency response function $H(j\omega)$ and sketch the magnitude plots $|H(j\omega)|$ for $-2\pi 100 < \omega < 2\pi 100$. to verify your design.
 - Low-pass filter;
 - High-pass filter;
 - Band-pass filter with passing band centered around $\pm 50\pi$;
 - Band-stop filter with passing band centered around $\pm 50\pi$;
15. Design the following four types of filters by specifying the zero and pole positions of a rational transfer function $H(z)$. Use minimum number of zeros and poles (no more than 2 for each). For each of the four cases, determine the expression of the frequency response function $H(e^{j\omega})$ and sketch the magnitude plots $|H(e^{j\omega})|$ for $-\pi < \omega < \pi$ to verify your design.
 - Low-pass filter;
 - High-pass filter;
 - Band-pass filter with passing band centered around $\pi/2$;
 - Band-stop filter with passing band centered around $\pi/2$;
16. Use the provided Matlab function ZeroPolePlots.m to explore the following:
 - The filtering effect of a continuous system with frequency response function $H(j\omega)$ (including both magnitude and phase) with different numbers of zeros and poles and various locations. In particular, set the order of the denominator polynomial to $N = 2$, and explore the Bode plots of $H(j\omega)$ for different order of the numerator polynomial $M = 1, 2, 3$.
 - The filtering effect of a discrete system with frequency response functions $H(e^{j\omega})$ (including both magnitude and phase) with different numbers of zeros and poles and various locations.

7 Fourier Related Orthogonal Transforms

The Fourier transform converts a complex signal into its complex spectrum. If the signal is real, as in most applications, the imaginary part of the signal is zero, and its spectrum is symmetric, i.e., in both time and frequency domains half of the data is redundant, causing unnecessary computational time and storage space. In this chapter, we will consider three real orthogonal transforms, all closely related to the Fourier transform with similar behaviors, but the problem of data redundancy is avoided. Here we will always assume the signal in question is real.

7.1 The Hartley Transform

7.1.1 Continuous Hartley Transform

The Hartley transform is an integral transform based on a real kernel function:

$$\begin{aligned}\phi_f(t) &= cas(2\pi ft) = \cos(2\pi ft) + \sin(2\pi ft) \\ &= \sqrt{2} \sin\left(2\pi ft + \frac{\pi}{4}\right) = \sqrt{2} \cos\left(2\pi ft - \frac{\pi}{4}\right) = \phi_t(f), \\ &\quad (-\infty < t, f < \infty)\end{aligned}\tag{7.1}$$

Here $cas(2\pi ft)$ is the *cosine-and-sine (CAS) function* defined as:

$$cas(2\pi ft) = \cos(2\pi ft) + \sin(2\pi ft)\tag{7.2}$$

We can show that this is a set of uncountable orthonormal functions satisfying:

$$\langle \phi_f(t), \phi_{f'}(t) \rangle = \delta(f - f'), \quad \text{and} \quad \langle \phi_t(f), \phi_{t'}(f) \rangle = \delta(t - t')\tag{7.3}$$

Proof:

$$\begin{aligned}
<\phi_f(t), \phi_{f'}(t)> &= \int_{-\infty}^{\infty} \phi_f(t) \phi_{f'}(t) dt \\
&= \int_{-\infty}^{\infty} [\cos(2\pi ft) + \sin(2\pi ft)] [\cos(2\pi f't) + \sin(2\pi f't)] dt \\
&= \int_{-\infty}^{\infty} [\cos(2\pi ft) \cos(2\pi f't) + \sin(2\pi ft) \sin(2\pi f't)] dt \\
&+ \int_{-\infty}^{\infty} [\cos(2\pi ft) \sin(2\pi f't) + \sin(2\pi ft) \cos(2\pi f't)] dt \\
&= \int_{-\infty}^{\infty} \cos(2\pi(f - f')t) dt + \int_{-\infty}^{\infty} \sin(2\pi(f + f')t) dt = \delta(f - f') \quad (7.4)
\end{aligned}$$

Here the first term is a Dirac delta $\delta(f - f')$ according to Eq.1.28, while the second term, an integral of an odd function $\sin(2\pi(f + f')t)$ over all t , is zero and therefore dropped. The second equation of Eq. 7.3 follows immediately as $\phi_f(t) = \phi_t(f)$ is symmetric with respect to t and f .

Given the transform kernel $\phi_f(t) = cas(2\pi ft)$, the Hartley transform is defined as:

$$\begin{aligned}
X_H(f) = \mathcal{H}[x(t)] &= <x(t), \phi_f(t)> = \int_{-\infty}^{\infty} x(t) cas(2\pi ft) dt \\
&= \int_{-\infty}^{\infty} x(t) [\cos(2\pi ft) + \sin(2\pi ft)] dt \quad (7.5)
\end{aligned}$$

Here $X_H(f)$ is a function of frequency f and is called the Hartley spectrum of the signal $x(t)$, similar to its Fourier spectrum $X_F(f)$.

The inverse Hartley transform can be obtained by taking an inner product with $\phi_{f'}(t') = \phi_t(f')$ on both sides of the forward transform above:

$$\begin{aligned}
< X_H(f), \phi_{f'}(f) > &= \int_{-\infty}^{\infty} X_H(f) \phi_{f'}(f)(f) df = \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} x(t) \phi_f(t) dt \right] \phi_{f'}(f)(f) df \\
&= \int_{-\infty}^{\infty} x(t) \left[\int_{-\infty}^{\infty} \phi_f(t) \phi_{f'}(f)(f) df \right] dt \\
&= \int_{-\infty}^{\infty} x(t) \delta(t - t') dt = x(t') \quad (7.6)
\end{aligned}$$

Putting both the forward and inverse Hartley transforms together, we get the following pair of equations:

$$\begin{aligned}
X_H(f) = \mathcal{H}[x(t)] &= <x(t), \phi_f(t)> = \int_{-\infty}^{\infty} x(t) cas(2\pi ft) dt \\
x(t) = \mathcal{H}^{-1}[X_H(f)] &= < X_H(f), \phi_t(f) > = \int_{-\infty}^{\infty} X_H(f) cas(2\pi ft) df \quad (7.7)
\end{aligned}$$

We see that the inverse transform is identical to the forward transform:

$$x(t) = \mathcal{H}^{-1}[X_H(f)] = \mathcal{H}[X_H(f)] = \mathcal{H}[\mathcal{H}[x(t)]] \quad (7.8)$$

7.1.2 Properties of the Hartley Transform

- **Relation to Fourier transform:**

Here we assume the signal $x(t) = \bar{x}(t)$ is real. Its Hartley spectrum can be written as:

$$\begin{aligned} X_H(f) &= \mathcal{H}[x(t)] = \int_{-\infty}^{\infty} x(t)[\cos(2\pi ft) + \sin(2\pi ft)] dt \\ &= \int_{-\infty}^{\infty} x(t) \cos(2\pi ft) dt + \int_{-\infty}^{\infty} x(t) \sin(2\pi ft) dt \\ &= X_e(f) + X_o(f) \end{aligned} \quad (7.9)$$

where $X_e(f)$ and $X_o(f)$ are respectively the even and odd components of the Hartley spectrum $X_H(f)$:

$$\begin{aligned} X_e(f) &= \frac{1}{2}[X_H(f) + X_H(-f)] = \int_{-\infty}^{\infty} x(t) \cos(2\pi ft) dt \\ X_o(f) &= \frac{1}{2}[X_H(f) - X_H(-f)] = \int_{-\infty}^{\infty} x(t) \sin(2\pi ft) dt \end{aligned}$$

On the other hand, the Fourier spectrum of $x(t)$ is:

$$\begin{aligned} X_F(f) &= \mathcal{F}[x(t)] = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft} dt = \int_{-\infty}^{\infty} x(t)[\cos(2\pi ft) - j\sin(2\pi ft)] dt \\ &= \int_{-\infty}^{\infty} x(t) \cos(2\pi ft) dt - j \int_{-\infty}^{\infty} x(t) \sin(2\pi ft) dt \\ &= X_e(f) - j X_o(f) \end{aligned} \quad (7.10)$$

We see that both the Hartley and Fourier spectra of a real signal $x(t)$ are composed of the same even and odd components $X_e(f)$ and $X_o(f)$, which are also the real and imaginary parts (negative version) of the Fourier spectrum $X_F(f)$:

$$X_e(f) = \text{Re}[X_F(f)], \quad X_o(f) = -\text{Im}[X_F(f)] \quad (7.11)$$

i.e., the Hartley spectrum can be obtained as a linear combination of the real and imaginary parts of the Fourier spectrum:

$$X_H(f) = X_e(f) + X_o(f) = \text{Re}[X_F(f)] - \text{Im}[X_F(f)] \quad (7.12)$$

In particular, we consider the two cases when the real signal $x(t)$ is either even or odd:

- If $x(t) = x(-t)$ is even, its Fourier spectrum is real and even with $\text{Im}[X_F(f)] = 0$ and $X_H(f) = X_F(f)$, i.e., the Hartley spectrum is identical to the Fourier spectrum.
- If $x(t) = -x(-t)$ is odd, its Fourier spectrum is imaginary and odd with $\text{Re}[X_F(f)] = 0$ and $X_H(f) = -X_F(f)$, i.e., the Hartley spectrum is the negative version of its Fourier spectrum.

- **Convolution in both time and frequency domains:**

Let $z(t) = x(t) * y(t)$ be the convolution of $x(t)$ and $y(t)$, then the Hartley spectrum $Z_H(f) = \mathcal{H}[z(t)]$ is:

$$\begin{aligned} Z_H(f) &= \mathcal{H}[x(t) * y(t)] \\ &= \frac{1}{2} [X_H(f)Y_H(f) - X_H(-f)Y_H(-f) + X_H(f)Y_H(-f) + X_H(-f)Y_H(f)] \end{aligned} \quad (7.13)$$

where $X_H(f) = \mathcal{H}[x(t)]$ and $Y_H(f) = \mathcal{H}[y(t)]$ are the Hartley spectra of $x(t)$ and $y(t)$, respectively.

Proof:

According to the convolution theorem of the Fourier transform (Eq.3.112), the Fourier spectrum $Z_F(f) = \mathcal{F}[z(t)]$ is the product of the spectra $X_F(f) = \mathcal{F}[x(t)]$ and $Y_F(f) = \mathcal{F}[y(t)]$:

$$\begin{aligned} Z_F(f) &= X_F(f) Y_F(f) = [X_e(f) - j X_o(f)] [Y_e(f) - j Y_o(f)] \\ &= [X_e(f)Y_e(f) - X_o(f)Y_o(f)] - j [X_o(f)Y_e(f) + X_e(f)Y_o(f)] \\ &= Z_e(f) - j Z_o(f) \end{aligned} \quad (7.14)$$

where $Z_e(f)$ and $Z_o(f)$ are respectively the even and odd components of $Z_H(f)$:

$$\begin{aligned} Z_e(f) &= X_e(f)Y_e(f) - X_o(f)Y_o(f) = \frac{1}{2}[X_H(f)Y_H(-f) + X_H(-f)Y_H(f)] \\ Z_o(f) &= X_e(f)Y_o(f) + X_o(f)Y_e(f) = \frac{1}{2}[X_H(f)Y_H(f) - X_H(-f)Y_H(-f)] \end{aligned} \quad (7.15)$$

Substituting these into $Z_H(f) = Z_e(f) + Z_o(f)$, we get Eq.7.13.

Also, based on Eq.3.113, we can similarly prove the Hartley spectrum of the product of two functions $z(t) = x(t)y(t)$ is:

$$\begin{aligned} Z_H(t) &= \mathcal{H}[x(t)y(t)] \\ &= \frac{1}{2}[X_H(f) * Y_H(f) - X_H(-f) * Y_H(-f) \\ &\quad + X_H(f) * Y_H(-f) + X_H(-f) * Y_H(f)] \end{aligned} \quad (7.16)$$

- **Correlation:**

Let $z(t) = x(t) \star y(t)$ be the correlation of $x(t)$ and $y(t)$, then the Hartley spectrum $Z_H(f) = \mathcal{H}[z(t)]$ is:

$$\begin{aligned} Z_H(f) &= \mathcal{H}[x(t) \star y(t)] \\ &= \frac{1}{2} [X_H(f)Y_H(f) + X_H(-f)Y_H(-f) + X_H(f)Y_H(-f) - X_H(-f)Y_H(f)] \end{aligned} \quad (7.17)$$

In particular, when $x(t) = y(t)$, i.e., $X_H(f) = Y_H(f)$, then the odd part $Z_o(f)$ of its spectrum is zero, and the correlation $x(t) \star y(t) = x(t) \star x(t)$ becomes

autocorrelation, the Eq.7.17 becomes:

$$\mathcal{H}[x(t) \star x(t)] = \frac{1}{2}[X_H^2(f) + X_H^2(-f)] \quad (7.18)$$

Proof:

According to the correlation property of the Fourier transform (Eq.3.107), the Fourier spectrum $Z_F(f) = \mathcal{F}[z(t)]$ is the product of the spectra $X_F(f) = \mathcal{F}[x(t)]$ and $Y_F(f) = \mathcal{F}[y(t)]$:

$$\begin{aligned} Z_F(f) &= X_F(f) \overline{Y_F(f)} = [X_e(f) - j X_o(f)] [Y_e(f) + j Y_o(f)] \\ &= [X_e(f)Y_e(f) + X_o(f)Y_o(f)] - j [X_o(f)Y_e(f) - X_e(f)Y_o(f)] \\ &= Z_e(f) - j Z_o(f) \end{aligned} \quad (7.19)$$

where $X_e(f)$, $X_o(f)$ and $Y_e(f)$, $Y_o(f)$ are the even and odd components of $X_H(f)$ and $Y_H(f)$, respectively:

$$\begin{aligned} X_e(f) &= \frac{1}{2}[X_H(f) + X_H(-f)], & X_o(f) &= \frac{1}{2}[X_H(f) - X_H(-f)] \\ Y_e(f) &= \frac{1}{2}[Y_H(f) + Y_H(-f)], & Y_o(f) &= \frac{1}{2}[Y_H(f) - Y_H(-f)] \end{aligned}$$

and $Z_e(f)$ and $Z_o(f)$ are the even and odd components of $Z_H(f)$:

$$\begin{aligned} Z_e(f) &= X_e(f)Y_e(f) + X_o(f)Y_o(f) = \frac{1}{2}[X_H(f)Y_H(f) + X_H(-f)Y_H(-f)] \\ Z_o(f) &= X_o(f)Y_e(f) - X_e(f)Y_o(f) = \frac{1}{2}[X_H(f)Y_H(-f) - X_H(-f)Y_H(f)] \end{aligned}$$

Substituting these into $Z_H(f) = Z_e(f) + Z_o(f)$, we get Eq.7.17.

7.1.3 Hartley Transform of Typical Signals

As the Hartley transform is closely related to the Fourier transform, the Hartley spectra of many signals are similar to or the same as their Fourier spectra. In particular, if the signal is either real even or real odd, its Hartley spectrum is either identical or the negative version of its Fourier spectrum. We therefore only consider the following two examples where the real signal is neither even nor odd.

- Combination of Sinusoids:

$$x(t) = \cos(2\pi f_0 t + \theta) = \frac{1}{2}[e^{j2\pi f_0 t} e^{j\theta} + e^{-j2\pi f_0 t} e^{-j\theta}] \quad (7.20)$$

The Fourier transform is:

$$\begin{aligned} X_F(f) &= \frac{1}{2}[\delta(f - f_0)e^{j\theta} + \delta(f + f_0)e^{-j\theta}] \\ &= \frac{1}{2}[\delta(f - f_0)(\cos \theta + j \sin \theta) + \delta(f + f_0)(\cos \theta - j \sin \theta)] \\ &= \frac{1}{2}[\delta(f - f_0) \cos \theta + \delta(f + f_0) \cos \theta] \\ &\quad + \frac{j}{2}[\delta(f - f_0) \sin \theta - \delta(f + f_0) \sin \theta] \end{aligned}$$

Its Hartley transform is

$$\begin{aligned} X_H(f) &= Re[X_F(f)] - Im[X_F(f)] \\ &= \frac{1}{2}[\delta(f - f_0)(\cos \theta - \sin \theta) + \delta(f + f_0)(\cos \theta + \sin \theta)] \end{aligned}$$

In particular, if $\theta = 0$, the signal becomes even $x(t) = \cos(2\pi f_0 t)$, and its Hartley spectrum becomes the same as the Fourier spectrum $X_F(f)$:

$$X_H(f) = \mathcal{H}[\cos(2\pi f_0 t)] = \frac{1}{2}[\delta(f - f_0) + \delta(f + f_0)] \quad (7.21)$$

Also if $\theta = -\pi/2$, we have $x(t) = \cos(2\pi f_0 t - \pi/2) = \sin(2\pi f_0 t)$, and its Hartley spectrum becomes:

$$X_H(f) = \mathcal{H}[\sin(2\pi f_0 t)] = \frac{1}{2}[\delta(f - f_0) - \delta(f + f_0)] \quad (7.22)$$

which is the negative version of imaginary part of the Fourier spectrum

$$X_F(f) = \frac{1}{2j}[\delta(f - f_0) - \delta(f + f_0)] = \frac{j}{2}[-\delta(f - f_0) + \delta(f + f_0)] \quad (7.23)$$

For a specific example, consider a signal containing four terms:

$$x(t) = 1 + 3 \cos(2\pi 16t) + 2 \sin(2\pi 64t) + 2 \cos(2\pi 128t + \pi/3) \quad (7.24)$$

In Fig.7.1 this signal, together with its reconstruction (dashed line) from its Hartley spectrum, is plotted (top), and its Hartley and Fourier spectra are plotted in the middle and bottom panels respectively. We see that the DC (1st term) and cosine component without phase shift (2nd term) appear the same in the two spectra, and the sine component (3rd term) appears in the two spectra as the negative version of each other. Finally the cosine function with a phase shift of $\pi/3$ (4th term) shows up in the Hartley spectrum as the difference between the real and imaginary parts of the Fourier spectrum.

- Exponential decay function:

$$x(t) = e^{-at}u(t) \quad (7.25)$$

This function together with its Hartley and Fourier spectra are shown respectively in top, middle and bottom panels of Fig.7.2.

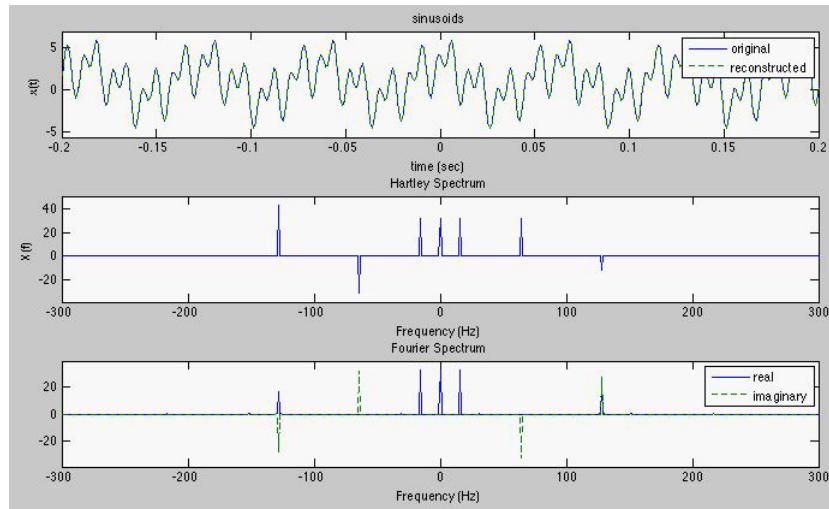


Figure 7.1 The Hartley and Fourier spectra of sinusoidal components of a signal

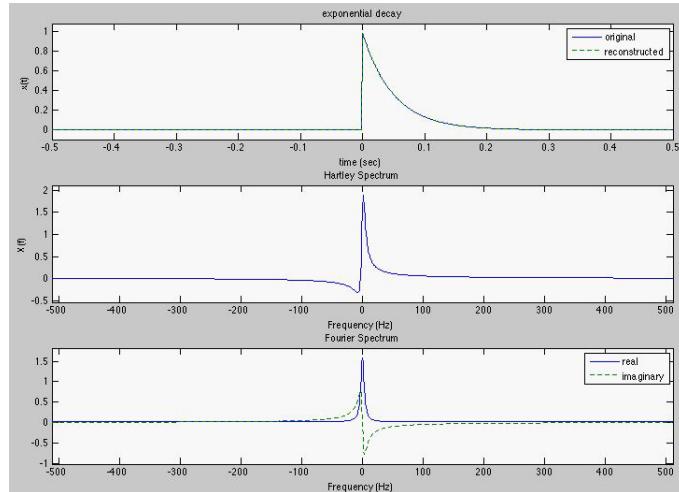


Figure 7.2 The Hartley and Fourier spectra of exponential decay

7.1.4 Discrete Hartley Transform

When a continuous signal $x(t)$ is truncated to have a finite duration $0 < t < T$ and sampled with sampling rate $F = 1/t_0$, it becomes a set of $N = T/t_0$ samples that form a vector $\mathbf{x} = [x[0], \dots, x[N-1]]^T$ in the N-D space. Correspondingly the Hartley transform also becomes discrete Hartley transform (DHT) based on

a discrete kernel:

$$\begin{aligned}\phi_k[n] &= \frac{1}{\sqrt{N}} \text{cas}\left(2\pi \left(\frac{nk}{N}\right)\right) \\ &= \frac{1}{\sqrt{N}} \left[\cos\left(2\pi \left(\frac{nk}{N}\right)\right) + \sin\left(2\pi \left(\frac{nk}{N}\right)\right) \right]\end{aligned}\quad (7.26)$$

which form a set of basis vectors $\phi_k = [\text{cas}(2\pi 0k/N), \dots, \text{cas}(2\pi (N-1)k/N)]^T$ ($k = 0, \dots, N-1$) that span the N-D vector space. We can show that these vectors are orthogonal:

$$\langle \phi_k, \phi_l \rangle = \frac{1}{N} \sum_{n=0}^{N-1} \text{cas}(2\pi nk/N) \text{cas}(2\pi nl/N) = \delta[k-l] \quad (7.27)$$

The proof is left for the reader as a homework problem. The discrete Hartley transform of a signal vector \mathbf{x} is then defined as:

$$\begin{aligned}X_H[k] &= \mathcal{H}[x[n]] = \sum_{n=0}^{N-1} x[n] \text{cas}\left(2\pi \frac{nk}{N}\right) \\ &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} \left[\cos\left(2\pi \frac{nk}{N}\right) + \sin\left(2\pi \frac{nk}{N}\right) \right]\end{aligned}\quad (7.28)$$

Here $X_H[k]$ ($k = 0, \dots, N-1$) are N frequency components of the signal, similar to the case of the discrete Fourier transform. Due to the orthogonality of ϕ_k and following the same method used to derive Eq.7.6, we get the inverse transform by which the signal can be reconstructed:

$$x[n] = \mathcal{H}^{-1}[X_H[k]] = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X_H[k] \text{cas}\left(2\pi \frac{nk}{N}\right) \quad (7.29)$$

Same as in the continuous case in Eq.7.12, the discrete Hartley transform is closely related to the discrete Fourier transform:

$$\begin{aligned}X_F[k] &= \mathcal{F}[x[n]] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x[n] e^{-j2\pi nk/N} \\ &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} \left[\cos\left(2\pi \frac{nk}{N}\right) - j \sin\left(2\pi \frac{nk}{N}\right) \right] = X_e[k] - j X_o[k], \\ &\quad (k = 0, \dots, N-1)\end{aligned}\quad (7.30)$$

where

$$\begin{aligned}X_e[k] &= \text{Re}[X_F[k]] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x[n] \cos\left(2\pi \frac{nk}{N}\right) \\ X_o[k] &= -\text{Im}[X_F[k]] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x[n] \sin\left(2\pi \frac{nk}{N}\right)\end{aligned}$$

and the discrete Hartley spectrum can also be obtained from the discrete Fourier transform:

$$X_H[k] = \mathcal{H}[x[n]] = X_e[k] + X_o[k] = Re[X_F[k]] - Im[X_F[k]] \quad (7.31)$$

Based on Eq.7.12, the discrete Hartley transform can be trivially implemented as the difference between the real and imaginary parts of the corresponding discrete Fourier transform. Correspondingly, the Hartley transform matrix can also be easily obtained as the difference between the real and imaginary parts of the Fourier transform matrix (Eq.4.120 in Chapter 4):

$$\mathbf{H} = Re[\overline{\mathbf{W}}] - Im[\overline{\mathbf{W}}] \quad (7.32)$$

In particular, the DHT matrices for $N = 2, 4, 8$ are listed below:

$$\mathbf{H}_{2 \times 2} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 0.71 & 0.71 \\ 0.71 & -0.71 \end{bmatrix} \quad (7.33)$$

$$\mathbf{H}_{4 \times 4} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \quad (7.34)$$

$$\mathbf{H}_{8 \times 8} = \frac{1}{\sqrt{8}} \begin{bmatrix} 1.0 & 1.00 & 1.0 & 1.00 & 1.0 & 1.00 & 1.0 & 1.00 \\ 1.0 & 1.41 & 1.0 & 0.00 & -1.0 & -1.41 & -1.0 & -0.00 \\ 1.0 & 1.00 & -1.0 & -1.00 & 1.0 & 1.00 & -1.0 & -1.00 \\ 1.0 & 0.00 & -1.0 & 1.41 & -1.0 & -0.00 & 1.0 & -1.41 \\ 1.0 & -1.00 & 1.0 & -1.00 & 1.0 & -1.00 & 1.0 & -1.00 \\ 1.0 & -1.41 & 1.0 & -0.00 & -1.0 & 1.41 & -1.0 & 0.00 \\ 1.0 & -1.00 & -1.0 & 1.00 & 1.0 & -1.00 & -1.0 & 1.00 \\ 1.0 & -0.00 & -1.0 & -1.41 & -1.0 & 0.00 & 1.0 & 1.41 \end{bmatrix} \quad (7.35)$$

Note that these matrices are real, orthogonal, and symmetric, $\mathbf{H}^{-1} = \mathbf{H}^T = \mathbf{H} = \overline{\mathbf{H}}$, i.e., they are used for both forward and inverse transforms. The $N = 8$ elements of each of the N row or column vectors can be considered as N samples of the corresponding continuous Hartley CAS functions $cas(2\pi ft) = \cos(2\pi ft) + \sin(2\pi ft)$ (third column of Fig.7.3), as the sum of the corresponding cosine and sine functions (first and second columns of Fig.7.3).

Example 7.1: As considered before, the DFT of a 8-D signal vector $\mathbf{x} = [0, 0, 2, 3, 4, 0, 0, 0]^T$ is (Eq.4.138) $\mathbf{X} = \mathbf{X}_r + j\mathbf{X}_j$ where:

$$\begin{aligned} \mathbf{X}_r &= [3.18, -2.16, 0.71, -0.66, 1.06, -0.66, 0.71, -2.16]^T \\ \mathbf{X}_j &= [0.0, -1.46, 1.06, -0.04, 0.0, 0.04, -1.06, 1.46]^T \end{aligned} \quad (7.36)$$

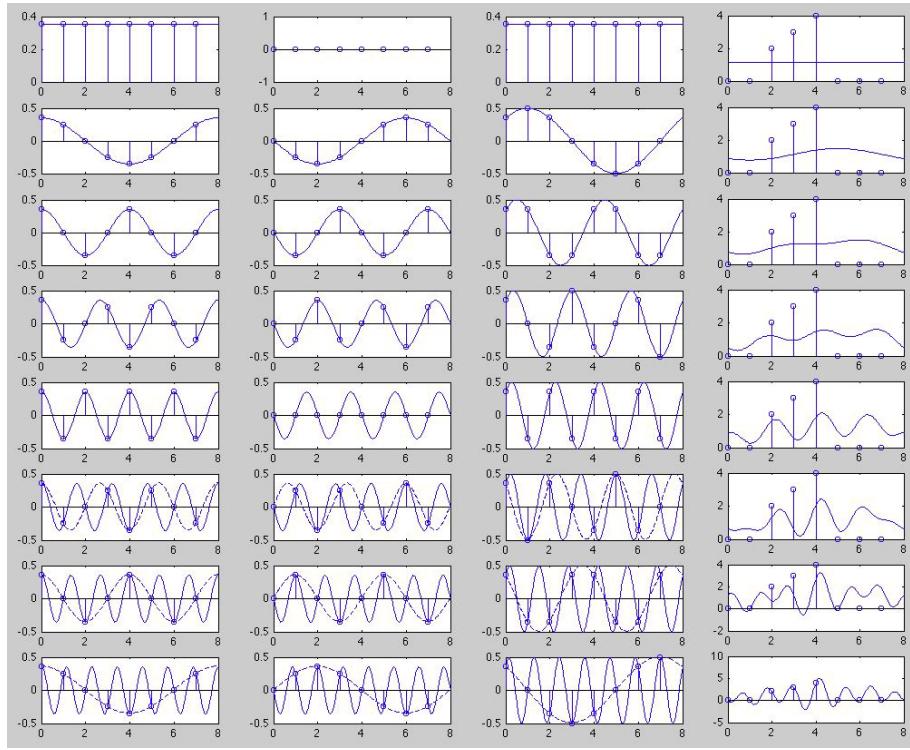


Figure 7.3 Basis functions and vectors of 8-point Hartley transform

The $N=8$ cosine and sine (real and imaginary) parts of the Fourier basis and their sum, the Hartley CAS transform, are shown in the 1st, 2nd and 3rd columns, respectively; the reconstructions of a discrete signal with progressively more and higher frequency components are shown in the 4th columns. The signal is perfectly reconstructed when all N components are included.

The discrete Hartley transform of this signal vector is:

$$\mathbf{X}_H = \mathbf{X}_r - \mathbf{X}_j = [3.18, -0.71, -0.35, -0.62, 1.06, -0.71, 1.77, -3.62]^T \quad (7.37)$$

The original signal can be reconstructed by the inverse Hartley transform as a linear combination of the CAS functions with progressively higher different frequencies, as shown in the right column of Fig.7.3

7.1.5 2-D Hartley Transform

Similar to the 2-D Fourier transform, the 2-D Hartley transform of a signal array $x[m, n]$ ($0 \leq m \leq M - 1, 0 \leq n \leq N - 1$) can be defined as:

$$X[k, l] = \mathcal{H}[x[m, n]] = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m, n] \phi_{k,l}[m, n] \quad (7.38)$$

where $\phi_{k,l}[m, n]$ is a discrete 2-D kernel function. Unlike the 2-D Fourier transform with a unique kernel function $\phi_{k,l}[m, n] = e^{j2\pi(\frac{mk}{M} + \frac{nl}{N})} = e^{j2\pi\frac{mk}{M}} e^{j2\pi\frac{nl}{N}}$, there exist two different versions of the 2-D Hartley transform depending on which of the following kernel function is used:

$$\phi'_{k,l}[m, n] = cas(2\pi(\frac{mk}{M} + \frac{nl}{N})) \quad (7.39)$$

$$\phi''_{k,l}[m, n] = cas(2\pi \frac{mk}{M}) cas(2\pi \frac{nl}{N}) \quad (7.40)$$

Note that, like the Fourier kernel, the second kernel is separable, i.e., it can be written as a product of two 1-D kernels one for each of the two dimensions, while the first one is not. As shown below, these two different kernel functions are very similar to but different from each other:

$$\begin{aligned} & cas(2\pi \frac{mk}{M}) cas(2\pi \frac{nl}{N}) \\ &= [\cos(2\pi \frac{mk}{M}) + \sin(2\pi \frac{mk}{M})] [\cos(2\pi \frac{nl}{N}) + \sin(2\pi \frac{nl}{N})] \\ &= [\cos(2\pi \frac{mk}{M}) \cos(2\pi \frac{nl}{N}) + \sin(2\pi \frac{mk}{M}) \sin(2\pi \frac{nl}{N})] \\ &\quad [\sin(2\pi \frac{mk}{M}) \cos(2\pi \frac{nl}{N}) + \cos(2\pi \frac{mk}{M}) \sin(2\pi \frac{nl}{N})] \\ &= \cos(2\pi(\frac{mk}{M} - \frac{nl}{N})) + \sin(2\pi(\frac{mk}{M} + \frac{nl}{N})) \\ &\neq \cos(2\pi(\frac{mk}{M} + \frac{nl}{N})) + \sin(2\pi(\frac{mk}{M} + \frac{nl}{N})) \\ &= cas(2\pi(\frac{mk}{M} + \frac{nl}{N})) \end{aligned} \quad (7.41)$$

We see that the only difference between the two kernels is the sign of the argument of the cosine function. Both of these kernel functions satisfy the orthogonality:

$$\sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \phi_{k,l}[m, n] \phi_{k',l'}[m, n] = \delta[k - k', l - l'] \quad (7.42)$$

and either of which can be used for the 2-D Hartley transform.

- Based on the inseparable kernel $\phi'_{k,l}[m, n] = cas(2\pi(\frac{mk}{M} + \frac{nl}{N}))$, the forward Hartley transform is carried out following the definition:

$$\begin{aligned} X'_H[k, l] &= \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m, n] cas(2\pi(\frac{mk}{M} + \frac{nl}{N})) \\ &= \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m, n] [\cos(2\pi(\frac{mk}{M} + \frac{nl}{N})) + \sin(2\pi(\frac{mk}{M} + \frac{nl}{N}))] \end{aligned} \quad (7.43)$$

This Hartley transform can be compared with the 2-D Fourier transform:

$$\begin{aligned} X_F[k, l] &= \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m, n] e^{-2\pi(\frac{mk}{M} + \frac{nl}{N})} \\ &= \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m, n] [\cos(2\pi(\frac{mk}{M} + \frac{nl}{N})) - j \sin(2\pi(\frac{mk}{M} + \frac{nl}{N}))] \\ &= X_e[k, l] - j X_o[k, l] = Re[X_e[k, l]] + j Im[X_o[k, l]] \end{aligned} \quad (7.44)$$

where

$$\begin{aligned} X_e[k, l] &= Re[X_F[k, l]] = \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m, n] \cos(2\pi(\frac{mk}{M} + \frac{nl}{N})) \\ X_o[k, l] &= -Im[X_F[k, l]] = \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m, n] \sin(2\pi(\frac{mk}{M} + \frac{nl}{N})) \end{aligned}$$

are respectively the 2-D even and odd components of $X_F[k, l]$. We see the same relationship between the Hartley and Fourier transforms as in 1-D case (Eq.7.12):

$$X'_H[k, l] = X_e[k, l] + X_o[k, l] = Re[X_F[k, l]] - Im[X_F[k, l]] \quad (7.45)$$

Extending the orthogonality in Eq.7.27 from 1-D to 2-D, we get:

$$\begin{aligned} &\frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} cas(2\pi(\frac{mk}{M} + \frac{nl}{N})) cas(2\pi(mk'/M + nl'/N)) \\ &= \delta[k - k', l - l'] \end{aligned} \quad (7.46)$$

Based on this orthogonality and following the same method used to derive Eq.7.29, we get the inverse transform by which the signal can be reconstructed:

$$x[m, n] = \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} X'_H[k, l] cas(2\pi(\frac{mk}{M} + \frac{nl}{N})) \quad (7.47)$$

- Based on the separable kernel $\phi''_{k,l}[m, n] = cas(2\pi mk/M) cas(2\pi nl/N)$, the 2-D Hartley transform can also be carried out in two steps of 1-D transforms each for one of the two dimensions, just as the 2-D Fourier kernel

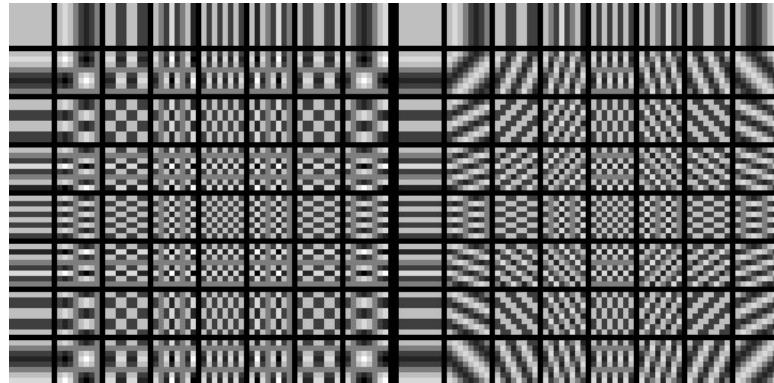


Figure 7.4 The 8 by 8 basis functions for the 2-D Hartley transform

The left half of the image shows the basis functions based on the separable kernel $\phi''_{k,l}[m, n]$, and the right half based on the inseparable kernel $\phi'_{k,l}[m, n]$. The DC component is at the top-left corner, and the highest frequency component in both horizontal and vertical directions is at the middle, same as the 2-D Fourier basis.

$$e^{j2\pi(mk/M+nl/N)}:$$

$$\begin{aligned} X''_H[k, l] &= \frac{1}{\sqrt{MN}} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} x[m, n] \text{cas}(2\pi \frac{mk}{M}) \text{cas}(2\pi \frac{nl}{N}) \\ &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} \left[\frac{1}{\sqrt{M}} \sum_{m=0}^{M-1} x[m, n] \text{cas}(2\pi \frac{mk}{M}) \right] \text{cas}(2\pi \frac{nl}{N}) \end{aligned} \quad (7.48)$$

According to Eq.7.41, this transform can be further written as:

$$\begin{aligned} X''_H[k, l] &= \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m, n] [\cos(2\pi(\frac{mk}{M} - \frac{nl}{N})) + \sin(2\pi(\frac{mk}{M} + \frac{nl}{N}))] \\ &= X_e[k, -l] + X_o[k, l] = \text{Re}[X_F[k, -l]] - \text{Im}[X_F[k, l]] \end{aligned} \quad (7.49)$$

Similarly the inverse transform can also be carried out in two stages

$$\begin{aligned} x[m, n] &= \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} X''_H[k, l] \text{cas}(2\pi \frac{mk}{M}) \text{cas}(2\pi \frac{nl}{N}) \\ &= \frac{1}{\sqrt{N}} \sum_{m=0}^{M-1} \left[\frac{1}{\sqrt{M}} \sum_{n=0}^{N-1} X''_H[k, l] \text{cas}(2\pi \frac{mk}{M}) \right] \text{cas}(2\pi \frac{nl}{N}) \end{aligned} \quad (7.50)$$

The inverse transform in either Eq.7.47 or Eq.7.50 is identical to the forward transform. Also, to better compare the two versions of the 2-D Hartley transform,

we put Eqs. 7.45 and 7.49 side by side:

$$\begin{aligned} X'_H[k, l] &= X_e[k, l] + X_o[k, l] = \operatorname{Re}[X_F[k, l]] - \operatorname{Im}[X_F[k, l]] \\ X''_H[k, l] &= X_e[k, -l] + X_o[k, l] = \operatorname{Re}[X_F[k, -l]] - \operatorname{Im}[X_F[k, l]] \end{aligned}$$

and note that the difference between the two methods is simply the sign of the argument in the even term, it is either $X_e[k, l]$ or $X_e[k, -l] = X_e[-k, l]$ (even). As $X_e[k + M, l + N] = X_e[k, l]$ are periodic, we have $X_e[k, -l] = X_e[k, N - l]$.

Example 7.2: Given a 2-D signal array:

$$\mathbf{x} = \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 70.0 & 80.0 & 90.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 90.0 & 100.0 & 110.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 110.0 & 120.0 & 130.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 130.0 & 140.0 & 150.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix} \quad (7.51)$$

The Hartley spectrum corresponding to inseparable kernel $\phi'_{k,l}[m, n]$ is:

$$\mathbf{X}' = \begin{bmatrix} 165.0 & -10.0 & -45.0 & -32.2 & 55.0 & -10.0 & 65.0 & -187.8 \\ 27.4 & -100.5 & 54.8 & 6.3 & 9.1 & -15.2 & -47.7 & 65.8 \\ 0.0 & 17.1 & -10.0 & -2.9 & 0.0 & 2.9 & 10.0 & -17.1 \\ -26.4 & -15.2 & 17.7 & 7.1 & -8.8 & -0.5 & -20.6 & 46.6 \\ 15.0 & 0.0 & -5.0 & -2.9 & 5.0 & 0.0 & 5.0 & -17.1 \\ -57.4 & 20.2 & 5.3 & 9.2 & -19.1 & 5.5 & -12.3 & 48.7 \\ 30.0 & -17.1 & 0.0 & -2.9 & 10.0 & -2.9 & 0.0 & -17.1 \\ -153.6 & 105.5 & -17.7 & 18.4 & -51.2 & 20.2 & 0.6 & 77.9 \end{bmatrix} \quad (7.52)$$

The Hartley spectrum corresponding to separable kernel $\phi''_{k,l}[m, n]$ is:

$$\mathbf{X}'' = \begin{bmatrix} 165.0 & -10.0 & -45.0 & -32.2 & 55.0 & -10.0 & 65.0 & -187.8 \\ 27.4 & -3.5 & -5.6 & -5.4 & 9.1 & -3.5 & 12.7 & -31.2 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ -26.4 & 1.5 & 7.3 & 5.2 & -8.8 & 1.5 & -10.3 & 30.0 \\ 15.0 & 0.0 & -5.0 & -2.9 & 5.0 & 0.0 & 5.0 & -17.1 \\ -57.4 & 3.5 & 15.6 & 11.2 & -19.1 & 3.5 & -22.7 & 65.4 \\ 30.0 & 0.0 & -10.0 & -5.9 & 10.0 & 0.0 & 10.0 & -34.1 \\ -153.6 & 8.5 & 42.7 & 30.0 & -51.2 & 8.5 & -59.8 & 174.9 \end{bmatrix} \quad (7.53)$$

In either case, the signal is perfectly reconstructed by the inverse transform (identical to the forward transform) corresponding to each of the two kernels.

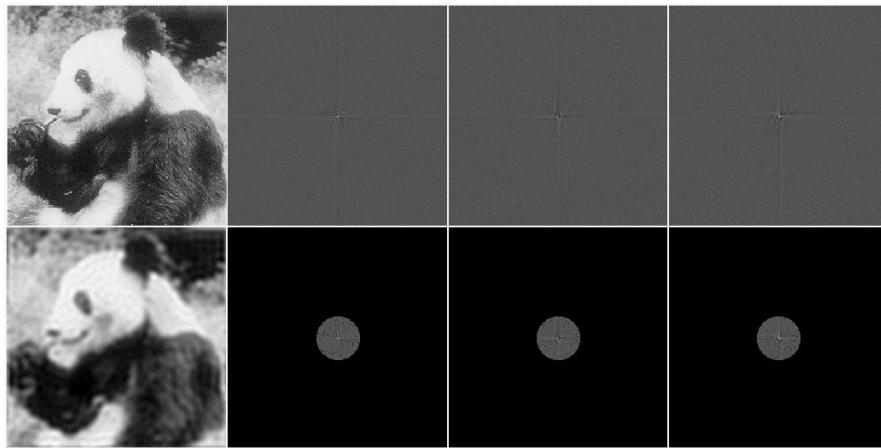


Figure 7.5 The Hartley and Fourier filtering of an image

The image and its Fourier and Hartley spectra before and after a low-pass filtering are shown in the top and bottom rows, respectively. The second and third panel of each row are the real and imaginary parts of the Fourier spectrum, while the forth panel is for the Hartley spectrum.

Example 7.3: An image and both of its Fourier and Hartley spectra are shown in the top row of Fig.7.5. The real and imaginary parts of the Fourier spectrum are shown respectively in the second and third panels, and the Hartley spectrum is shown in the forth. These spectra are then low-pass filtered and then inverse transformed as shown in the bottom row of the figure. The Hartley filtering effect is identical to that of the Fourier filtering, shown in the first panel of the bottom row.

7.2 The Discrete Sine and Cosine Transforms

Same as the Hartley transform, both the sine and cosine transforms, also derived from the Fourier transform, convert a real signal into its real spectrum. Moreover, their discrete versions, the discrete sine and cosine transforms (DST and DCT, respectively), can also be carried out based on the fast Fourier transform.

7.2.1 The Continuous Cosine and Sine Transforms

We first consider the Fourier transform of a real signal $x(t) = \bar{x}(t)$:

$$\begin{aligned} X(f) &= \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt = \int_{-\infty}^{\infty} x(t)[\cos(2\pi ft) - j\sin(2\pi ft)]dt \\ &= X_r(f) - jX_j(f) \end{aligned} \quad (7.54)$$

where the real part of the spectrum $X_r(f)$ is even and the imaginary part $X_j(f)$ is odd:

$$\begin{aligned} X_r(f) &= \int_{-\infty}^{\infty} x(t)\cos(2\pi ft)dt = X_r(-f) \\ X_j(f) &= \int_{-\infty}^{\infty} x(t)\sin(2\pi ft)dt = -X_j(-f) \end{aligned} \quad (7.55)$$

We further assume the signal $x(t)$ is either even or odd:

- If $x(t) = x(-t)$ is even, then the integrand $x(t)\sin(2\pi ft)$ in Eq.7.54 is odd with respect to t and $X_j(f) = 0$, but $x(t)\cos(2\pi ft)$ is even and the equation above becomes:

$$X(f) = \int_{-\infty}^{\infty} x(t)\cos(2\pi ft)dt = 2 \int_0^{\infty} x(t)\cos(2\pi ft)dt = X(-f) \quad (7.56)$$

This spectrum $X(f)$ is real and even with respect to f . The inverse transform becomes:

$$\begin{aligned} x(t) &= \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df = \int_{-\infty}^{\infty} X(f)\cos(2\pi ft)df + j \int_{-\infty}^{\infty} X(f)\sin(2\pi ft)df \\ &= 2 \int_0^{\infty} X(f)\cos(2\pi ft)df \end{aligned} \quad (7.57)$$

due to the fact that $X(f)\sin(2\pi ft)$ is odd with respect to f , and the second term is zero, but $X(f)\cos(2\pi ft)$ is even. Now we get the cosine transform pair of an even signal $x(t)$:

$$\begin{aligned} X_c(f) &= 2 \int_0^{\infty} x(t)\cos(2\pi ft)dt \\ x(t) &= 2 \int_0^{\infty} X_c(f)\cos(2\pi ft)df \end{aligned} \quad (7.58)$$

- If $x(t) = -x(-t)$ is odd, then the integrands $x(t)\cos(2\pi ft)$ in Eq.7.54 is odd with respect to t and $X_r(f) = 0$, but $x(t)\sin(2\pi ft)$ is even and the equation becomes:

$$X(f) = -j \int_{-\infty}^{\infty} x(t)\sin(2\pi ft)dt = -j2 \int_0^{\infty} x(t)\sin(2\pi ft)dt = -X(-f) \quad (7.59)$$

This spectrum $X(f)$ is imaginary and odd with respect to f . The inverse transform becomes:

$$\begin{aligned} x(t) &= \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df = \int_{-\infty}^{\infty} X(f)\cos(2\pi ft)df + j \int_{-\infty}^{\infty} X(f)\sin(2\pi ft)df \\ &= 2j \int_0^{\infty} X(f)\sin(2\pi ft)df \end{aligned} \quad (7.60)$$

due to the fact that $X(f)\cos(2\pi ft)$ is odd with respect to f and the first term is zero, but $X_S(f)\sin(2\pi ft)$ is even. Defining the sine transform as $X_S(f) = jX(f)$ we get the sine transform pair of an odd signal $x(t)$:

$$\begin{aligned} X_s(f) &= 2 \int_0^{\infty} x(t)\sin(2\pi ft)dt \\ x(t) &= 2 \int_0^{\infty} X_s(f)\sin(2\pi ft)df \end{aligned} \quad (7.61)$$

We see that similar to the Fourier transform, both the sine and cosine transforms also represent a real signal as a linear combination of a set of uncountably infinite basis sinusoids of different frequencies. However, different from the Fourier transform, as here the weighting functions $X_C(f)$ and $X_S(f)$ are real, they only represent the amplitudes of the basis functions but not their phases, which are always zero.

The cosine and sine transforms above are valid only if the signal in question is either even or odd. If the signal is neither even nor odd, but it is known to be zero before $t = 0$, i.e., $x(t) = x(t)u(t)$, then the following even and odd functions can be constructed:

$$x'_e(t) = \begin{cases} x(t) & t \geq 0 \\ x(-t) & t \leq 0 \end{cases} \quad x'_o(t) = \begin{cases} x(t) & t > 0 \\ 0 & t = 0 \\ -x(-t) & t < 0 \end{cases} \quad (7.62)$$

so that the cosine and sine transforms can still be applied. Note that $x'_o(0)$ is defined as zero for $x'_o(t)$ to be odd.

7.2.2 From the DFT to the DCT and DST

The consideration above for continuous signals can be extended to discrete signals of a finite duration. The corresponding cosine and sine transforms are called the discrete cosine transform (DCT) and discrete sine transform (DST). However, different from the continuous case, here in the discrete case there is more than one way to construct an even or odd signal based on a set of finite data samples $x[0], \dots, x[N-1]$. For example, by assuming $x[-n] = x[n]$, we can obtain a sequence of $2N-1$ samples that is even with respect to $n=0$. Alternatively, we can also let $x[-n] = x[n-1]$, i.e., $x[-1] = x[0]$, $x[-2] = x[1]$, etc., and $x[-N] = x[N-1]$ to get a sequence of $2N$ samples that is even with respect to $n = -1/2$. Moreover, there may be different ways to assume the periodicity beyond these $2N-1$ or $2N$ data samples. In the following, we will take the

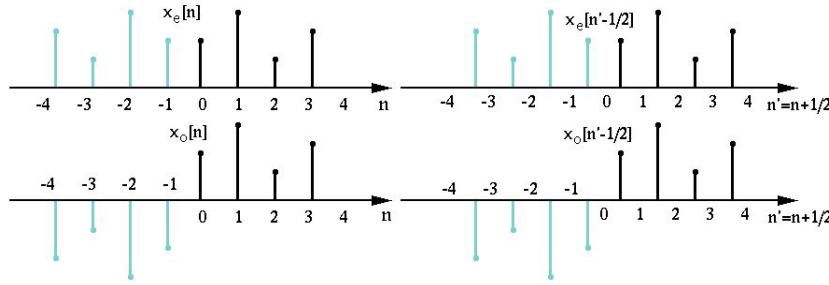


Figure 7.6 Construction of even (top) and odd signals (bottom)

Given an $N=4$ point signal $x[0], \dots, x[3]$ (black), the even and odd versions can be constructed by including N additional points $x[-1] = \pm x[0], \dots, x[-4] = \pm x[3]$ (gray). This signal of $2N = 8$ points is symmetric with respect to $n = 1/2$. If we define $n' = n + 1/2$ then $x[n] = x[n' - 1/2]$ (from $-N + 1/2 = -3.5$ to $N - 1/2 = 3.5$) are symmetric with respect to $n' = 0$.

second approach to construct a sequence of $2N$ points and assume it is periodic beyond its two ends. Then the DCT and DST can be derived by applying the DFT to this sequence of $2N$ points.

Given an N -point real signal sequence $x[0], \dots, x[N - 1]$, we construct two sequences of $2N$ points:

$$x_e[n] = \begin{cases} x[n] & (0 \leq n \leq N - 1) \\ x[-n - 1] & (-N \leq n \leq -1) \end{cases} \quad (7.63)$$

and

$$x_o[n] = \begin{cases} x[n] & (0 \leq n \leq N - 1) \\ -x[-n - 1] & (-N \leq n \leq -1) \end{cases} \quad (7.64)$$

which are respectively even and odd with respect to $n = -1/2$, as shown in Fig.7.6. If we shift them to the right by $1/2$, or, equivalently, define a new index $n' = n + 1/2$, i.e., $n = n' - 1/2$, then $x_e[n] = x_e[n' - 1/2]$ and $x_o[n] = x_o[n' - 1/2]$ are respectively even and odd with respect to $n' = 0$. These $2N$ -point sequences are further assumed to repeat itself outside the range $-N \leq n \leq N - 1$, i.e., it is periodic with period $2N$:

$$\begin{aligned} x_e[n] &= x_e[n + 2N] = x_e[-n - 1] = x_e[2N - n - 1] \\ x_o[n] &= x_o[n + 2N] = -x_o[-n - 1] = -x_o[2N - n - 1] \end{aligned} \quad (7.65)$$

Applying the $2N$ -point DFT to this constructed signal of $2N$ points, now simply denoted by $x[n]$, we get:

$$\begin{aligned} X[k] &= \frac{1}{\sqrt{2N}} \sum_{n'=-N+1/2}^{N-1/2} x \left[n' - \frac{1}{2} \right] e^{-j2\pi n' k / 2N} \\ &= \frac{1}{\sqrt{2N}} \sum_{n'=-N+1/2}^{N-1/2} x \left[n' - \frac{1}{2} \right] \cos \left(\frac{2\pi n' k}{2N} \right) \\ &\quad - \frac{j}{\sqrt{2N}} \sum_{n'=-N+1/2}^{N-1/2} x \left[n' - \frac{1}{2} \right] \sin \left(\frac{2\pi n' k}{2N} \right), \quad (k = 0, \dots, 2N-1) \end{aligned} \quad (7.66)$$

Note that $\cos(2\pi n' k / 2N)$ and $\sin(2\pi n' k / 2N)$ are respectively even and odd with respect to $n' = 0$, and $x[n' - 1/2]$ is also either even or odd symmetric to $n' = 0$. We consider the following two cases:

- **Discrete Cosine Transform (DCT)**

If $x[n] = x_e[n]$ is real and even, all $2N$ terms in the first summation of Eq.7.66 are even and their sum is equal to twice the sum of half of the terms, while all $2N$ terms in the second summation are odd and their sum is zero, and we have:

$$X[k] = \sqrt{\frac{2}{N}} \sum_{n'=1/2}^{N-1/2} x \left[n' - \frac{1}{2} \right] \cos \left(\frac{2\pi n' k}{2N} \right) \quad (k = 0, \dots, 2N-1) \quad (7.67)$$

Note that $X_c[n] = X_c[-n]$ is real, even (with respect to $n = 0$), and of period $2N$. Specifically, we have $X[N+k] = X[N+k-2N] = X[-N+k] = X[N-k]$, indicating that a point $X[N+k]$ in the second half of the $2N$ coefficients is equal to a corresponding point $X[N-n]$ in the first half, for all $n = 0, 1, \dots, N-1$. In other words, the range for the index k of $X[k]$ above can be from 0 to $N-1$, as the second half is redundant and can therefore be dropped. Finally, replacing n' by $n + 1/2$, we get the discrete cosine transform (DCT):

$$X_c[k] = \sqrt{\frac{2}{N}} \sum_{n=0}^{N-1} x[n] \cos \left(\frac{(2n+1)k\pi}{2N} \right), \quad (k = 0, \dots, N-1) \quad (7.68)$$

Here $X_c[k]$ is the nth DCT coefficient corresponding to the frequency component $\cos((2n+1)k\pi/2N)$. In particular, when $k = 0$, $X_c[0]$ is proportional to $\sum_{n=0}^{N-1} x[n]$ representing the DC component of the signal.

- **Discrete Sine Transform (DST)**

If $x[n] = x_o[n]$ is real and odd, the $2N$ terms in the first summation of Eq.7.66 are odd and their sum is zero, while all $2N$ terms in the second summation are even and their sum is equal to twice the sum of the half of the terms. And,

following Eq.7.68, we get the discrete sine transform (DST):

$$X_s[k] = \sqrt{\frac{2}{N}} \sum_{n=0}^{N-1} x[n] \sin\left(\frac{(2n+1)k\pi}{2N}\right), \quad (k = 0, \dots, N-1) \quad (7.69)$$

As before, here the spectrum $X_s[k]$ has been redefined to include j . Note that $X_s[k] = -X_s[-k]$ is real, odd, and with period $2N$. Also note that $X_s[0] = 0$, independent of the signal (unlike $X_c[0]$ of DCT is for the DC). The DST above further is modified by replacing k by $k+1$ to exclude the first term of constant zero and include one more term at the end:

$$X_s[k] = \sqrt{\frac{2}{N}} \sum_{n=0}^{N-1} x[n] \sin\left(\frac{(2n+1)(k+1)\pi}{2N}\right), \quad (k = 0, \dots, N-1) \quad (7.70)$$

However, note that due to this modification, the zeroth DST basis function becomes $\sin((2n+1)/2N)$, which is a function of the time index n (instead of a constant for the zeroth basis function of the DCT or any other transform). In other words, different from all other orthogonal transforms, the zeroth frequency component $X_s[0]$ of the DST does not represent the DC component or average of the signal.

Comparing the DCT and DST defined above with the DFT considered in Chapter 4 we see the following advantages:

- The DCT and DST are both real transforms without any complex operations needed by the complex DFT.
- The k th DCT coefficient $X_c[k]$ (Eq.7.68) represents a sinusoid of frequency $k/2N$ and the k th DST coefficient $X_s[k]$ (Eq.7.70) represents a sinusoid of frequency $(k+1)/2N$, both of which are half of the frequency k/N represented by the k th DFT coefficient $X_F[k]$.
- The highest frequencies represented by the DCT coefficient $X_c[N-1]$ and DST coefficient $X_s[N-1]$ are approximately the same: $f_{max} = 1/2$ with period $T = 1/f_{max} = 2$, also the same as the highest frequency represented by the DFT Eq.4.168, i.e., all three transforms cover the same frequency range.
- The frequency resolution of DCT and DST is twice that of the DFT, as N frequencies are represented by the DCT/DST (one by each coefficient), but only $N/2$ frequencies are represented by the DFT (one by a pair of two coefficients). as can be seen by comparing Fig.7.7 for the DCT and DST with Fig.4.16 for the DFT.
- To perform the Fourier transform on a physical signal, it needs to be truncated to have a finite duration $0 \leq t \leq T$, and then assumed to be periodic beyond T (Figs.3.1 and 4.13). In this process, discontinuity is likely to be introduced that would cause certain artifact high frequencies. However, in the case of the DCT, as an even symmetry is assumed while truncating and imposing periodicity on the time signal, no discontinuity is introduced and all related

artifacts are avoided. However, this does not apply to the DST, which, like the DFT, also introduces discontinuities.

7.2.3 Matrix Forms of DCT and DST

The DCT matrix \mathbf{C} and DST matrix \mathbf{S} can be constructed respectively by a set of N orthogonal column vectors:

$$\mathbf{C} = [\mathbf{c}_0 \cdots \mathbf{c}_{N-1}] = \begin{bmatrix} c[0,0] & \cdots & c[0,N-1] \\ \vdots & \ddots & \vdots \\ c[0,N-1] & \cdots & c[N-1,N-1] \end{bmatrix} \quad (7.71)$$

and

$$\mathbf{S} = [\mathbf{s}_0 \cdots \mathbf{s}_{N-1}] = \begin{bmatrix} s[0,0] & \cdots & s[0,N-1] \\ \vdots & \ddots & \vdots \\ s[0,N-1] & \cdots & s[N-1,N-1] \end{bmatrix} \quad (7.72)$$

where $c[n, k]$ and $s[n, k]$ are the elements in nth row and kth column of \mathbf{C} and \mathbf{S} ($n, k = 0, 1, \dots, N-1$), respectively:

$$c[n, k] = \cos\left(\frac{(2n+1)k\pi}{2N}\right), \quad s[n, k] = \sin\left(\frac{(2n+1)(k+1)\pi}{2N}\right) \quad (7.73)$$

and \mathbf{c}_k and \mathbf{s}_k are the kth columns of \mathbf{C} and \mathbf{S} , respectively:

$$\mathbf{c}_k = \left[\cos\left(\frac{k\pi}{2N}\right), \cos\left(\frac{3k\pi}{2N}\right), \cos\left(\frac{5k\pi}{2N}\right), \dots, \cos\left(\frac{(2N-1)k\pi}{2N}\right) \right]^T \quad (7.74)$$

and

$$\mathbf{s}_k = \left[\sin\left(\frac{(k+1)\pi}{2N}\right), \sin\left(\frac{3(k+1)\pi}{2N}\right), \dots, \sin\left(\frac{(2N-1)(k+1)\pi}{2N}\right) \right]^T \quad (7.75)$$

representing respectively a sinusoid of frequency $k/2N$ and $(k+1)/2N$.

We now show that the column vectors of both matrices \mathbf{C} and \mathbf{S} are orthogonal. To do so, we need the following identity (to be proved as a homework problem):

$$\sum_{n=0}^{N-1} \cos\left(\frac{(2n+1)k\pi}{2N}\right) = \begin{cases} N, & k = 0 \\ -N, & k = 2N \\ 0, & \text{else} \end{cases}, \quad (0 \leq n \leq 2N) \quad (7.76)$$

Given this identify, we are ready to show the following orthogonality:

$$\langle \mathbf{c}_k, \mathbf{c}_l \rangle = 0 \quad (7.77)$$

$$\langle \mathbf{s}_k, \mathbf{s}_l \rangle = 0, \quad (7.78)$$

$$(k, l = 0, \dots, N-1, \quad l \neq k)$$

First we consider Eq.7.77:

$$\begin{aligned} \langle \mathbf{c}_k, \mathbf{c}_l \rangle &= \sum_{n=0}^{N-1} c[n, k] c[n, l] = \sum_{n=0}^{N-1} \cos\left(\frac{(2n+1)k\pi}{2N}\right) \cos\left(\frac{(2n+1)l\pi}{2N}\right) \\ &= \frac{1}{2} \sum_{n=0}^{N-1} \cos\left(\frac{(2n+1)(k-l)\pi}{2N}\right) + \frac{1}{2} \sum_{n=0}^{N-1} \cos\left(\frac{(2n+1)(k+l)\pi}{2N}\right) \end{aligned} \quad (7.79)$$

Here we have used the identity $2 \cos \alpha \cos \beta = \cos(\alpha + \beta) + \cos(\alpha - \beta)$. When $l \neq k$, both terms are zero according to Eq. 7.76, i.e., the column vectors of \mathbf{C} are indeed orthogonal. When $l = k$, the inner product becomes:

$$\langle \mathbf{c}_k, \mathbf{c}_k \rangle = \frac{N}{2} + \frac{1}{2} \sum_{n=0}^{N-1} \cos\left(\frac{(2n+1)2k\pi}{2N}\right) = \begin{cases} N & k = 0 \\ N/2 & \text{else} \end{cases} \quad (7.80)$$

as the second term is either $N/2$ if $k = 0$ or zero if $k \neq 0$. In order for all N columns of \mathbf{C} to be normalized, we introduce a scaling factor $a[k]$ defined as:

$$a[k] = \begin{cases} \sqrt{1/N} & k = 0 \\ \sqrt{2/N} & k \neq 0 \end{cases} \quad (7.81)$$

so that all columns of the modified version of the DCT matrix, still denoted by \mathbf{C} , are orthonormal: $\langle \mathbf{c}_k, \mathbf{c}_l \rangle = \delta[k - l]$.

Next we consider Eq.7.78:

$$\begin{aligned} \langle \mathbf{s}_k, \mathbf{s}_l \rangle &= \sum_{n=0}^{N-1} s[n, k] s[n, l] \\ &= \sum_{n=0}^{N-1} \sin\left(\frac{(2n+1)(k+1)\pi}{2N}\right) \sin\left(\frac{(2n+1)(l+1)\pi}{2N}\right) \\ &= \frac{1}{2} \sum_{n=0}^{N-1} \cos\left(\frac{(2n+1)(k-l)\pi}{2N}\right) - \frac{1}{2} \sum_{n=0}^{N-1} \cos\left(\frac{(2n+1)(k+l+2)\pi}{2N}\right) \end{aligned} \quad (7.82)$$

Here we have used the identity $2 \sin \alpha \sin \beta = \cos(\alpha - \beta) - \cos(\alpha + \beta)$. When $l \neq k$, both terms are zero according to Eq. 7.76, i.e., the column vectors of \mathbf{S} are orthogonal. When $l = k$, the inner product becomes:

$$\langle \mathbf{s}_k, \mathbf{s}_k \rangle = \frac{N}{2} - \frac{1}{2} \sum_{n=0}^{N-1} \cos\left(\frac{(2n+1)(2k+2)\pi}{2N}\right) = \begin{cases} N & k = N - 1 \\ N/2 & \text{else} \end{cases} \quad (7.83)$$

as the second term is either $-N/2$ if $k = N - 1$ or zero otherwise. In order for all N columns of \mathbf{S} to be normalized, we introduce a scaling factor $b[k]$ defined as:

$$b[k] = \begin{cases} \sqrt{2/N} & k \neq N - 1 \\ \sqrt{1/N} & k = N - 1 \end{cases} \quad (7.84)$$

so that the columns of the modified version of the DST matrix, still denoted by \mathbf{S} , are orthonormal: $\langle \mathbf{s}_k, \mathbf{s}_l \rangle = \delta[k - l]$.¹

As now both \mathbf{C} and \mathbf{S} are orthonormal, i.e., $\mathbf{C}^T \mathbf{C} = \mathbf{S}^T \mathbf{S} = \mathbf{I}$, they can be used to define the DCT and DST. Given any N-D signal vector $\mathbf{x} = [x[0], \dots, x[N-1]]^T$, its DCT and DST coefficients can be found simply by matrix multiplication:

$$\mathbf{X}_c = \mathbf{C}^T \mathbf{x} = \begin{bmatrix} \mathbf{c}_0^T \\ \vdots \\ \mathbf{c}_{N-1}^T \end{bmatrix} \mathbf{x}, \quad \mathbf{X}_s = \mathbf{S}^T \mathbf{x} = \begin{bmatrix} \mathbf{s}_0^T \\ \vdots \\ \mathbf{s}_{N-1}^T \end{bmatrix} \mathbf{x} \quad (7.85)$$

The k th components $X_c[k]$ of \mathbf{X}_c and $X_s[k]$ of \mathbf{X}_s are respectively the projections of \mathbf{x} onto the k th basis vectors \mathbf{c}_k and \mathbf{s}_k ($k = 0, \dots, N-1$):

$$X_c[k] = \langle \mathbf{x}, \mathbf{c}_k \rangle = \mathbf{x}^T \mathbf{c}_k = a[k] \sum_{n=0}^{N-1} x[n] \cos\left(\frac{(2n+1)k\pi}{2N}\right) \quad (7.86)$$

$$X_s[k] = \langle \mathbf{x}, \mathbf{s}_k \rangle = \mathbf{x}^T \mathbf{s}_k = b[k] \sum_{n=0}^{N-1} x[n] \sin\left(\frac{(2n+1)(k+1)\pi}{2N}\right) \quad (7.87)$$

These are Eqs.7.68 and 7.70 respectively with a scaling factor $a[k] = b[k] = \sqrt{2/N}$, except $a[0] = b[N-1] = 1/\sqrt{N}$, i.e.,

$$a[N-1-k] = b[k], \quad (k = 0, \dots, N-1) \quad (7.88)$$

The signal vector \mathbf{x} can be reconstructed by the inverse DCT or DST as a linear combination of the corresponding basis:

$$\begin{aligned} \mathbf{x} &= \mathbf{C} \mathbf{X}_c = [\mathbf{c}_0, \dots, \mathbf{c}_{N-1}] \begin{bmatrix} X_c[0] \\ \vdots \\ X_c[N-1] \end{bmatrix} = \sum_{k=0}^{N-1} X_c[k] \mathbf{c}_k \\ &= \mathbf{S} \mathbf{X}_s = [\mathbf{s}_0, \dots, \mathbf{s}_{N-1}] \begin{bmatrix} X_s[0] \\ \vdots \\ X_s[N-1] \end{bmatrix} = \sum_{k=0}^{N-1} X_s[k] \mathbf{s}_k \end{aligned} \quad (7.89)$$

In component form the n th component $x[n]$ ($n = 0, \dots, N-1$) can be found as:

$$x[n] = \sum_{k=0}^{N-1} c[n, k] X_c[k] = \sum_{k=0}^{N-1} X_c[k] a[k] \cos\left(\frac{(2n+1)k\pi}{2N}\right) \quad (7.90)$$

$$= \sum_{k=0}^{N-1} s[n, k] X_s[k] = \sum_{k=0}^{N-1} X_s[k] b[k] \sin\left(\frac{(2n+1)(k+1)\pi}{2N}\right) \quad (7.91)$$

¹ In Matlab the DST and IDST are defined differently. Also for Parseval's identity to hold, they need to be rescaled: $X = dst(x)/sqrt((length(x)+1)/2)$ and $x = idst(X) * sqrt((length(x)+1)/2)$.

We list below the DCT and DST matrices for $N = 2$, $N = 4$ and $N = 8$ as some specific examples. Here we use $n = \log_2 N$ as the subscript for the corresponding N -point transform matrices \mathbf{C}_n and \mathbf{S}_n , in consistence with the notation used in the following chapter.

$$\mathbf{C}_1 = \mathbf{S}_1 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (7.92)$$

This matrix is composed of two row vectors $\mathbf{c}_0^T = [1 \ 1]/\sqrt{2}$ and $\mathbf{c}_1^T = [1 \ -1]/\sqrt{2}$ and is identical to the 2-point DFT matrix considered previously. The DCT of a 2-point signal $\mathbf{x} = [x[0], x[1]]^T$ is

$$\mathbf{X} = \begin{bmatrix} X[0] \\ X[1] \end{bmatrix} = 0.707 \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \end{bmatrix} = 0.707 \begin{bmatrix} x[0] + x[1] \\ x[1] - x[0] \end{bmatrix} \quad (7.93)$$

The first component $X[0]$ is proportional to the sum $x[0] + x[1]$ of the two samples representing the average or DC component of the signal, and the second component $X[1]$ is proportional to the difference $x[0] - x[1]$ between the two samples. This is also the case for the DFT, as well as all orthogonal transforms when $N = 2$ (as we will see later). When $N = 4$, we have:

$$\mathbf{C}_2^T = \begin{bmatrix} 0.50 & 0.50 & 0.50 & 0.50 \\ 0.65 & 0.27 & -0.27 & -0.65 \\ 0.50 & -0.50 & -0.50 & 0.50 \\ 0.27 & -0.65 & 0.65 & -0.27 \end{bmatrix}, \quad \mathbf{S}_2^T = \begin{bmatrix} 0.27 & 0.65 & 0.65 & 0.27 \\ 0.50 & 0.50 & -0.50 & -0.50 \\ 0.65 & -0.27 & -0.27 & 0.65 \\ 0.50 & -0.50 & 0.50 & -0.50 \end{bmatrix} \quad (7.94)$$

When $N = 8$, we have:

$$\mathbf{C}_3^T = \begin{bmatrix} 0.35 & 0.35 & 0.35 & 0.35 & 0.35 & 0.35 & 0.35 & 0.35 \\ 0.49 & 0.42 & 0.28 & 0.10 & -0.10 & -0.28 & -0.42 & -0.49 \\ 0.46 & 0.19 & -0.19 & -0.46 & -0.46 & -0.19 & 0.19 & 0.46 \\ 0.42 & -0.10 & -0.49 & -0.28 & 0.28 & 0.49 & 0.10 & -0.42 \\ 0.35 & -0.35 & -0.35 & 0.35 & 0.35 & -0.35 & -0.35 & 0.35 \\ 0.28 & -0.49 & 0.10 & 0.42 & -0.42 & -0.10 & 0.49 & -0.28 \\ 0.19 & -0.46 & 0.46 & -0.19 & -0.19 & 0.46 & -0.46 & 0.19 \\ 0.10 & -0.28 & 0.42 & -0.49 & 0.49 & -0.42 & 0.28 & -0.10 \end{bmatrix} \quad (7.95)$$

$$\mathbf{S}_3^T = \begin{bmatrix} 0.10 & 0.28 & 0.42 & 0.49 & 0.49 & 0.42 & 0.28 & 0.10 \\ 0.19 & 0.46 & 0.46 & 0.19 & -0.19 & -0.46 & -0.46 & -0.19 \\ 0.28 & 0.49 & 0.10 & -0.42 & -0.42 & 0.10 & 0.49 & 0.28 \\ 0.35 & 0.35 & -0.35 & -0.35 & 0.35 & 0.35 & -0.35 & -0.35 \\ 0.42 & 0.10 & -0.49 & 0.28 & 0.28 & -0.49 & 0.10 & 0.42 \\ 0.46 & -0.19 & -0.19 & 0.46 & -0.46 & 0.19 & 0.19 & -0.46 \\ 0.49 & -0.42 & 0.28 & -0.10 & -0.10 & 0.28 & -0.42 & 0.49 \\ 0.35 & -0.35 & 0.35 & -0.35 & 0.35 & -0.35 & 0.35 & -0.35 \end{bmatrix} \quad (7.96)$$

The column vectors \mathbf{c}_k and \mathbf{s}_k ($k = 0, \dots, N-1$) of \mathbf{C} and \mathbf{S} (row vectors of \mathbf{C}^T and \mathbf{S}^T) form an orthonormal basis of space \mathbb{R}^N . The N elements

of each of the N vectors can also be considered as N samples of the corresponding continuous cosine function $a[k] \cos((2t + 1)k\pi)/2N)$ or sine functions $b[k] \sin((2t + 1)(k + 1)\pi/2N)$ with progressively higher frequencies, as shown in the first two columns in Fig.7.7.

Also note that the elements of the first row of \mathbf{S}^T for the DST are not the same, unlike either \mathbf{W} for the DFT or \mathbf{C} for the DCT (or any of the orthogonal transforms to be considered later). In other words, the DC component of the signal is not represented by the DST.

Example 7.4: The DCT and DST coefficients of an $N=8$ point signal $\mathbf{x} = [0, 0, 2, 3, 4, 0, 0, 0]^T$ can be found by a matrix multiplication:

$$\begin{aligned}\mathbf{X}_c &= \mathbf{C}^T \mathbf{x} = [3.18, 0.46, -3.62, -0.70, 1.77, -0.22, -0.42, 1.32]^T \\ \mathbf{X}_s &= \mathbf{S}^T \mathbf{x} = [4.26, 0.73, -2.72, -0.35, 0.96, -0.84, -0.13, 1.06]^T.\end{aligned}\quad (7.97)$$

The interpretation of these DCT and DST coefficients is much more straightforward than that of the DFT. $X[0]$ represents the DC component or the average of the signal, while the subsequent coefficients $X[k]$ ($k = 1, \dots, N - 1$) represent the magnitudes of progressively high frequency components contained in the signal.

The signal is perfectly reconstructed by the inverse DCT or DST as a linear combination of the column vectors of $\mathbf{C} = [\mathbf{c}_0, \dots, \mathbf{c}_{N-1}]$ or $\mathbf{S} = [\mathbf{s}_0, \dots, \mathbf{s}_{N-1}]$ as the basis spanning \mathbb{R}^8 :

$$\mathbf{x} = \mathbf{C} \mathbf{X}_c = \sum_{k=0}^7 X_c[k] \mathbf{c}_k = \mathbf{S} \mathbf{X}_s = \sum_{k=0}^7 X_s[k] \mathbf{s}_k \quad (7.98)$$

The reconstruction of the signal by the linear combination of the eight vectors is shown in Fig. 7.7

7.2.4

Fast Algorithms for DCT and DST

The computational complexity of both DCT and DST is $O(N^2)$ if implemented as a matrix multiplication ($O(N)$ for each of the N coefficients $X[k]$). However, as the DCT and DST are closely related to the DFT, they can be implemented by the fast Fourier transform FFT algorithm with complexity $O(N \log_2 N)$. We will first consider the fast algorithm for the DCT and then show that the DST can be carried out based on the fast algorithm for DCT.

We first define a new sequence $y[0], \dots, y[N - 1]$ based on the given signal $x[0], \dots, x[N - 1]$:

$$\begin{cases} y[n] = x[2n] \\ y[N - 1 - n] = x[2n + 1] \end{cases} \quad (n = 0, \dots, N/2 - 1) \quad (7.99)$$

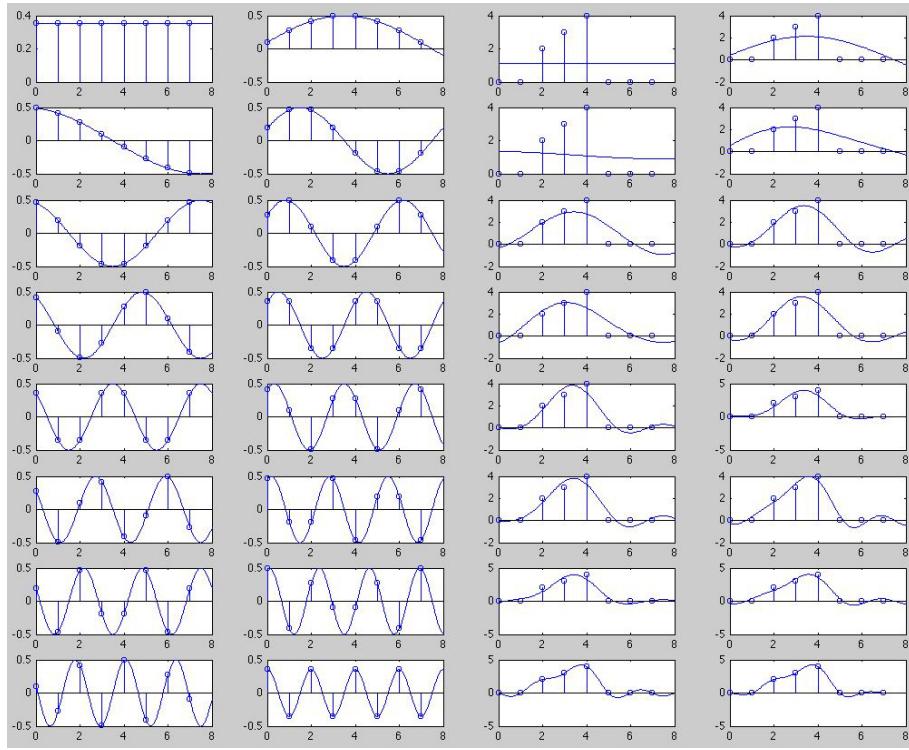


Figure 7.7 Basis functions and vectors of 8-point DCT and DST

The $N=8$ continuous and discrete basis functions of DCT and DST are shown respectively in the 1st and 2nd columns; the reconstructions of a discrete signal with progressively more and higher DCT and DST frequency components are shown respectively in the 3rd and 4th columns (Eq.7.97). The signal is perfectly reconstructed when all N components are used.

Note that the first half of $y[n]$ contains all even components of $x[n]$, while the second half of $y[n]$ contains all odd ones but in reverse order. The N -point DCT of the given signal $x[n]$ now becomes:

$$\begin{aligned}
 X[k] &= a[k] \sum_{n=0}^{N-1} x[n] \cos\left(\frac{(2n+1)k\pi}{2N}\right) \\
 &= a[k] \sum_{n=0}^{N/2-1} x[2n] \cos\left(\frac{(4n+1)k\pi}{2N}\right) + a[k] \sum_{n=0}^{N/2-1} x[2n+1] \cos\left(\frac{(4n+3)k\pi}{2N}\right) \\
 &= a[k] \sum_{n=0}^{N/2-1} y[n] \cos\left(\frac{(4n+1)k\pi}{2N}\right) + a[k] \sum_{n=0}^{N/2-1} y[N-1-n] \cos\left(\frac{(4n+3)k\pi}{2N}\right)
 \end{aligned} \tag{7.100}$$

Here the first summation is for all even terms and second all odd terms. We define $n' = N - 1 - n$ and rewrite the second summation as:

$$a[k] \sum_{n'=N/2}^{N-1} y[n'] \cos\left(2k\pi - \frac{(4n'+1)k\pi}{2N}\right) = a[k] \sum_{n'=N/2}^{N-1} y[n'] \cos\left(\frac{(4n'+1)k\pi}{2N}\right) \quad (7.101)$$

Now the two summations in the expression of $X[k]$ can be combined to become

$$X[k] = a[k] \sum_{n=0}^{N-1} y[n] \cos\left(\frac{(4n+1)k\pi}{2N}\right) \quad (7.102)$$

We next consider the DFT of $y[n]$:

$$Y[k] = \sum_{n=0}^{N-1} y[n] e^{-j2\pi nk/N} \quad (7.103)$$

If we multiply both sides by $e^{-jk\pi/2N}$ and take the real part of the result, we get:

$$\begin{aligned} \operatorname{Re}[e^{-jk\pi/2N} Y[k]] &= \operatorname{Re} \left[\sum_{n=0}^{N-1} y[n] e^{-j2\pi nk/N} e^{-jk\pi/2N} \right] \\ &= \operatorname{Re} \left[\sum_{n=0}^{N-1} y[n] \left[\cos\left(\frac{(4n+1)k\pi}{2N}\right) - j \sin\left(\frac{(4n+1)k\pi}{2N}\right) \right] \right] \\ &= \sum_{n=0}^{N-1} y[n] \cos\left(\frac{(4n+1)k\pi}{2N}\right) \end{aligned} \quad (7.104)$$

As $y[n]$ is real, the second term of the sine function is imaginary and is therefore dropped. The DCT coefficient $X[k]$ in Eq.7.102 can be further written as:

$$X[k] = a[k] \operatorname{Re}[e^{-jk\pi/2N} Y[k]], \quad (k = 0, \dots, N-1) \quad (7.105)$$

Now we obtained the fast algorithm for the forward DCT which can be carried out in the following three steps:

- **Step 1:** Generate a sequence $y[n]$ from the given sequence $x[n]$:

$$\begin{cases} y[n] = x[2n] \\ y[N-1-n] = x[2n+1] \end{cases} \quad (n = 0, \dots, N/2-1) \quad (7.106)$$

- **step 2:** Carry out DFT of $y[n]$ by FFT (as $y[n]$ is real, $Y[n]$ is symmetric and only half of the data points need be computed):

$$Y[k] = \mathcal{F}[y[n]], \quad (k = 0, \dots, N-1) \quad (7.107)$$

- **step 3:** Obtain DCT $X[k]$ from $Y[k]$ ($k = 0, \dots, N-1$):

$$\begin{aligned} X[k] &= a[k] \operatorname{Re}[e^{-jk\pi/2N} Y[k]] \\ &= a[k] [Y_r[k] \cos(k\pi/2N) + Y_j[k] \sin(k\pi/2N)] \end{aligned} \quad (7.108)$$

where $Y_r[k]$ and $Y_j[k]$ are the real and imaginary part of $Y[k]$, respectively.

Note that the DCT scaling factor $a[k]$ is included in the third step, but no scaling factor (either $1/N$ or $1/\sqrt{N}$) is used during the DFT of $y[n]$.

We next derive the fast algorithm of the inverse DCT. We first consider the real part of the inverse DFT of a sequence $Y[k] = a[k]e^{jk\pi/2N}X[k]$ ($k = 0, \dots, N-1$):

$$\begin{aligned} Re \left[\sum_{k=0}^{N-1} a[k]X[k]e^{jk\pi/2N}e^{j2\pi nk/N} \right] &= Re \left[\sum_{k=0}^{N-1} a[k]X[k]e^{j(4n+1)k\pi/2N} \right] \\ &= \sum_{k=0}^{N-1} a[k]X[k]\cos\left(\frac{(4n+1)k\pi}{2N}\right) = x[2n], \quad (n = 0, \dots, N-1) \end{aligned} \quad (7.109)$$

The first half of these N values are the $N/2$ even components $x[2n]$, ($n = 0, \dots, N/2-1$). To obtain the odd components, recall that $x[n] = x[2N-n-1]$ (first equation in Eq. 7.65), and we have:

$$x[2n+1] = x[2N-(2n+1)-1] = x[2(N-n-1)], \quad (n = 0, \dots, N/2-1) \quad (7.110)$$

i.e., the $N/2$ odd components are actually the second half ($n = N/2, \dots, N-1$) of the previous equation but in reverse order. Now we have the following three steps for the inverse DCT:

- **step 1:** Generate a sequence $Y[k]$ from the given DCT coefficients $X[k]$:

$$Y[k] = a[k]X[k]e^{jk\pi/2N}, \quad (k = 0, \dots, N-1) \quad (7.111)$$

- **step 2:** Carry out inverse DFT of $Y[n]$ by FFT (Only the real part need be computed):

$$y[n] = Re[\mathcal{F}^{-1}[Y[k]]] \quad (7.112)$$

- **Step 3:** Obtain $x[n]'s$ from $y[n]'s$ by

$$\begin{cases} x[2n] = y[n] \\ x[2n+1] = y[N-1-n] \end{cases} \quad (n = 0, \dots, N/2-1) \quad (7.113)$$

Note again that no scaling factor (either $1/N$ or $1/\sqrt{N}$) is used during the inverse DFT of $Y[k]$. Now both the forward or inverse DCT are implemented as a slightly modified DFT which can be carried out by the FFT algorithm with much reduced computational complexity of $O(N \log_2 N)$.

As the DCT and DST are closely related, the fast DCT algorithm considered above can be readily used for the DST. Specifically, we replace k by $N-1-k$

in Eq.7.86 and note $a[N - 1 - k] = b[k]$ (Eq.7.88) to get:

$$\begin{aligned}
 & a[N - 1 - k] \sum_{n=0}^{N-1} x[n] \cos \left(\frac{(2n+1)(N-1-k)\pi}{2N} \right) \\
 &= b[k] \sum_{n=0}^{N-1} x[n] \cos \left(\frac{\pi}{2} + n\pi - \frac{(2n+1)(k+1)\pi}{2N} \right) \\
 &= b[k] \sum_{n=0}^{N-1} x[n] \sin \left(\frac{(2n+1)(k+1)\pi}{2N} - n\pi \right) \\
 &= b[k] \sum_{n=0}^{N-1} x[n](-1)^n \sin \left(\frac{(2n+1)(k+1)\pi}{2N} \right) = Y_s[k]
 \end{aligned} \tag{7.114}$$

This is the DST of a signal $y[n] = x[n](-1)^m$. Based on this result, the DST of a signal vector $\mathbf{x} = [x[0], \dots, x[N-1]]^T$ can be implemented by the following steps:

- **Step 1:** Negate all odd components of \mathbf{x} : $y[n] = x[n](-1)^n$ for all $n = 0, \dots, N-1$;
- **Step 2:** Carry out DCT of $y[n]$ to get $Y_c[k]$;
- **Step 3:** Reverse the order of the DCT coefficients to get the DST coefficients: $X_s[k] = Y_c[N-1-k]$ for all $k = 0, \dots, N-1$.

The inverse DST can be carried out simply by reversing the steps above:

- **Step 1:** Reverse the order of the DST coefficients: $Y_c[k] = X_s[N-1-k]$ for all $k = 0, \dots, N-1$.
- **Step 2:** Carry out inverse DCT of $Y_c[k]$ to get $y[n]$;
- **Step 3:** Negate odd-indexed time samples $x[n] = y[n](-1)^n$ for all $n = 0, \dots, N-1$;

The C code for the fast algorithms of both DCT and DST is given below. The function fdct carries out the DCT (if inv=0) to convert a data vector $x[n]$ ($n = 0, \dots, N-1$) into its DCT coefficients $X[k]$ ($k = 0, \dots, N-1$). This function is also used for the inverse DCT (if inv=1) to reconstruct the signal vector based on its DCT coefficients. This is an in-place algorithm, i.e., the input data will be overwritten by the output. The DST can be implemented by the following function fdst based on the function fdct. The complexity of both functions is $O(N \log_2 N)$ as they are based on the FFT algorithm.

```

fdct(x,N,inv)      // for forward or inverse DCT
    float *x; int N,inv;
{
    int m,n;
    float a,w, *yr,*yi;
    w=3.14159265/2/N;

```

```

a=sqrt(2.0/N);
yr=(float *)malloc(N*sizeof(float)); // allocate memory for
yi=(float *)malloc(N*sizeof(float)); // two temporary vectors
if (inv) {                                // for IDCT
    for (n=0; n<N; n++) x[n]=x[n]*a;
    x[0]=x[0]/sqrt(2.0);
    for (n=0; n<N; n++) {
        yr[n]=x[n]*cos(n*w);
        yi[n]=x[n]*sin(n*w);
    }
}                                         // for DCT
else {
    for (m=0; m<N/22; m++) {
        yr[m]=x[2*m];
        yr[N-1-m]=x[2*m+1];
        yi[m]=yi[N/22+m]=0;
    }
}
fft(yr,yi,N,inv);                      // call FFT function
if (inv) {                                // for inverse DCT
    for (m=0; m<N/2; m++) {
        x[2*m]=yr[m];
        x[2*m+1]=yr[N-1-m];
    }
}
else {                                     // for DCT
    for (n=0; n<N; n++)
        x[n]=cos(n*w)*yr[n]+sin(n*w)*yi[n];
    for (n=0; n<N; n++) x[n]=x[n]*a;
    x[0]=x[0]/sqrt(2.0);
}
free(yr); free(yi);
}

fdst(x,N,inv) // for DST or inverse DST
    float *x; int N,inv;
{
    int n;
    float v;
    if (inv) {                      // inverse DST
        for (n=0; n<N/2; n++)
            { v=x[n]; x[n]=x[N-1-n]; x[N-1-n]=v; }
        fdct(x,N,1);
        for (n=1; n<N; n+=2) x[n]=-x[n];
    }
}

```

```

    }
else {                                // forward DST
    for (n=1; n<N; n+=2) x[n]=-x[n];
    fdct(x,N,0);
    for (n=0; n<N/2; n++)
        { v=x[n]; x[n]=x[N-1-n]; x[N-1-n]=v; }
}
}

```

7.2.5 The DCT and DST Filtering

As a real-valued transform, the computation of the DCT or DST is more straight forward compared to the DFT filtering. In the discussion below we will mainly consider the DCT filtering as the DST filtering is mostly the same.

Example 7.5: The signal shown in the top-left panel of Fig.7.8 is a signal with three frequency components: the DC, as well as two sinusoids at frequencies of 8 Hz and 19 Hz. Moreover, the signal (solid line) is also contaminated by some white noise (dashed line). The DCT spectrum of the signal is shown in the top-right panel in which the three frequency components are clearly seen (solid line), together with the white noise whose energy is spread over all frequencies (dashed line), therefore the name white noise. The lower-right panel of the figure shows the filtered DCT spectrum containing only the frequency component at 8 Hz, and the lower-left panel shows the filtered signal obtained by inverse transform of the filtered spectrum. We can see clearly that only the 8-Hz sinusoid remains while all other components in the original signal are filtered out (solid line), which is compared with the original signal (dashed line). If we assume this 8-Hz sinusoid is the signal of interest and all other components are interference and noise, then this filtering process has effectively extracted the signal by removing the interference and suppressing the noise.

Example 7.6:

Here we compare two different types of signals and their DCTs. Shown in Fig.7.9 are images of two natural scenes, the clouds on the left and the sand on the right, with very different textures. Specifically, In the cloud image, the value of a pixel is very likely to be similar to those of its neighbors, i.e., they are highly correlated, while in the sand image, the values of neighboring pixels are not likely to be related, i.e., they are much less correlated. Such a difference can be quantitatively described by the auto-correlation of the signal defined before

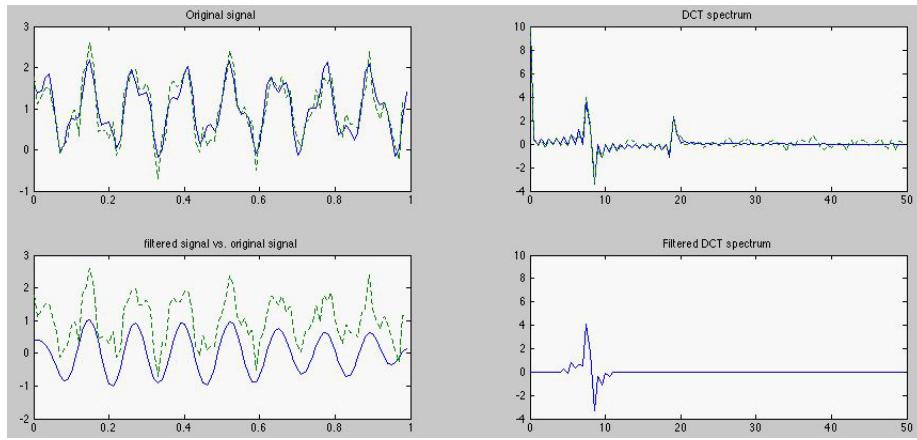


Figure 7.8 Signal before (top) and after (bottom) DCT filtering in both time (left) and frequency (right) domain

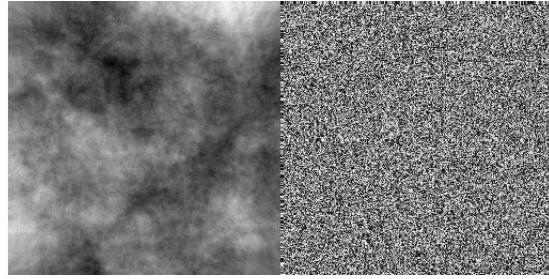


Figure 7.9 Two types of natural scenes: clouds and sand
From left to right and then top to bottom:

in Eq.3.110:

$$r_x(t) = \int_{-\infty}^{\infty} x(\tau)x(\tau-t)d\tau = \int_{-\infty}^{\infty} |X(f)|^2 e^{j2\pi f\tau} df = \mathcal{F}^{-1}[S_x(f)] \quad (7.115)$$

where $X(f) = \mathcal{F}[x(t)]$ is the Fourier spectrum of signal $x(t)$ and $S_x(f) = |X(f)|^2$ is the power density spectrum of signal.

To compare the two types of signals, we take one row of each of the two images as a 1-D signal and consider the auto-correlations of the signal as well as its DCT, as shown in Fig.7.10. The four panels on the left are for the clouds (1st) and sand (3rd) together with their auto-correlation (2nd and 4th). Note that the signal of clouds is highly correlated, and the closer (smaller t in $r_x(t)$) two signal samples the more they are correlated, but the signal for sand is not correlated. (The auto-correlations look symmetric due to the imposed signal periodicity.) The four panels on the right show DCT spectra corresponding to the two signals (1st and 3rd) together with their auto-correlations (2nd and 4th). We see that in frequency domain the frequency components are hardly correlated

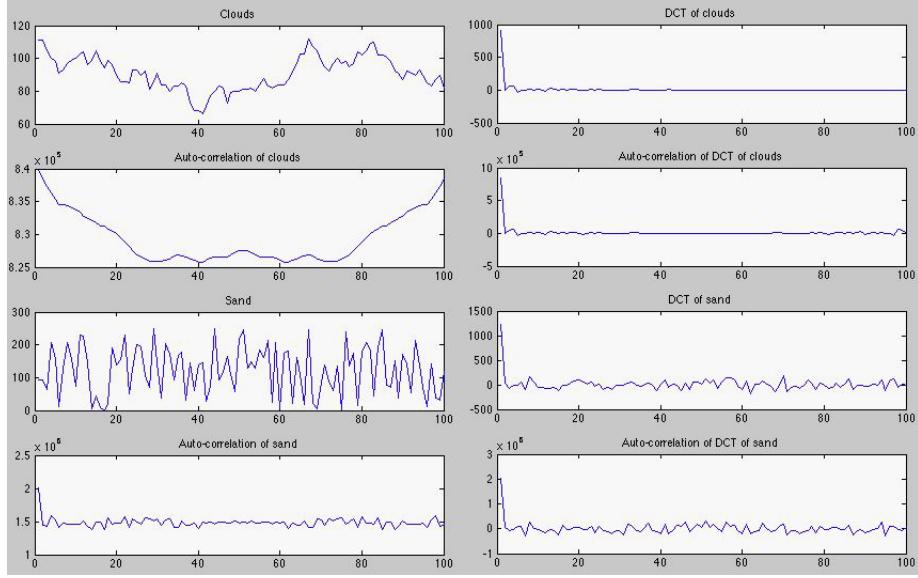


Figure 7.10 Decorrelation of cloud and sand signals

at all. These two very different types of signals of high and low correlations will be reconsidered in the future discussion regarding the statistical properties of the signals (Chapter 10).

In general, all natural signals are correlated to different degrees, depending on their specific natures. Most signals are highly correlated, such as the example of clouds, although some exceptions are less so, such as the sand. But in either case, the components in the spectrum of the signal after an orthogonal transform, such as the Fourier or cosine transform, or any other orthogonal transform for this matter, are much less correlated, i.e., all orthogonal transforms tend to decorrelate the signal, as the autocorrelation of a typical signal is significantly reduced in transform domain.

7.2.6 The Two-Dimensional DCT and DST

Here we consider the DCT and DST filtering of a 2-D signal $x[m, n]$ ($m = 0, \dots, M - 1, n = 0, \dots, N - 1$), such as an image. In the discussion below we will again mainly consider the 2-D DCT filtering as the 2-D DST filtering is mostly the same.

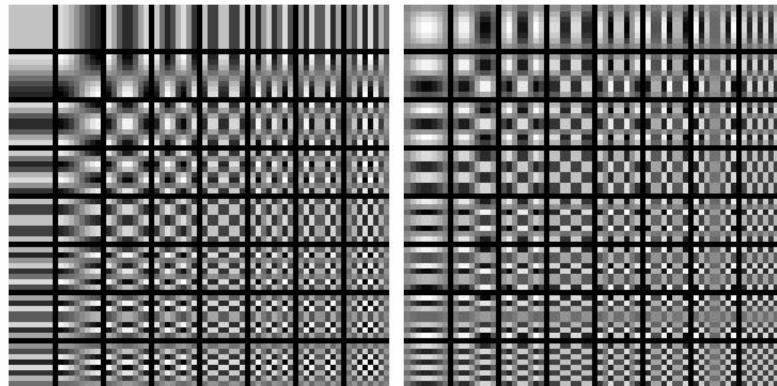


Figure 7.11 The basis $M \times N = 8 \times 8 = 64$ functions of 2-D DCT (left) and DST (right)

The DC component is at the top-left corner, and the highest frequency component in both horizontal and vertical directions is at the lower-right corner.

The forward and inverse 2-D DCT are defined respectively as:

$$\begin{aligned} X[k, l] &= a[k]a[l] \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} x[m, n] \cos\left(\frac{(2m+1)k\pi}{2M}\right) \cos\left(\frac{(2n+1)l\pi}{2N}\right), \\ x[m, n] &= \sum_{l=0}^{N-1} a[l] \sum_{k=0}^{M-1} a[k]X[k, l] \cos\left(\frac{(2m+1)k\pi}{2M}\right) \cos\left(\frac{(2n+1)l\pi}{2N}\right), \\ &\quad (m, k = 0, \dots, M-1, n, l = 0, \dots, N-1) \end{aligned} \quad (7.116)$$

The inverse DCT (second equation) represents the given signal as a linear combination of a set of MN 2-D basis functions each of size $M \times N$ as a product of two sinusoidal functions in horizontal and vertical directions. Fig. 7.11 displays a set of $M \times N = 8 \times 8 = 64$ such 2-D basis functions for both the DCT (left) and DST (right). Each of these MN basis function is weighted by its corresponding coefficient $X[k, l]$, which can be obtained by the forward DCT (first equation) as the projection of the signal onto the corresponding basis function.

Similar to the 2-D DFT, the two summations in either the forward or inverse DCT in Eq. 7.116 can be carried separately in two separate steps. First, we can carry out N M -point 1-D DCTs for each of the columns of the 2-D signal array (the inner summation with respect to m in Eq. 7.116), and then carry out M N -point 1-D DCTs for each of the rows of the resulting array after the first step (the outer summation with respect to n in Eq. 7.116). Of course we can also carry out the row DCTs first and then the column DCTs. In matrix multiplication form, the forward and inverse 2-D DCT can be represented as

$$\begin{cases} \mathbf{X} = \mathbf{C}^T \mathbf{x} \mathbf{C} & \text{(forward)} \\ \mathbf{x} = \mathbf{C} \mathbf{X} \mathbf{C}^T & \text{(inverse)} \end{cases} \quad (7.117)$$

where both the 2-D signal \mathbf{x} and its spectrum \mathbf{X} are $M \times N$ matrices, and the pre-multiplication matrix \mathbf{C} is $M \times M$ for the column transforms, while the post-multiplication matrix \mathbf{C} is $N \times N$ for the row transforms. The DCT spectrum of a 2-D $M \times N$ real signal (e.g., an image) is also a $M \times N$ real matrix composed of MN DCT coefficients $X[k, l]$ ($k = 0, \dots, M - 1, l = 0, \dots, N - 1$) representing the magnitudes of the corresponding basis functions.

The DCT matrix \mathbf{C} can be expressed in terms of its column vectors and the inverse transform can be written as:

$$\begin{aligned} \mathbf{x} &= [\mathbf{c}_0, \dots, \mathbf{c}_{M-1}] \begin{bmatrix} X[0, 0] & \cdots & X[0, N-1] \\ \vdots & \ddots & \vdots \\ X[M-1, 0] & \cdots & X[M-1, N-1] \end{bmatrix} \begin{bmatrix} \mathbf{c}_0^T \\ \vdots \\ \mathbf{c}_{N-1}^T \end{bmatrix} \\ &= \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} X[k, l] \mathbf{c}_k \mathbf{c}_l^T = \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} X[k, l] \mathbf{B}_{kl} \end{aligned} \quad (7.118)$$

Here we have defined $\mathbf{B}_{kl} = \mathbf{c}_k \mathbf{c}_l^T$, where \mathbf{c}_k is the k th column vector of the $M \times M$ DCT matrix for the row transforms and \mathbf{c}_l is the l th column vector of the $N \times N$ DCT matrix for the column transforms. We see that the 2-D signal $\mathbf{x}_{M \times N}$ is now expressed as a linear combination of a set of MN 2-D ($M \times N$) DCT basis functions \mathbf{B}_{kl} ($k, l = 0, \dots, N - 1$), each of which can be obtained from the inverse transform above when all elements of \mathbf{X} are zero except $X[k, l] = 1$. When $M = N = 8$, the $8 \times 8 = 64$ such 2-D DCT basis functions are shown in Fig.7.11. Any 8×8 2-D signal can be expressed as a linear combination of these 64 2-D orthogonal basis functions.

In the equation above, each basis function \mathbf{B}_{kl} is weighted by the kl -th DCT coefficients $X[k, l]$, which can be obtained by the forward transform:

$$\mathbf{X} = \begin{bmatrix} \mathbf{C}_0^T \\ \vdots \\ \mathbf{C}_{M-1}^T \end{bmatrix} \mathbf{x}[\mathbf{c}_0, \dots, \mathbf{c}_{N-1}] \quad (7.119)$$

The kl -th coefficient $X[k, l]$ of this 2-D spectrum is:

$$\begin{aligned} X[k, l] &= \mathbf{c}_k^T \begin{bmatrix} x[0, 0] & \cdots & x[0, N-1] \\ \vdots & \ddots & \vdots \\ x[M-1, 0] & \cdots & x[M-1, N-1] \end{bmatrix} \mathbf{c}_l \\ &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m, n] B_{kl}[m, n] = \langle \mathbf{x}, \mathbf{B}_{kl} \rangle \end{aligned} \quad (7.120)$$

Same as in the 2-D DFT case (Eq.4.262), the coefficient $X[k, l]$ can be found as the projection of the 2-D signal \mathbf{x} onto the kl -th DCT basis function \mathbf{B}_{kl} .

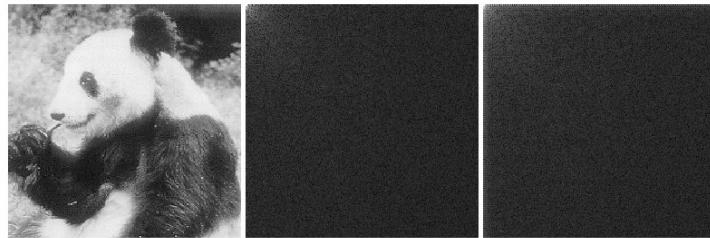


Figure 7.12 An image (left) and its DCT (middle) and DST (right) spectra

Example 7.7: An image and its DCT and DST spectra are shown in Fig. 7.12. Different from the DFT spectrum composed of complex DFT coefficients representing the magnitudes and phases for the frequency components, here the spectrum of either DCT or DST is a real matrix representing the magnitudes of the frequency components all with zero phases.

Various types of filtering, such as high-pass (LP) and low-pass (HP) filtering, can be carried out in the frequency domain by modifying the spectrum of the signal. Fig. 7.13 shows some HP and LP results using two different types of filters, the ideal filter and the Butterworth filter. In the case of an ideal filter, all frequency components farther away from the DC component (top-left corner of the spectrum) than a distance corresponding to the cut-off frequency are suppressed to zero while all other components remain unchanged. The modified spectrum and the resulting low-pass filtered image after the inverse DCT are shown in the figure at top-left and bottom-left, respectively. Similar to the case of the DFT, some obvious ringing artifacts can be observed in the ideal-filtered image. To avoid this, the Butterworth filter without sharp edges can be used, as shown by the pair of images second from the left. The same ideal and Butterworth filters can also be used for HP filtering, as shown by the other two pairs of images on the right. Again, note that the ringing artifacts due to the ideal filter is avoided by Butterworth filtering.

Example 7.8: The DCT can also be used for data compression as illustrated in Fig. 7.14. In this case, 90% of the DCT coefficients (corresponding mostly to some high frequency components) with magnitudes less than a certain threshold value were surprised to zero (black in the image). The image is then reconstructed based on the remaining 10% of the coefficients but containing over 99.6% of the signal energy. As can be seen in the figure, the reconstructed image, with only 0.4% energy lost, looks very much the same as the original one except some very fine details corresponding to high frequency components which were suppressed.

We can throw away 90% of the coefficients but still keep over 99% of the energy only in the frequency domain, but not in the spatial domain, due to the two general properties of all orthogonal transforms: (a) decorrelation of signals and

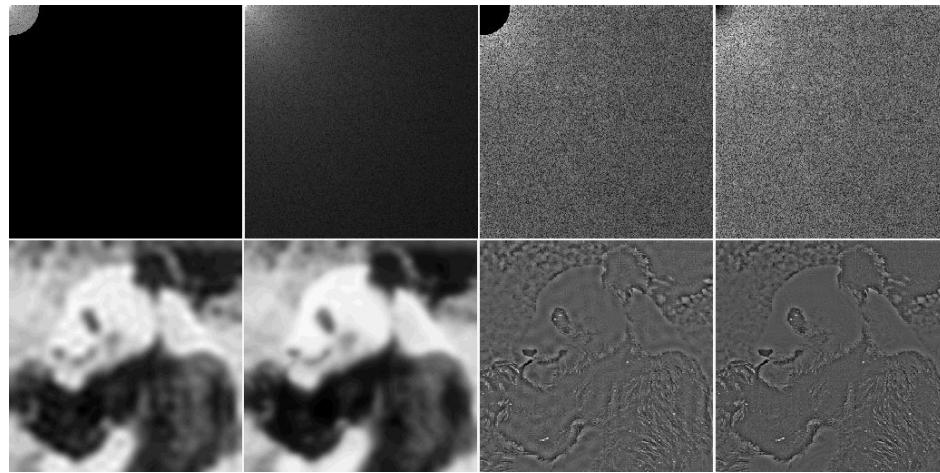


Figure 7.13 LP and HP filtering of an image

Similar to the Fourier transform, DCT also suffers from the ringing artifacts caused by the ideal filters (first and third), which can be avoided by the smooth Butterworth filter.

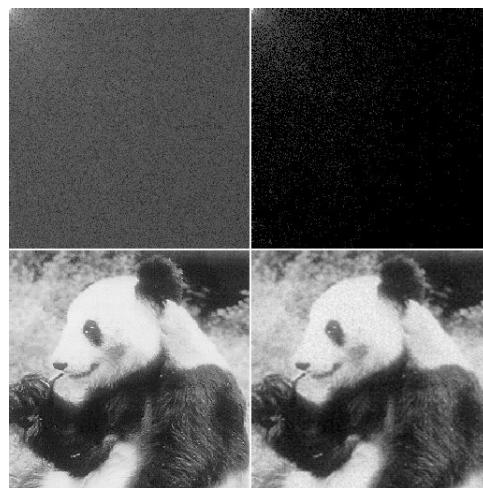


Figure 7.14 Image Compression based on DCT

An image with its DCT spectrum (left) and the reconstructed image based on 10% of the coefficients containing 99.6% of the total energy (right).

(b) compaction of signal energy. In this example, the effect of energy compaction of the DCT is stronger than that of the DFT discussed before. In general, as a real transform method, the DCT is more widely used in image compression than the DFT. For example, it is used in the most popular image compression standard JPEG (<http://en.wikipedia.org/wiki/JPEG>).

7.3 Homework Problems

1. Prove the orthogonality of the DHT given in Eq.7.27. Hint: consider the trigonometric identities

$$\begin{aligned}\sin(\alpha \pm \beta) &= \sin \alpha \cos \beta \pm \sin \beta \cos \alpha \\ \cos(\alpha \pm \beta) &= \cos \alpha \cos \beta \mp \sin \beta \sin \alpha\end{aligned}\quad (7.121)$$

and then use the result of Eq.1.40.

2. Prove the following relation (Eq.7.76):

$$\sum_{n=0}^{N-1} \cos\left(\frac{(2n+1)k\pi}{2N}\right) = \begin{cases} N & k = 0 \\ -N & k = 2N \\ 0 & \text{else} \end{cases}, \quad (0 \leq n \leq 2N) \quad (7.122)$$

Hints: You may find it is helpful to use the identity $\sum_{n=0}^{N-1} x^n = (1 - x^N)/(1 - x)$, and to consider the two different cases when k is either even or odd.

3. Let $x[n] = n + 1$ ($n = 0, 1, 2, 3$) be a discrete signal with period $N = 4$. Find its DHT, DCT and DST by matrix multiplication $\mathbf{X} = \mathbf{Hx}$, $\mathbf{X} = \mathbf{C}^T \mathbf{x}$ and $\mathbf{X} = \mathbf{S}^T \mathbf{x}$, respectively, where \mathbf{H} , \mathbf{C} and \mathbf{S} are given in Eqs.7.35 and 7.94. Then carry out the inverse transform also by matrix multiplication $\mathbf{x} = \mathbf{TX}$, $\mathbf{x} = \mathbf{CX}$ and $\mathbf{x} = \mathbf{SX}$ to confirm that the signal is perfectly reconstructed.
4. Develop a Matlab function for the discrete Hartley transform (DHT). Apply it to an $N=8$ sequence $\mathbf{x} = [x[0], \dots, x[7]]^T$ of your choice to obtain its N transform coefficients, then carry out the inverse transform to reconstruct the sequence from these transform coefficients.
5. Understand the C code for the fast DCT algorithm provided in the text and convert it into a Matlab function. Then carry out the forward and inverse DCT using the $N=8$ sequence chosen for the previous problem. Confirm the perfect reconstruction is achieved.
6. Develop a Matlab function for the DST and repeat the above.
7. Implement the 2-D DHT of the same image used in the homework of Chapter 5 and carry out various types of filtering (low-pass, high-pass, etc.) of the image in transform domain. Then carry out the inverse transform and display the filtered image. Compare the filtering effects with those obtained by the Fourier transform obtained in Chapter 5.
8. Carry out compression of the image used before as shown in Fig.5.21 by suppressing to zero all sequency components lower than a certain threshold. Obtain the percentage of such suppressed frequency components, and the percentage of lost energy (in terms of signal value squared). (Note that this exercise only serves to illustrate the basic idea of image compression but it is not how image compression is practically done, where those components suppressed need to be recorded as well.)
9. Repeat the two problems above for the discrete cosine transform DCT.

10. Repeat the two problems above for the discrete sine transform DST.

8 The Walsh-Hadamard, Slant and Haar Transforms

In this chapter, we will consider a set of three real orthogonal transforms including the Walsh-Hadamard transform (WHT), slant transform (ST) and discrete Haar transform (DHT), all of which are defined quite differently from the previously considered transforms all closely related to the Fourier transform based on sinusoidal kernel functions. In fact, the transforms considered here are no longer continuous and smooth in nature, and they can be used to capture some different types of features and components of the signal being transformed.

8.1 The Walsh-Hadamard Transform

8.1.1 Hadamard Matrix

The Walsh-Hadamard transform matrix can be most conveniently defined based on the concept of *Kronecker product*. Given two matrices $\mathbf{A} = [a_{ij}]_{m \times n}$ and $\mathbf{B} = [b_{ij}]_{k \times l}$, we can define their Kronecker product as:

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & \cdots & a_{1n}\mathbf{B} \\ \cdots & \cdots & \cdots \\ a_{m1}\mathbf{B} & \cdots & a_{mn}\mathbf{B} \end{bmatrix}_{mk \times nl} \quad (8.1)$$

Note that in general, $\mathbf{A} \otimes \mathbf{B} \neq \mathbf{B} \otimes \mathbf{A}$. Now we can define the *Hadamard Matrix* recursively as:

$$\mathbf{H}_1 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (8.2)$$

$$\mathbf{H}_n = \mathbf{H}_1 \otimes \mathbf{H}_{n-1} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{H}_{n-1} & \mathbf{H}_{n-1} \\ \mathbf{H}_{n-1} & -\mathbf{H}_{n-1} \end{bmatrix} \quad (8.3)$$

Here \mathbf{H}_n is $N \times N$ matrix with $N = 2^n$. When $n = 2$, $N = 2^2 = 4$ and we have:

$$\mathbf{H}_2 = \mathbf{H}_1 \otimes \mathbf{H}_1 = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{H}_1 & \mathbf{H}_1 \\ \mathbf{H}_1 & -\mathbf{H}_1 \end{bmatrix} = \frac{1}{\sqrt{4}} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \quad (8.4)$$

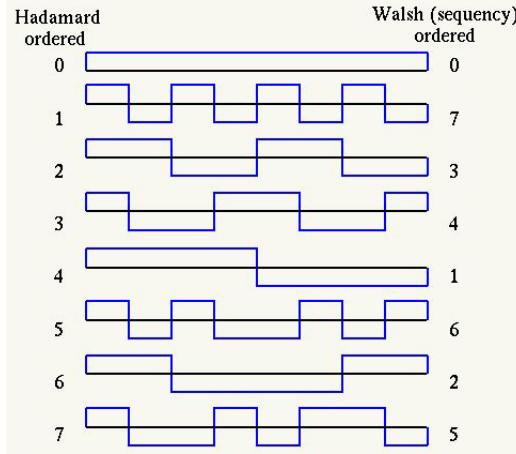


Figure 8.1 The basis functions of the WHT

When $n = 3$, $N = 2^3 = 8$ and we have:

$$\mathbf{H}_3 = \mathbf{H}_1 \otimes \mathbf{H}_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{H}_2 & \mathbf{H}_2 \\ \mathbf{H}_2 & -\mathbf{H}_2 \end{bmatrix} = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \begin{matrix} 0 & 0 \\ 1 & 7 \\ 2 & 3 \\ 3 & 4 \\ 4 & 1 \\ 5 & 6 \\ 6 & 2 \\ 7 & 5 \end{matrix} \quad (8.5)$$

The first column to the right of the matrix is for the index number k of the $N = 8$ rows, and the second column is for the *sequency* s of each row, defined as the number of zero-crossings or sign changes in the row. Similar to frequency, sequency also measures the rate of changes or variations in a signal. However, sequency can also be used to measure non-periodical signals as well as periodic ones. We see that in \mathbf{H}_3 , sequencies $s = 0, 1, 2, 3, 4, 5, 6, 7$ correspond to $k = 0, 4, 6, 2, 3, 7, 5, 1$, respectively. The conversion between sequency s and index number k will be considered later.

Alternatively, a Hadamard matrix \mathbf{H} can also be defined in terms of its element $h[k, m]$ in the k th row and m th column as below (for simplicity, the scaling factor $1/\sqrt{N}$ is neglected for now):

$$h[k, m] = (-1)^{\sum_{i=0}^{n-1} k_i m_i} = \prod_{i=0}^{n-1} (-1)^{k_i m_i} = h[m, k] \quad (k, m = 0, 1, \dots, N-1) \quad (8.6)$$

where

$$k = \sum_{i=0}^{n-1} k_i 2^i = (k_{n-1} k_{n-2} \cdots k_1 k_0)_2, \quad (k_i = 0, 1) \quad (8.7)$$

$$m = \sum_{i=0}^{n-1} m_i 2^i = (m_{n-1} m_{n-2} \cdots m_1 m_0)_2 \quad (m_i = 0, 1) \quad (8.8)$$

i.e., $(k_{n-1} k_{n-2} \cdots k_1 k_0)_2$ and $(m_{n-1} m_{n-2} \cdots m_1 m_0)_2$ are the binary representations of k and m , respectively. Obviously, we need $n = \log_2 N$ bits in these binary representations. For example, when $n = 3$ and $N = 2^n = 8$, the element $h[k, m]$ in row $k = 2 = (010)_2$ and column $m = 3 = (011)_2$ of \mathbf{H}_3 is $(-1)^{0+1+0} = -1$.

It is easy to show that this alternative definition of the Hadamard matrix is actually equivalent to the previous recursive definition given in Eqs. 8.2 and 8.3. First, when $n = 1$ and $N = 2^n = 2$, the two rows and columns indexed by a single bit of k_0 and m_0 , respectively, and the product $k_0 m_0$ of the two bits has four possible values, $0 \times 0 = 0$, $0 \times 1 = 0$, $1 \times 0 = 0$ and $1 \times 1 = 1$, and they correspond to the four elements of the matrix, i.e., $h[0, 0] = h[0, 1] = h[1, 0] = (-1)^{k_0 m_0} = (-1)^0 = 1$ and $h[1, 1] = (-1)^{k_0 m_0} = (-1)^1 = -1$. This is actually Eq. 8.2.

Next, when n is increased by 1, the size $N = 2^n$ of the matrix is doubled, and one more bit k_{n-1} and m_{n-1} (the most significant bit) is needed for the binary representations of k and m , respectively. The product of these two most significant bits $k_{n-1} m_{n-1}$ determines the four quadrants of the new matrix \mathbf{H}_n . The first three quadrants (upper-left, upper-right and lower-left) corresponding to $k_{n-1} m_{n-1} = 0$ are therefore identical to \mathbf{H}_{n-1} , while the lower-right quadrant corresponding to $k_{n-1} m_{n-1} = 1$ is the negation of \mathbf{H}_{n-1} . This is the recursion in Eq. 8.3.

The Hadamard matrix \mathbf{H} is real and symmetric, and also orthogonal:

$$\mathbf{H} = \mathbf{H}^* = \mathbf{H}^T = \mathbf{H}^{-1} \quad (8.9)$$

The orthogonality of \mathbf{H} can be proven by induction. This is left for the reader as a homework exercise.

8.1.2 Hadamard Ordered Walsh-Hadamard Transform (WHT _{h})

The Hadamard matrix can be written in terms of its columns:

$$\mathbf{H} = [\mathbf{h}_0, \dots, \mathbf{h}_{N-1}] \quad (8.10)$$

As \mathbf{H} is an orthogonal matrix, its N vectors are orthonormal:

$$\langle \mathbf{h}_k, \mathbf{h}_l \rangle = \mathbf{h}_k^T \mathbf{h}_l = \delta[k - l] \quad (8.11)$$

they form a complete basis that spans the N -dimensional vector space, and the Hadamard matrix \mathbf{H} can be used to define an orthogonal transform, called

Hadamard ordered Walsh-Hadamard transform (WHT_{*h*}): ¹

$$\begin{cases} \mathbf{X} = \mathbf{H}\mathbf{x} & \text{(forward)} \\ \mathbf{x} = \mathbf{H}\mathbf{X} & \text{(inverse)} \end{cases} \quad (8.12)$$

Here $\mathbf{x} = [x[0], \dots, x[N-1]]^T$ is an *N*-point signal vector and $\mathbf{X} = X[0], \dots, X[N-1]]^T$ is its WHT spectrum vectors. As $\mathbf{H}^{-1} = \mathbf{H}$, the forward (first equation) and inverse (second equation) transforms are identical. Also note that the WHT can be carried out by additions and subtractions alone.

The inverse transform (IWHT_{*h*}) can be further written as:

$$\mathbf{x} = [\mathbf{h}_0, \dots, \mathbf{h}_{N-1}] \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \sum_{k=0}^{N-1} X[k] \mathbf{h}_k \quad (8.13)$$

This expression represents the signal vector \mathbf{x} is expressed as a linear combination of the *N* basis vectors \mathbf{h}_k weighted by the WHT coefficients $X[k]$ ($k = 0, \dots, N-1$), which can be found as the projection of \mathbf{x} onto the *k*th basis vector:

$$X[k] = \langle \mathbf{x}, \mathbf{h}_k \rangle = \mathbf{h}_k^T \mathbf{x}, \quad (k = 0, \dots, N-1) \quad (8.14)$$

this just the component form of the forward WHT:

$$\mathbf{X} = \mathbf{H}\mathbf{x} = \mathbf{H}^T \mathbf{x} = \begin{bmatrix} \mathbf{h}_0^T \\ \vdots \\ \mathbf{h}_{N-1}^T \end{bmatrix} \mathbf{x} \quad (8.15)$$

Note that $X[k]$ can also be written as

$$X[k] = \sum_{m=0}^{N-1} h[k, m] x[m] = \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} x[m] \prod_{i=0}^{n-1} (-1)^{m_i k_i} \quad (8.16)$$

8.1.3 Fast Walsh-Hadamard Transform Algorithm

The complexity of WHT implemented as a matrix multiplication $\mathbf{X} = \mathbf{H}\mathbf{x}$ is of course $O(N^2)$. However, similar to the FFT algorithm, we can also derive a fast WHT algorithm with complexity of $O(N \log_2 N)$ as shown below. We assume

¹ In Matlab the forward and inverse WHT can be carried out by functions fwht and ifwht, respectively. However, for Parseval's identity to hold, these forward and inverse transforms need to be rescaled: $X = fwht(x) * sqrt(length(x))$ and $x = ifwht(X) / sqrt(length(x))$.

$n = 3$ and $N = 2^n = 8$, and write the WHT _{h} of an 8-point signal \mathbf{x} as:

$$\mathbf{X} = \begin{bmatrix} X[0] \\ \vdots \\ X[3] \\ X[4] \\ \vdots \\ X[7] \end{bmatrix} = \mathbf{H}_3 \mathbf{x} = \begin{bmatrix} \mathbf{H}_2 & \mathbf{H}_2 \\ \mathbf{H}_2 & -\mathbf{H}_2 \end{bmatrix} \begin{bmatrix} x[0] \\ \vdots \\ x[3] \\ x[4] \\ \vdots \\ x[7] \end{bmatrix} \quad (8.17)$$

This equation can be separated into two parts. The first half of vector \mathbf{X} can be obtained as

$$\begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ X[3] \end{bmatrix} = \mathbf{H}_2 \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix} + \mathbf{H}_2 \begin{bmatrix} x[4] \\ x[5] \\ x[6] \\ x[7] \end{bmatrix} = \mathbf{H}_2 \begin{bmatrix} x_1[0] \\ x_1[1] \\ x_1[2] \\ x_1[3] \end{bmatrix} \quad (8.18)$$

where we have defined

$$x_1[i] = x[i] + x[i+4] \quad (i = 0, \dots, 3) \quad (8.19)$$

Similarly the second half of vector \mathbf{X} can be obtained as

$$\begin{bmatrix} X[4] \\ X[5] \\ X[6] \\ X[7] \end{bmatrix} = \mathbf{H}_2 \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix} - \mathbf{H}_2 \begin{bmatrix} x[4] \\ x[5] \\ x[6] \\ x[7] \end{bmatrix} = \mathbf{H}_2 \begin{bmatrix} x_1[4] \\ x_1[5] \\ x_1[6] \\ x_1[7] \end{bmatrix} \quad (8.20)$$

where we have defined

$$x_1[i+4] = x[i] - x[i+4] \quad (i = 0, \dots, 3) \quad (8.21)$$

What we did above is to convert an 8-point WHT into two 4-point WHTs. This process can be carried out recursively. We next rewrite Eq. 8.18 as:

$$\begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ X[3] \end{bmatrix} = \begin{bmatrix} \mathbf{H}_1 & \mathbf{H}_1 \\ \mathbf{H}_1 & -\mathbf{H}_1 \end{bmatrix} \begin{bmatrix} x_1[0] \\ x_1[1] \\ x_1[2] \\ x_1[3] \end{bmatrix} \quad (8.22)$$

which can again be separated into two halves. The first half is

$$\begin{aligned} \begin{bmatrix} X[0] \\ X[1] \end{bmatrix} &= \mathbf{H}_1 \begin{bmatrix} x_1[0] \\ x_1[1] \end{bmatrix} + \mathbf{H}_1 \begin{bmatrix} x_1[2] \\ x_1[3] \end{bmatrix} = \mathbf{H}_1 \begin{bmatrix} x_2[0] \\ x_2[1] \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_2[0] \\ x_2[1] \end{bmatrix} = \begin{bmatrix} x_2[0] + x_2[1] \\ x_2[0] - x_2[1] \end{bmatrix} \end{aligned} \quad (8.23)$$

where

$$x_2[i] = x_1[i] + x_1[i+2] \quad (i = 0, 1) \quad (8.24)$$

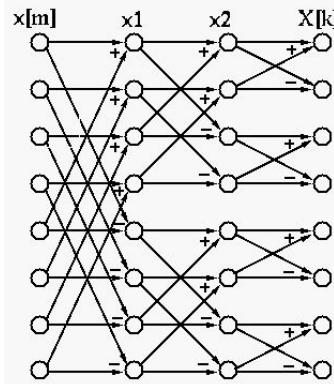


Figure 8.2 The fast WHT algorithm

and

$$X[0] = x_2[0] + x_2[1], \quad X[1] = x_2[0] - x_2[1] \quad (8.25)$$

The second half is

$$\begin{aligned} \begin{bmatrix} X[2] \\ X[3] \end{bmatrix} &= \mathbf{H}_1 \begin{bmatrix} x_1[0] \\ x_1[1] \end{bmatrix} - \mathbf{H}_1 \begin{bmatrix} x_1[2] \\ x_1[3] \end{bmatrix} = \mathbf{H}_1 \begin{bmatrix} x_2[2] \\ x_2[3] \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_2[2] \\ x_2[3] \end{bmatrix} = \begin{bmatrix} x_2[2] + x_2[3] \\ x_2[2] - x_2[3] \end{bmatrix} \end{aligned} \quad (8.26)$$

where

$$x_2[i+2] = x_1[i] - x_1[i+2] \quad (i = 0, 1) \quad (8.27)$$

and

$$X[2] = x_2[2] + x_2[3], \quad X[3] = x_2[2] - x_2[3] \quad (8.28)$$

Similarly the coefficients $X[4]$ through $X[7]$ in the second half of the transform in Eq. 8.20 can be obtained by the same process. Summarizing the above steps of Equations 8.19, 8.21, 8.24, 8.25, 8.27, 8.28, we get the fast WHT algorithm as illustrated in Fig. 8.2.

8.1.4 Sequencey Ordered Walsh-Hadamard Matrix (WHT_w)

The rows (or columns) in the WHT matrix \mathbf{H} are not arranged in order of their sequencies, while it makes better sense if the elements of the WHT spectrum $\mathbf{X} = [X[0], X[1], \dots, X[N-1]]^T$ are arranged according to the sequencies of the corresponding basis vectors, so that they represent different sequencey components contained in the signal in a low-to-high order, similar to the coefficients in the Fourier transform. To do so, we need to re-order the rows (or columns) of the Hadamard matrix H according to their sequencies. We first consider the con-

version of a given sequency number s into the corresponding row index number k in Hadamard order, in the following three steps:

1. Represent s in binary form:

$$s = (s_{n-1} \cdots s_0)_2 = \sum_{i=0}^{n-1} s_i 2^i \quad (8.29)$$

2. Convert this n -bit binary number to an n -bit Gray code :

$$g = (g_{n-1} \cdots g_0)_2, \quad \text{where } g_i = s_i \oplus s_{i+1} \quad (i = 0, \dots, n-1) \quad (8.30)$$

Here \oplus represents exclusive OR of two bits, and $s_n = 0$ is defined as zero.

3. Bit-reverse the Gray code bits g_i 's to get k'_i :

$$k_i = g_{n-1-i} = s_{n-1-i} \oplus s_{n-i} \quad (8.31)$$

Now the row index k can be obtained:

$$k = (k_{n-1} k_{n-2} \cdots k_1 k_0)_2 = \sum_{i=0}^{n-1} s_{n-1-i} \oplus s_{n-i} 2^i = \sum_{j=0}^{n-1} s_j \oplus s_{j+1} 2^{n-1-j} \quad (8.32)$$

where $j = n - 1 - i$ or equivalently $i = n - 1 - j$.

For example, when $n = 3$ and $N = 2^3 = 8$, we have

s	0	1	2	3	4	5	6	7
binary	000	001	010	011	100	101	110	111
Gray code	000	001	011	010	110	111	101	100
bit-reverse	000	100	110	010	011	111	101	001
k	0	4	6	2	3	7	5	1

Now the sequency-ordered, also called Walsh-ordered, Walsh-Hadamard matrix can be obtained as

$$\mathbf{H}_w = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix} \begin{matrix} 0 & 0 \\ 1 & 4 \\ 2 & 6 \\ 3 & 2 \\ 4 & 3 \\ 5 & 7 \\ 6 & 5 \\ 7 & 1 \end{matrix} \quad (8.34)$$

Here a subscript w is used to indicate the row vectors of this matrix \mathbf{H} is sequency-ordered (or Walsh-ordered). The two columns to the right of the matrix are the indices of the row vectors in the sequency order (first column) and the original Hadamard order (second column). Note that this sequency-ordered matrix is still symmetric: $\mathbf{H}_w^T = \mathbf{H}_w$.

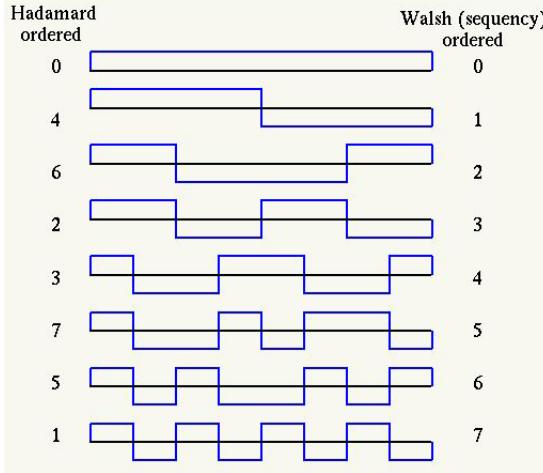


Figure 8.3 The basis functions of the WHT (sequency ordered)

Now the sequency-ordered Walsh-Hadamard transform (WHT_w) can be carried out as

$$\mathbf{X} = \mathbf{H}_w \mathbf{x} \quad (8.35)$$

or in component form:

$$X[k] = \sum_{m=0}^{N-1} h_w[k, m] x[m] \quad (8.36)$$

where $h_w[k, m]$ is the element in the k th row and n th column of \mathbf{H}_w .

8.1.5 Fast Walsh-Hadamard Transform (Sequency Ordered)

The sequency ordered Walsh-Hadamard transform WHT_w can be obtained by first carrying out the fast WHT _{h} and then reordering the components of \mathbf{X} as shown above. Alternatively, we can use the following fast WHT _{w} directly with better efficiency.

Similar to the WHT shown in Eq.8.16, the sequency ordered WHT of $x[m]$ can be represented as:

$$\begin{aligned} X[k] &= \sum_{m=0}^{N-1} h_w[k, m] x[m] = \sum_{m=0}^{N-1} x[m] \prod_{j=0}^{n-1} (-1)^{(k_{n-1-j} + k_{n-j})m_j} \\ &= \sum_{m=0}^{N-1} x[m] \prod_{i=0}^{n-1} (-1)^{(k_i + k_{i+1})m_{n-1-i}} \end{aligned} \quad (8.37)$$

Here $N = 2^n$ and $k_n = 0$. The second equal sign is due to the conversion of index k from Hadamard order to sequency order (Eq.8.32). Here we have also defined $i = n - 1 - j$ and note that $(-1)^{k_i \oplus k_{i+1}} = (-1)^{k_i + k_{i+1}}$, where $m_i, k_i = 0, 1$.

In the following, we assume $n = 3$, $N = 2^3 = 8$, and represent m and k in binary form as $m = (m_2 m_1 m_0)_2$ and $k = (k_2 k_1 k_0)_2$ respectively:

$$m = \sum_{i=0}^{n-1} m_i 2^i = 4m_2 + 2m_1 + m_0, \quad k = \sum_{i=0}^{n-1} k_i 2^i = 4k_2 + 2k_1 + k_0 \quad (8.38)$$

Here $k_n = k_3 = 0$ by definition. This 8-point WHT_w can be carried out in these steps:

- As the first step of the algorithm, we rearrange the order of the samples $x[m]$ by bit-reversal to define:

$$x_0[4m_0 + 2m_1 + m_2] = x[4m_2 + 2m_1 + m_0] \quad \text{for } m = 0, 1, \dots, 7 \quad (8.39)$$

Now Eq.8.37 can be written as:

$$\begin{aligned} X[k] &= \sum_{m_2=0}^1 \sum_{m_1=0}^1 \sum_{m_0=0}^1 x_0[4m_0 + 2m_1 + m_2] \prod_{i=0}^2 (-1)^{(k_i+k_{i+1})m_{n-1-i}} \\ &= \sum_{l_0=0}^1 \sum_{l_1=0}^1 \sum_{l_2=0}^1 x_0[4l_2 + 2l_1 + l_0] \prod_{i=0}^2 (-1)^{(k_i+k_{i+1})l_i} \end{aligned} \quad (8.40)$$

Here we have defined $l_i = m_{n-1-i}$.

- Expanding the 3rd summation into two terms for $l_2 = 0$ and $l_2 = 1$, we get

$$\begin{aligned} X[k] &= \sum_{l_0=0}^1 \sum_{l_1=0}^1 \prod_{i=0}^1 (-1)^{(k_i+k_{i+1})l_i} [x_0[2l_1 + l_0] + (-1)^{k_2+k_3} x_0[4 + 2l_1 + l_0]] \\ &= \sum_{l_0=0}^1 \sum_{l_1=0}^1 \prod_{i=0}^1 (-1)^{(k_i+k_{i+1})l_i} x_1[4k_2 + 2l_1 + l_0] \end{aligned} \quad (8.41)$$

where x_1 is defined as

$$x_1[4k_2 + 2l_1 + l_0] = x_0[2l_1 + l_0] + (-1)^{k_2+k_3} x_0[4 + 2l_1 + l_0] \quad (8.42)$$

- Again, expanding the 2nd summation into two terms for $l_1 = 0$ and $l_1 = 1$, we get

$$\begin{aligned} X[k] &= \sum_{l_0=0}^1 (-1)^{(k_i+k_{i+1})l_0} [x_1[4k_2 + l_0] + (-1)^{k_1+k_2} x_1[4k_2 + 2 + l_0]] \\ &= \sum_{l_0=0}^1 (-1)^{(k_i+k_{i+1})l_0} x_2[4k_2 + 2k_1 + m_0] \end{aligned} \quad (8.43)$$

where x_2 is defined as

$$x_2[4k_2 + 2k_1 + l_0] = x_1[4k_2 + l_0] + (-1)^{k_1+k_2} x_1[4k_2 + 2 + l_0] \quad (8.44)$$

- Finally, expanding the 1st summation into two terms for $l_0 = 0$ and $l_0 = 1$, we have

$$X[k] = x_2[4k_2 + 2k_1] + (-1)^{k_0+k_1} x_2[4k_2 + 2k_1 + 1] \quad (8.45)$$

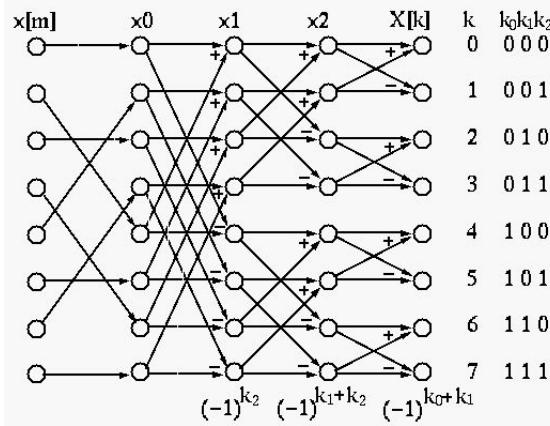


Figure 8.4 The fast WHT algorithm (sequency ordered)

Summarizing the above steps, we get the fast WHT_w algorithm composed of the bit-reversal and the three equations (11), (12), and (13), as illustrated in Fig.8.4. In general, the algorithm has $\log_2 N$ stages each with complexity $O(N)$, the total complexity is $O(N \log_2 N)$.

The C code for the fast WHT algorithm is given below. The WHT function takes a data vector $x[m]$ ($m = 0, \dots, N - 1$) and converts it to WHT coefficients $X[k]$ ($k = 0, \dots, N - 1$), which are Hadamard ordered if the argument sequency=0, or sequency ordered if sequency=1. This is an in-place algorithm, i.e., the input data will be overwritten by the output. The function can be used for both forward and inverse WHT transforms as they are identical.

```
wht(x,N,sequency)
    float *x;
    int N,sequency;
{ int i,j,k,j1,m,n;
    float w,*y,t;

    m=log2f((float)N);
    y=(float *)malloc(N*sizeof(float));
    for (i=0; i<m; i++) {           // for log2 N stages
        n=pow(2,m-1-i);           // length of section
        k=0;
        while (k<N-1) {           // for all sections in a stage
            for (j=0; j<n; j++) { // for all points in a section
                j1=k+j;
                t=x[j1]+x[j1+n];
                x[j1+n]=x[j1]-x[j1+n];
                x[j1]=t;
            }
        }
    }
    free(y);
}
```

```

    }
    k+=2*n;           // move on to next section
}
}

w=1.0/sqrt((float)N);
for (i=0; i<N; i++) x[i]=x[i]*w;
if (sequency)      // converting to sequency (Walsh) order
{
    for (i=0; i<N; i++) { j=h2w(i,m); y[i]=x[j]; }
    for (i=0; i<N; i++) x[i]=y[i];
}
free(y);
}

```

where h2w is a function that converts a sequency index i to Hadamard index j :

```

int h2w(i,m)      // converts a sequency index i to Hadamard index j
{
    int i,m;
    int j,k;
    i=i^(i>>1);
    j=0;
    for (k=0; k<m; ++k)
        j=(j << 1) | (1 & (i >> k));      // bit-reversal
    return j;
}

```

Example 8.1: The sequency ordered WHT of an 8-point signal vector $\mathbf{x} = [0, 0, 2, 3, 4, 0, 0, 0]^T$ can be obtained by matrix multiplication:

$$\mathbf{X} = \mathbf{H}_w \mathbf{x} = [3.18, 0.35, -3.18, -0.35, 1.77, -1.06, -1.77, 1.06]^T \quad (8.46)$$

where \mathbf{H}_w is given in Eq.8.34. The inverse transform (which is identical to the forward transform as $\mathbf{H}_w^{-1} = \mathbf{H}_w$) represents the signal vector as a linear combination of a set of square waves of different sequencies:

$$\mathbf{x} = \mathbf{H}_w \mathbf{X} = [h_0, \dots, h_7] \mathbf{X} = \sum_{k=0}^7 X[k] \mathbf{h}_k [0, 0, 2, 3, 4, 0, 0, 0]^T \quad (8.47)$$

This example is illustrated in Fig.8.5.

Example 8.2:

The WHT and DCT transforms of a set of signals are shown in Fig.8.6. The original signals are shown in the first and third columns, in comparison with

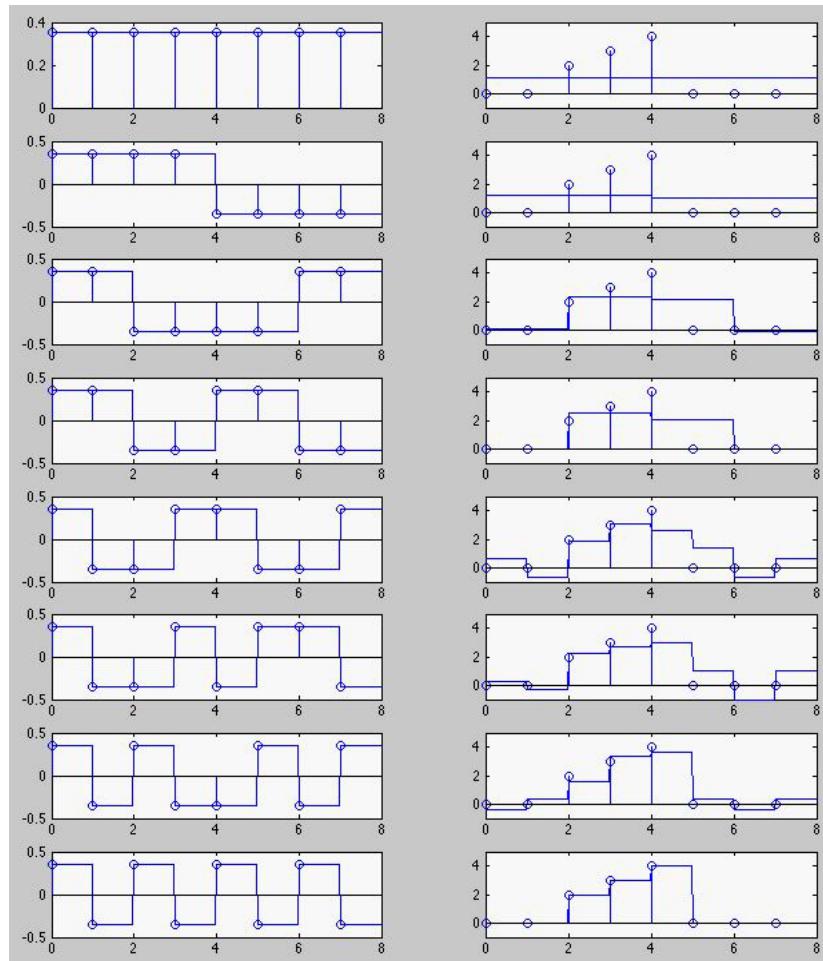


Figure 8.5 The WHT of a 8-point signal

The left column shows the 8 basis WHT functions (both continuous and discrete), while the right column shows how a signal can be reconstructed by the inverse WHT (Eq.8.47) as a linear combination of these basis functions weighted by WHT coefficients obtained by the forward WHT (Eq.8.46). The signal is reconstructed using progressively more components of higher sequencies (from DC component alone to all 8 sequency components).

the reconstructions by the inverse transforms based on 20% of the transform coefficients with the greatest magnitudes (the DCT and WHT coefficients are shown in the 1st and 3rd columns, respectively), while the remaining 80% coefficients are completely removed (suppressed to zero). The reconstruction errors (in percentage) depend on the transform method, as well as the specific signal types, as listed in Table 8.2. We see that the two transform methods are each good for the representation of certain types of signals. For example, the DCT

is effective for sinusoidal signals such as in cases 1, 2 and 3, while the WHT is effective for sawtooth and square waves in cases 5 and 7. Note that as the square wave happens to be proportional to one of the basis vectors of the WHT, it can be perfectly represented by a signal WHT coefficient for that basis vector.

	Signal type	Percentage Error	
		DCT	WHT
1	Sinusoid	0.00	0.00
2	Two-tune sinusoids	2.23	8.19
3	Decaying sinusoid	0.08	0.47
4	Chirp	24.39	21.55
5	Sawtooth	2.12	0.00
6	Triangle	0.00	0.00
7	Square wave	1.05	0.00
8	Impulses	42.31	47.42
9	Random noise	31.68	31.69

8.2 The Slant Transform

8.2.1 Slant Matrix

Like the Hadamard matrix, the matrix for the slant transform (ST) can also be generated recursively. Initially when $n = 1$, the slant transform matrix of size $N = 2^n = 2$ is identically defined as \mathbf{H}_1 for the Hadamard matrix (Eq.8.2):

$$\mathbf{S}_1 = \mathbf{S}_1^T = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (8.48)$$

The recursive definition for matrix \mathbf{S}_n of size $N = 2^n$ for $n > 1$ is given below. Here we will use \mathbf{S}^T in the discussion below for a reason to be given later.

$$\mathbf{S}_n^T = \mathbf{R}_n [\mathbf{S}_1^T \otimes \mathbf{S}_{n-1}^T] = \frac{1}{\sqrt{2}} \mathbf{R}_n \begin{bmatrix} \mathbf{S}_{n-1}^T & \mathbf{S}_{n-1}^T \\ \mathbf{S}_{n-1}^T - \mathbf{S}_{n-1}^T & \end{bmatrix} = \frac{1}{\sqrt{2}} \mathbf{R}_n \begin{bmatrix} \mathbf{S}_{n-1} & \mathbf{S}_{n-1} \\ \mathbf{S}_{n-1} - \mathbf{S}_{n-1} & \end{bmatrix}^T \quad (8.49)$$

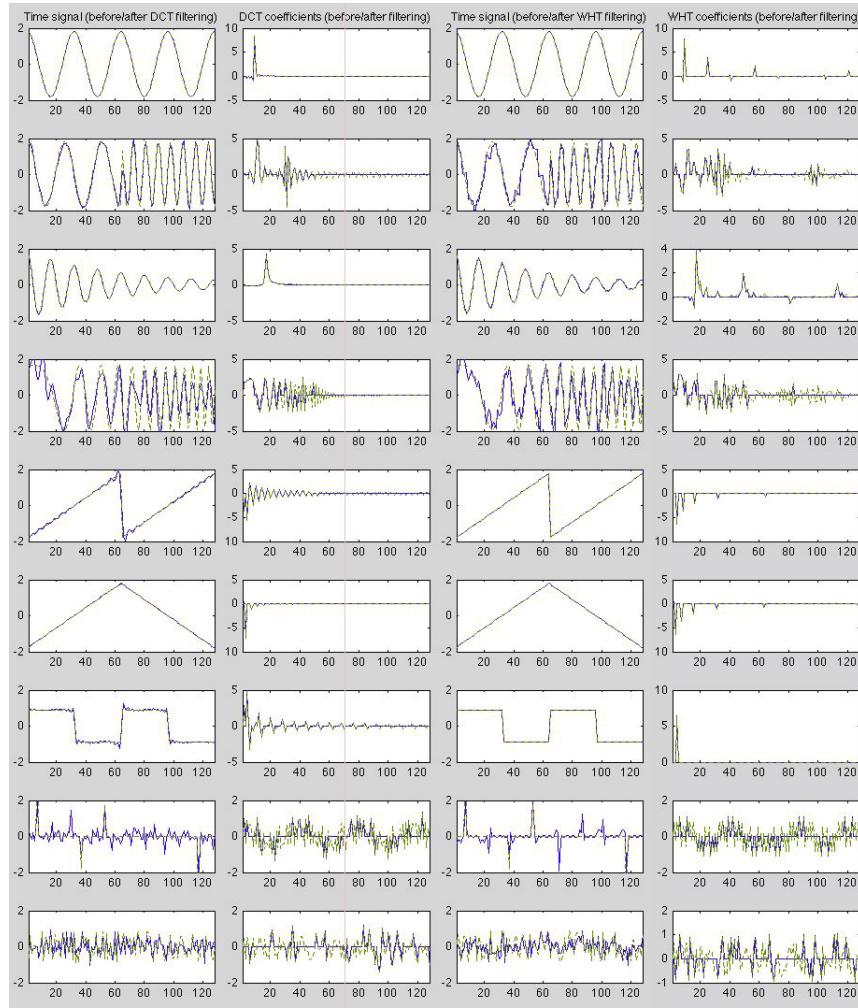


Figure 8.6 Compression of some typical signals by DCT and WHT

The 1st and 3rd columns show the time signals compared with their reconstructions based on only 20% of the transform coefficients, as shown in the 2nd and 4th columns for the DCT and WHT, respectively. In both time and transform domains, the signals before (dashed curves) and after (solid curves) the compression are shown for comparison.

where \mathbf{R}_n is rotation matrix of size $N = 2^n$ by which the $N/4$ -th row and $N/2$ -th row are rotated by an angle θ_n :

$$\mathbf{R}_n = \begin{bmatrix} 1 & & & \\ \ddots & & & \\ & 1 & & \\ & & \cos \theta_n & -\sin \theta_n \\ & & \sin \theta_n & \cos \theta_n \\ & & & 1 \\ & & & \ddots & & (2^{n-2} = N/4)\text{th row} \\ & & & & & \\ & & & & & 1 \\ & & & & & \ddots & (2^{n-1} = N/2)\text{th row} \\ & & & & & & 1 \end{bmatrix} \quad (8.50)$$

where

$$\begin{aligned}\cos \theta_n &= \left(\frac{2^{2n-2} - 1}{2^{2n} - 1} \right)^{1/2} = \sqrt{\frac{3N^2}{4N^2 - 4}} \\ \sin \theta_n &= \left(\frac{2^{2n-2} - 2^{2n-2}}{2^{2n} - 1} \right)^{1/2} = \sqrt{\frac{N^2 - 4}{4N^2 - 4}}\end{aligned}\quad (8.51)$$

Note that the trigonometric identity $\sin^2 \theta_n + \cos^2 \theta_n = 1$ is indeed satisfied.

The rotation matrix is obviously orthogonal $\mathbf{R}_n^T \mathbf{R}_n = \mathbf{I}_n$. Specially if $\theta_n = 0$ then $\mathbf{R}_n = \mathbf{I}_n$, and Eq.8.49 for the slant matrix becomes the same as Eq.8.3 for Hadamard matrix.

The slant transform matrix \mathbf{S}_n is real but not symmetric, and we can show that it is also orthogonal:

$$\mathbf{S}_n^T = \mathbf{S}_n^{-1}, \quad \text{i.e.,} \quad \mathbf{S}_n^T \mathbf{S}_n = \mathbf{S}_n \mathbf{S}_n^T = \mathbf{I}_n \quad (8.52)$$

Similar to the way we prove the orthogonality of the WHT matrix, here the orthogonality of the ST matrix \mathbf{S} can also be proven by induction. This is left for the reader as a homework problem.

As the slant matrix \mathbf{S}_n is closely related to the Hadamard matrix \mathbf{H}_n , the sequences of their corresponding rows are the same. The same re-ordering method given in Eq. 8.33 can be used to rearrange the rows of \mathbf{S}^T , i.e., the columns of \mathbf{S} , in ascending order of their sequences. Based on the recursion of Eq.8.49 and after conversion to the sequency order, the slant transform matrices of the next two levels for $n = 2$ and $n = 3$ can be obtained as:

$$\mathbf{S}_2^T = \frac{1}{2} \begin{bmatrix} 1.00 & 1.00 & 1.00 & 1.00 \\ 1.34 & 0.45 & -0.45 & -1.34 \\ 1.00 & -1.00 & -1.00 & 1.00 \\ 0.45 & -1.34 & 1.34 & -0.45 \end{bmatrix} \quad (8.53)$$

and

$$\mathbf{S}_3^T = \frac{1}{\sqrt{8}} \begin{bmatrix} 1.00 & 1.00 & 1.00 & 1.00 & 1.00 & 1.00 & 1.00 & 1.00 \\ 1.53 & 1.09 & 0.65 & 0.22 & -0.22 & -0.65 & -1.09 & -1.53 \\ 1.34 & 0.45 & -0.45 & -1.34 & -1.34 & -0.45 & 0.45 & 1.34 \\ 0.68 & -0.10 & -0.88 & -1.66 & 1.66 & 0.88 & 0.10 & -0.68 \\ 1.00 & -1.00 & -1.00 & 1.00 & 1.00 & -1.00 & -1.00 & 1.00 \\ 1.00 & -1.00 & -1.00 & 1.00 & -1.00 & 1.00 & 1.00 & -1.00 \\ 0.45 & -1.34 & 1.34 & -0.45 & -0.45 & 1.34 & -1.34 & 0.45 \\ 0.45 & -1.34 & 1.34 & -0.45 & 0.45 & -1.34 & 1.34 & -0.45 \end{bmatrix} \quad (8.54)$$

We make the following observations and comments:

- Unlike the Walsh-Hadamard matrix, the slant matrix $\mathbf{S}^T \neq \mathbf{S}$ is not symmetric;
- The sequences of the row vectors in matrix \mathbf{S}^T increase from 0 of the first row to $N - 1$ of the last one, i.e., these row vectors form a basis containing N basis vectors of space \mathbb{R}^N .

- In particular, the second row with sequency of 1 has a negative linear slope, thereby the name “slant” matrix.
- In general, we always treat the *column* vectors of an orthogonal transform matrix as the basis vectors of different frequencies/sequencies. This is why we have used \mathbf{S}^T in the discussion above, so that the slant transform matrix $\mathbf{S}_n = [\mathbf{s}_0, \dots, \mathbf{s}_{N-1}]$ is composed of N columns for the N basis vectors \mathbf{s}_n ($n = 0, \dots, N-1$) of sequency n .

8.2.2 Slant Transform and Its Fast Algorithm

Given an orthogonal matrix \mathbf{S}_n , an orthogonal transform of an N-D vector \mathbf{x} can be defined as:

$$\mathbf{X} = \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \mathbf{S}^T \mathbf{x} = \begin{bmatrix} \mathbf{s}_0^T \\ \vdots \\ \mathbf{s}_{N-1}^T \end{bmatrix} \mathbf{x}, \quad (8.55)$$

or in component form:

$$X[k] = \mathbf{s}_k^T \mathbf{x} = \langle \mathbf{s}_k, \mathbf{x} \rangle \quad (8.56)$$

i.e., $X[k]$ is the projection of the signal vector \mathbf{x} onto the k th basis vector \mathbf{s}_k . The inverse transform reconstructs the signal from its transform coefficients:

$$\mathbf{x} = \mathbf{S}\mathbf{X} = [\mathbf{s}_0, \dots, \mathbf{s}_{N-1}] \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \sum_{k=0}^{N-1} X[k] \mathbf{s}_k \quad (8.57)$$

Like the Walsh-Hadamard transform, the slant transform also has a fast algorithm with computational complexity of $O(N \log_2 N)$ instead of $O(N^2)$. This algorithm can be explained in the following example of $n = 3$. The slant transform of a vector \mathbf{x} of size $N = 2^3 = 8$ is:

$$\mathbf{X} = \mathbf{S}_3^T \mathbf{x} = \frac{1}{\sqrt{2}} \mathbf{R}_3 \begin{bmatrix} \mathbf{S}_2^T & \mathbf{S}_2^T \\ \mathbf{S}_2^T & -\mathbf{S}_2^T \end{bmatrix} \begin{bmatrix} x[0] \\ \vdots \\ x[3] \\ x[4] \\ x[5] \\ x[6] \\ x[7] \end{bmatrix} = \frac{1}{\sqrt{2}} \mathbf{R}_3 \begin{bmatrix} \mathbf{S}_2^T \mathbf{x}_1 \\ \mathbf{S}_2^T \mathbf{x}_2 \end{bmatrix} \quad (8.58)$$

where

$$\mathbf{x}_1 = \begin{bmatrix} x[0] \\ \vdots \\ x[3] \end{bmatrix} + \begin{bmatrix} x[4] \\ \vdots \\ x[7] \end{bmatrix}, \quad \mathbf{x}_2 = \begin{bmatrix} x[0] \\ \vdots \\ x[3] \end{bmatrix} - \begin{bmatrix} x[4] \\ \vdots \\ x[7] \end{bmatrix} \quad (8.59)$$

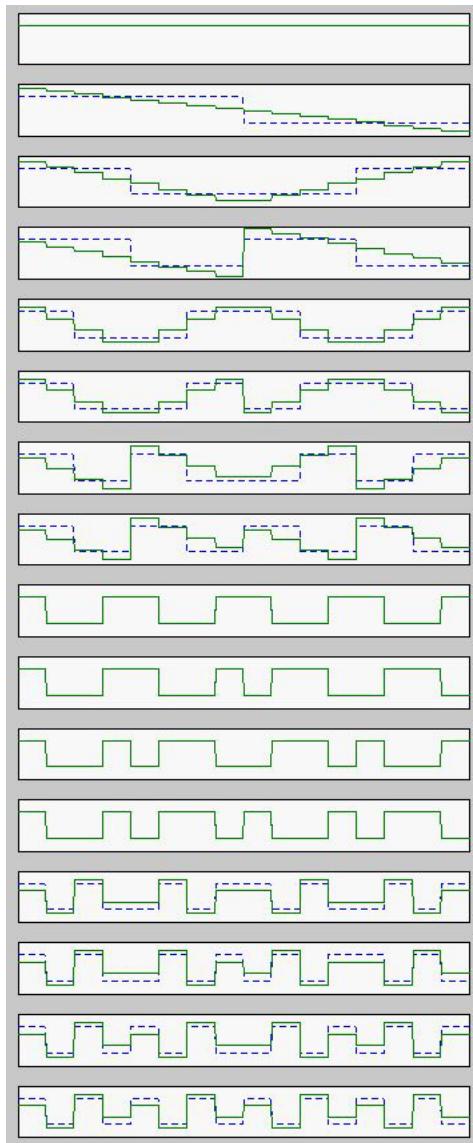


Figure 8.7 Comparison of the basis vectors of slant (solid lines) and Hadamard (dashed lines) transform

We see that an 8-point slant transform is converted into two 4-point slant transforms, each of which can be converted in turn to two 2-point transforms. This recursive process is illustrated in the diagram in Fig.8.8. The three nested boxes (dashed line) represent three levels of recursion for the 8-point, 4-point and 2-point transforms, respectively. In general, an N -point transform can be implemented by this algorithm in $\log_2 N$ stages each requiring $O(N)$ operations, i.e., the total complexity is $O(N \log_2 N)$. This algorithm is almost identical to the

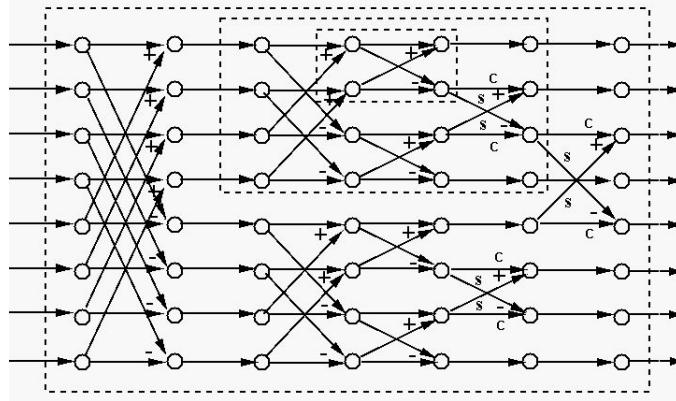


Figure 8.8 A recursive algorithm of fast slant transform

The three nested boxes (dashed line) are for 8, 4 and 2-point transforms, respectively. Letter c and s represents $\cos \theta_n$ and $\sin \theta_n$ for the rotation for each of the transforms (except for $N = 2$).

WHT algorithm shown in Fig.8.4, except an additional rotation for two of the rows at each level.

While the algorithm can be implemented in a manner very similar to the WHT code discussed previously, here we present an alternative implementation based on recursion, which fits the algorithm most naturally.

```

slantf(float *x, int N)
{
    int i,j,k,l,m,n;
    float c,s,u,v,*y1,*y2;
    y1=(float*)malloc(N/2 * sizeof(float));
    y2=(float*)malloc(N/2 * sizeof(float));
    if (N==2) {           // 2-point transform
        u=x[0]; v=x[1];
        x[0]=(u+v)/Sqrt2;
        x[1]=(u-v)/Sqrt2;
    }
    else {
        for (n=0; n<N/2; n++) {
            y1[n]=x[n]+x[N/2+n];
            y2[n]=x[n]-x[N/2+n];
        }
        slantf(y1,N/2);      // recursion
        slantf(y2,N/2);
        for (n=0; n<N/2; n++) {
            x[n]=y1[n]/Sqrt2;
        }
    }
}
```

```

    x[N/2+n]=y2[n]/Sqrt2;
}
w=4*N*N-4;
c=sqrt(3*N*N/w);
s=sqrt((N*N-4)/w);
u=x[N/4]; v=x[N/2];
x[N/4]=c*u-s*v;           // rotation
x[N/2]=s*u+c*v;
}
free(y1); free(y2);
}

```

The inverse transform can be implemented by reversing the steps and operations both mathematically and order-wise in the forward transform:

```

slanti(float *x, int N)
{
    int i,j,k,l,m,n;
    float c,s,u,v,w,*y1,*y2;
    y1=(float*)malloc(N/2 * sizeof(float));
    y2=(float*)malloc(N/2 * sizeof(float));
    if (N==2) {               // 2-point transform
        u=x[0]; v=x[1];
        x[0]=(u+v)/Sqrt2;
        x[1]=(u-v)/Sqrt2;
    }
    else {
        w=4*N*N-4;
        c=sqrt(3*N*N/w);
        s=sqrt((N*N-4)/w);
        u=x[N/4]; v=x[N/2];
        x[N/4]=c*u+s*v;           // rotation
        x[N/2]=c*v-s*u;
        for (n=0; n<N/2; n++) {
            y1[n]=x[n]*Sqrt2;
            y2[n]=x[N/2+n]*Sqrt2;
        }
        slanti(y1,N/2);          // recursion
        slanti(y2,N/2);
        for (n=0; n<N/2; n++) {
            x[n]=(y1[n]+y2[n])/2;
            x[N/2+n]=(y1[n]-y2[n])/2;
        }
    }
}

```

```

    free(y1); free(y2);
}

```

Example 8.3: The slant transform of an 8-point signal vector $\mathbf{x} = [0, 0, 2, 3, 4, 0, 0, 0]^T$ can be obtained by matrix multiplication:

$$\mathbf{X} = \mathbf{S}_3^T \mathbf{x} = [3.18, 0.39, -3.64, -0.03, 1.77, -1.06, -0.16, 1.11]^T \quad (8.60)$$

where \mathbf{S}_3 is given in Eq.8.54. The inverse transform will bring the original signal back: $\mathbf{x} = \mathbf{S}_3 \mathbf{X} = [0, 0, 2, 3, 4, 0, 0, 0]^T$.

8.3 The Haar Transform

8.3.1 Continuous Haar Transform

Similar to the Walsh-Hadamard transform, the Haar transform is yet another orthogonal transform defined by a set of rectangular shaped basis functions. However, compared to all orthogonal transform methods considered so far, the Haar transform has some unique significance as it is also a special type of the wavelet transforms to be discussed in Chapter 11.

The family of Haar functions $h_k(t)$, ($k = 0, 1, 2, \dots$) are defined on the interval $0 \leq t \leq 1$. Except $h_0(t) = 1$ which is defined as a constant, the shape of the k th function $h_k(t)$ for $k > 0$ are determined by two parameters p and q , which are related to k by:

$$k = 2^p + q \quad (8.61)$$

In other words, p and q are uniquely determined so that $2^p < k$ is the highest power of 2 contained in k , and $q = k - 2^p$ is the remainder. For example, the values of p and q corresponding to $k = 1, \dots, 15$ are shown in the table:

k	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
p	0	1	1	2	2	2	2	3	3	3	3	3	3	3	3
q	0	0	1	0	1	2	3	0	1	2	3	4	5	6	7

(8.62)

Now the family of Haar functions can be defined as:

- When $k = 0$:

$$h_0(t) = 1, \quad (0 \leq t < 1) \quad (8.63)$$

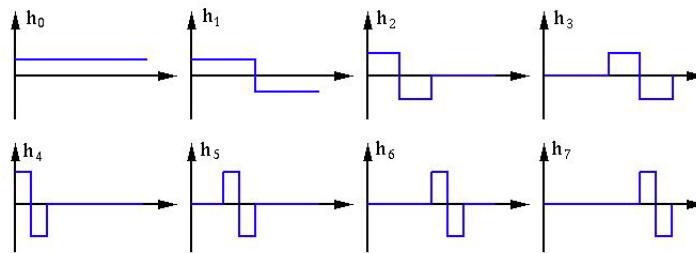


Figure 8.9 The 8 basis functions for the Haar transform

- When $k > 0$, $h_k(t)$ is defined in terms of p and q :

$$h_k(t) = \begin{cases} \sqrt{2^p} & q/2^p \leq t < (q + 0.5)/2^p \\ -\sqrt{2^p} & (q + 0.5)/2^p \leq t < (q + 1)/2^p \\ 0 & \text{else} \end{cases} \quad (8.64)$$

The first $N = 8$ Haar functions are shown in Fig.8.9. We see that the Haar functions $h_k(t)$ for all $k > 0$ contain a single prototype shape composed of a square wave followed by its negative copy, with the two parameters p specifying the magnitude and width (or scale) of the shape and q specifying the position (translation) of the shape. For example, if $k = 5$, then $p = 2$, $q = 1$, and we have:

$$h_5(t) = \begin{cases} 2 & 2/8 \leq t < 3/8 \\ -2 & 3/8 \leq t < 4/8 \\ 0 & \text{else} \end{cases} \quad (8.65)$$

These Haar functions are obviously orthonormal:

$$\langle h_k(t), h_l(t) \rangle = \int_0^1 h_k(t) h_l(t) dt = \delta[k - l] \quad (8.66)$$

and they can be used as the basis functions to span a function space over $0 \leq t < 1$. A signal $x(t)$ in this space can be expressed as a linear combination of these Haar functions:

$$x(t) = \sum_{k=0}^{\infty} X[k] h_k(t) \quad (8.67)$$

where the k th coefficient $X[k]$ can be obtained as the projection of $x(t)$ onto the k th basis function $h_k(t)$:

$$X[k] = \langle x(t), h_k(t) \rangle = \int_0^1 x(t) h_k(t) dt \quad (8.68)$$

When $k = 0$, the coefficient

$$X[0] = \int_0^1 x(t) h_0(t) dt = \int_0^1 x(t) dt \quad (8.69)$$

represents the average or DC component of the signal, same as all orthogonal transforms discussed before. When $k > 0$, the coefficient $X[k]$ represents three specific aspects of the signal characteristics:

- certain type of detailed features contained in the signal, in the form of the difference between two consecutive segments of the signal
- the time interval during which such detailed features occur, and
- the time scale of such features

For example, a large value (either positive or negative) of the coefficient $X[3]$ for the basis function $h_3(t)$ would indicate that the signal value has some significant variation of the scale of half of its duration in the second half of its duration.

It is interesting to compare the Haar transform with other orthogonal transforms such as the Fourier, cosine, Walsh-Hadamard, and slant transform discussed before. What all of these transforms, as well as the Haar transform, have in common is that their coefficients represent some type of details contained in the signal, in terms of different frequencies (Fourier transform and cosine transform), sequencies (Walsh-Hadamard transform), or scales (Haar transform), in the sense that more detailed information is represented by coefficients for higher frequencies, sequencies, or scales. However, none of these transforms is able to indicate when in time such details occur, except the Haar transform, which represents not only the details of different scales, but also their temporal positions. However, we note that this additional capability is gained with the cost of much reduced number of scale levels. All N -point orthogonal transforms can represent N different frequencies/sequencies, but an N -point Haar transform can only represent $\log_2 N$ different scale levels. Due to such different behaviors the Haar transform is in fact also a special form of the wavelet transform to be discussed in Chapter 11.

8.3.2 Discrete Haar Transform

The discrete Haar transform (DHT) is defined based on the family of Haar functions. Specifically, by sampling each of the first N Haar functions $h_k(t)$ ($k = 0, \dots, N - 1$) at time moments $t = n/N$ ($n = 0, 1, 2, \dots, N - 1$), we get N orthogonal vectors. Moreover, if a scaling factor $1/\sqrt{N}$ is included, these vectors become orthonormal:

$$\langle \mathbf{h}_k, \mathbf{h}_l \rangle = \mathbf{h}_k^T \mathbf{h}_l = \delta[k - l] \quad (8.70)$$

These N orthonormal vectors form a basis that spans the N -dimensional vector space, and they form an N by N DHT matrix \mathbf{H} (not to be confused with the WHT matrix):

$$\mathbf{H} = [\mathbf{h}_0, \dots, \mathbf{h}_{N-1}], \quad \text{or} \quad \mathbf{H}^T = \begin{bmatrix} \mathbf{h}_0^T \\ \vdots \\ \mathbf{h}_{N-1}^T \end{bmatrix} \quad (8.71)$$

which is obviously real and orthonormal (but not symmetric):

$$\mathbf{H} = \mathbf{H}^*, \quad \mathbf{H}^{-1} = \mathbf{H}^T, \quad \text{i.e.} \quad \mathbf{H}^T \mathbf{H} = \mathbf{I} \quad (8.72)$$

As some examples, the DHT matrices corresponding to $N = 2, 4, 8$ are listed below.

- When $N = 2$, the 2×2 DHT matrix is identical to the transform matrices for all other discrete transforms including DFT, DCT and WHT:

$$\mathbf{H}_1^T = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 0.71 & 0.71 \\ 0.71 & -0.71 \end{bmatrix} \quad (8.73)$$

The first row represents the average of the signal, while the second represents the difference between the first and second halves of the signal, same for all transform methods.

- When $N = 4$

$$\mathbf{H}_2^T = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix} = \begin{bmatrix} 0.50 & 0.50 & 0.50 & 0.50 \\ 0.50 & 0.50 & -0.50 & -0.50 \\ 0.71 & -0.71 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.71 & -0.71 \end{bmatrix} \quad (8.74)$$

The DCT matrix \mathbf{C} and the Walsh-ordered WHT matrix \mathbf{H}_w are also listed below for comparison:

$$\mathbf{C}^T = \begin{bmatrix} 0.50 & 0.50 & 0.50 & 0.50 \\ 0.65 & 0.27 & -0.27 & -0.65 \\ 0.50 & -0.50 & -0.50 & 0.50 \\ 0.27 & -0.65 & 0.65 & -0.27 \end{bmatrix}, \quad \mathbf{H}_w^T = \begin{bmatrix} 0.50 & 0.50 & 0.50 & 0.50 \\ 0.50 & 0.50 & -0.50 & -0.50 \\ 0.50 & -0.50 & -0.50 & 0.50 \\ 0.50 & -0.50 & 0.50 & -0.50 \end{bmatrix} \quad (8.75)$$

We see that the first rows of all three matrices \mathbf{H} , \mathbf{C} and \mathbf{H}_w are identical, representing the DC component of the signal. The elements of their second rows have the same polarities (but of different values), representing the difference between the first and second halves of the signal. However, their third and forth rows are quite different. For DCT and WHT, these two rows represent progressively higher frequency or sequency components in the signal, but in the case of the DHT, these rows represent the same level of details (variations) at a finer scale than the second row, as well as their different temporal locations (either in the first or second half).

- When $N = 8$, we have

$$\mathbf{H}_3^T = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{bmatrix} \begin{array}{l} 0 \varphi_{0,0}(t) \\ 1 \psi_{0,0}(t) \\ 2 \psi_{1,0}(t) \\ 3 \psi_{1,1}(t) \\ 4 \psi_{2,0}(t) \\ 5 \psi_{2,1}(t) \\ 6 \psi_{2,2}(t) \\ 7 \psi_{2,3}(t) \end{array} \quad (8.76)$$

It is obvious that the additional four rows represent still more detailed and finer signal variations and their temporal positions at a finer scale than the previous two rows. Note that each row is also labeled as a function ($\varphi_0(t)$ for the first row and $\psi_{p,q}(t)$ for the rest) on the right. The significance of these labelings will be clear in the future when we discuss discrete wavelet transforms.

Now any N -point signal vector $\mathbf{x} = [x[0], \dots, x[N-1]]^T$ can be expressed as a linear combination of the column vectors \mathbf{h}_k ($k = 0, \dots, N-1$) of the DHT matrix \mathbf{H} :

$$\mathbf{x} = \mathbf{H}\mathbf{X} = [\mathbf{h}_0, \dots, \mathbf{h}_{N-1}] \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \sum_{k=0}^{N-1} X[k] \mathbf{h}_k \quad (8.77)$$

This is the inverse discrete Haar transform (IDHT), where the k th coefficient $X[k]$ for the vector \mathbf{h}_k can be obtained as the projection of the signal vector \mathbf{x} onto the k th basis vector \mathbf{h}_k :

$$X[k] = \langle \mathbf{x}, \mathbf{h}_k \rangle = \mathbf{h}_k^T \mathbf{x} \quad (k = 0, 1, \dots, N-1) \quad (8.78)$$

or in matrix form:

$$\mathbf{X} = \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \mathbf{H}^{-1} \mathbf{x} = \mathbf{H}^T \mathbf{x} = \begin{bmatrix} \mathbf{h}_0^T \\ \vdots \\ \mathbf{h}_{N-1}^T \end{bmatrix} \mathbf{x} \quad (8.79)$$

This is the forward discrete Haar transform (DHT), which can also be obtained by pre-multiplying \mathbf{H}^{-1} on both sides of the IDHT equation above. The DHT pair can be written as:

$$\begin{cases} \mathbf{X} = \mathbf{H}^T \mathbf{x} & \text{(forward)} \\ \mathbf{x} = \mathbf{H}\mathbf{X} & \text{(inverse)} \end{cases} \quad (8.80)$$

Example 8.4: The Haar transform coefficients of an 8-point signal $\mathbf{x} = [0, 0, 2, 3, 4, 0, 0, 0]^T$ can be obtained by the DHT as:

$$\mathbf{X} = \mathbf{H}^T \mathbf{x} = [3.18, 0.35, -2.50, 2.0, 0.0, -0.71, 2.83, 0.0]^T \quad (8.81)$$

where the 8-point Haar transform matrix is given in Eq.8.76. Same as in the DCT, WHT and ST, $X[0] = 3.18$ and $X[1] = 0.35$ represent respectively the sum and difference between the first and second halves of the signal. However, the interpretations of the remaining DHT coefficients are quite different from the DCT and WHT. $X[2] = -2.5$ represents the difference between the first and second quarters in the first half of the signal, while $X[3] = 2$ represents the difference between the third and forth quarters in the second half of the signal. Similarly, $X[4], \dots, X[7]$ represent the next level of details in terms of the difference between two consecutive eighths of the signal in each of the four quarters of the signal.

The signal vector is reconstructed by the inverse transform IDHT which expresses the signal as a linear combination of the basis functions, as shown in Eq.8.77.

8.3.3 Computation of Discrete Haar Transform

The computational complexity of an N -point discrete Haar transform implemented as a matrix multiplication is $O(N^2)$. However, a fast algorithm with linear complexity $O(N)$ exists for both DHT and IDHT, as illustrated in Fig.8.10 for the 8-point DHT transform. The forward transform $\mathbf{X} = \mathbf{H}_3^T \mathbf{x}$ can be written in matrix form as:

$$\begin{aligned} \begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ X[3] \\ X[4] \\ X[5] \\ X[6] \\ X[7] \end{bmatrix} &= \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \\ x[4] \\ x[5] \\ x[6] \\ x[7] \end{bmatrix} \\ &= \begin{bmatrix} (1 & 1 & 1 & 1 & 1 & 1 & 1) / \sqrt{2}^3 \\ (1 & 1 & 1 & 1 & -1 & -1 & -1) / \sqrt{2}^3 \\ (1 & 1 & -1 & -1 & 0 & 0 & 0) / \sqrt{2}^2 \\ (0 & 0 & 0 & 0 & 1 & 1 & -1) / \sqrt{2}^2 \\ (1 & -1 & 0 & 0 & 0 & 0 & 0) / \sqrt{2} \\ (0 & 0 & 1 & -1 & 0 & 0 & 0) / \sqrt{2} \\ (0 & 0 & 0 & 0 & 1 & -1 & 0) / \sqrt{2} \\ (0 & 0 & 0 & 0 & 0 & 0 & 1 & -1) / \sqrt{2} \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \\ x[4] \\ x[5] \\ x[6] \\ x[7] \end{bmatrix} \end{aligned} \quad (8.82)$$

By inspection of this matrix multiplication, we see that each of the last four coefficients $X[4], \dots, X[7]$ in the second half of vector \mathbf{X} can be obtained as the difference between a pair of two signal samples, e.g., $X[4] = (x[0] - x[1])/\sqrt{2}$. Similarly, each of the last two coefficients $X[2]$ and $X[3]$ of the first half of \mathbf{X} can

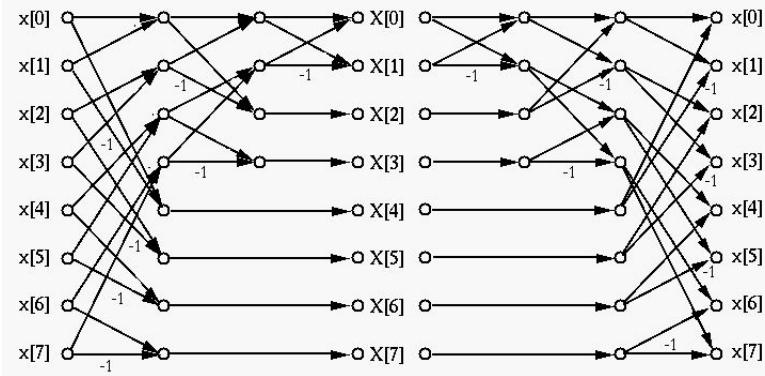


Figure 8.10 The fast Haar transform algorithm

The forward DHT transform shown on the left of the diagram converts the signal \mathbf{x} to its DHT coefficients \mathbf{X} in the middle, while the inverse transform IDHT shown on the right converts \mathbf{X} back to time domain (reconstruction).

be obtained as the difference between two sums of two signal components, e.g., $X[2] = [(x[0] + x[1]) - (x[2] + x[3])] / 2$. This process can be carried out recursively as shown on the left of Fig.8.10, each performing some additions and subtractions on the first half of the data points produced in the previous stage, and in $\log_2 8 = 3$ consecutive stages, the N DHT coefficients $X[0], \dots, X[7]$ can be obtained. Moreover, if the results of each stage are divided by $\sqrt{2}$, the normalization of the transform can also be taken care of.

The inverse transform $\mathbf{x} = \mathbf{H}_3 \mathbf{X}$ can also be written in matrix form:

$$\begin{aligned}
 \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \\ x[4] \\ x[5] \\ x[6] \\ x[7] \end{bmatrix} &= \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & \sqrt{2} & 0 & 2 & 0 & 0 & 0 \\ 1 & 1 & \sqrt{2} & 0 & -2 & 0 & 0 & 0 \\ 1 & 1 & -\sqrt{2} & 0 & 0 & 2 & 0 & 0 \\ 1 & 1 & -\sqrt{2} & 0 & 0 & -2 & 0 & 0 \\ 1 & -1 & 0 & \sqrt{2} & 0 & 0 & 2 & 0 \\ 1 & -1 & 0 & \sqrt{2} & 0 & 0 & -2 & 0 \\ 1 & -1 & 0 & -\sqrt{2} & 2 & 0 & 0 & 2 \\ 1 & -1 & 0 & -\sqrt{2} & 0 & 0 & 0 & -2 \end{bmatrix} \begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ X[3] \\ X[4] \\ X[5] \\ X[6] \\ X[7] \end{bmatrix} \\
 &= \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & -1 & 0 & 0 & 0 \\ 1 & 1 & -1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & -1 & 0 & 0 & -1 & 0 & 0 \\ 1 & -1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & -1 & 0 & 1 & 0 & 0 & -1 & 0 \\ 1 & -1 & 0 & -1 & 0 & 0 & 0 & 1 \\ 1 & -1 & 0 & -1 & 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} X[0]/\sqrt{2^3} \\ X[1]/\sqrt{2^3} \\ X[2]/\sqrt{2^2} \\ X[3]/\sqrt{2^2} \\ X[4]/\sqrt{2} \\ X[5]/\sqrt{2} \\ X[6]/\sqrt{2} \\ X[7]/\sqrt{2} \end{bmatrix} \quad (8.83)
 \end{aligned}$$

By inspection we see that this matrix multiplication can also be carried out in $\log_2 8 = 3$ stages as shown on the right side of the diagram in Fig.8.10. Again, the output of each stage needs to be divided by $\sqrt{2}$.

Moreover, from this example of 8-point transform, we can also obtain the computational complexity of the DHT (as well as the fast wavelet transform to be considered in Chapter 11). While the fast algorithms for all orthogonal transforms considered previously have the same complexity $O(N \log_2 N)$, the fast algorithm for DHT shown in Fig.8.10 is even more efficient with a complexity $O(N)$. The proof of this linear complexity can be easily obtained and is left for the reader as a homework problem.

The C code for both the forward and inverse discrete Haar transform is listed below:

```
dht(x,N,inverse)
    float *x;
    int N,inverse;
{ int i,n;
    float *y,r2=sqrt(2.0);
    y=(float *)malloc(N*sizeof(float));
    if (inverse) {
        n=1;
        while(n<N) {
            for (i=0; i<n; i++) {
                y[2*i] =(x[i]+x[i+n])/r2;
                y[2*i+1]=(x[i]-x[i+n])/r2;
            }
            for (i=0; i<n*2; i++) x[i]=y[i];
            n=n*2;
        }
    }
    else {
        n=N;
        while(n>1) {
            n=n/2;
            for (i=0; i<n; i++) {
                y[i] =(x[2*i]+x[2*i+1])/r2;
                y[i+n]=(x[2*i]-x[2*i+1])/r2;
            }
            for (i=0; i<n*2; i++) x[i]=y[i];
        }
        free(y);
    }
}
```

8.3.4 Filter Bank Implementation

The fast algorithm of the Haar transform can also be viewed as a special case of the filter bank algorithm for general wavelet transforms, to be discussed in Chapter 11. Here we briefly discuss such an implementation as a preview of the filter bank idea. To see how this algorithm works, we first consider the convolution of a signal sequence $x[n]$ with some convolution kernel $h[n]$:

$$x'[n] = x[n] * h[n] = \sum_m h[m]x[n-m] \quad (8.84)$$

In particular, for the Haar transform, we consider four different 2-point convolution kernels:

- $h_0[0] = h_0[1] = 1/\sqrt{2}$
- $h_1[0] = 1/\sqrt{2}, h_1[1] = -1/\sqrt{2}$
- $g_0[0] = g_0[1] = 1/\sqrt{2}$
- $g_1[0] = -1/\sqrt{2}, g_1[1] = 1/\sqrt{2}$

Note that $g_i[n]$ is the time-reversed version of $h_i[n]$ ($i = 0, 1$), i.e., the order of the elements in the 2-point sequence is reversed (the two elements of g_0 and h_0 are identical). Depending on the kernel, the convolution above can be considered as either a highpass or lowpass filter. Specifically, for kernel h_0 (or g_0), we have

$$y[n] = x[n] * h_0[n] = \sum_{m=0}^1 h_0[m]x[n-m] = \frac{1}{\sqrt{2}}(x[n-1] + x[n]) \quad (8.85)$$

This can be considered as a lowpass filter as the output $y[n]$ represents the average of any two consecutive data points $x[n-1]$ and $x[n]$ (corresponding to low frequencies). On the other hand, if the kernel is h_1 , then

$$y[n] = x[n] * h_1[n] = \sum_{m=0}^1 h_1[m]x[n-m] = \frac{1}{\sqrt{2}}(x[n-1] - x[n]) \quad (8.86)$$

This can be considered as a highpass filter as the output $y[n]$ represents the difference of the two consecutive data points (corresponding to high frequencies). Finally, if the kernel is g_1 , the convolution is also a highpass filter:

$$y[n] = x[n] * g_1[n] = \frac{1}{\sqrt{2}}(x[n] - x[n-1]) = -x[n] * h_1[n] \quad (8.87)$$

Due to the convolution theorem of Z-transform, these convolutions can also be represented as multiplications in Z-domain:

$$Y(z) = H_i(z)X(z), \quad Y(z) = G_i(z)X(z), \quad (i = 0, 1) \quad (8.88)$$

Now the forward transform of the fast DHT shown on the left of Fig. 8.10 can be considered as a recursion of the following two operations:

- Operation \mathcal{A} (average or approximation): a lowpass filter implemented as $y[n] = x[n] * h_0[n]$, followed by downsampling (every other point in $y[n]$ is eliminated);
- Operation \mathcal{D} (difference or detail): a highpass filter implemented as $y[n] = x[n] * h_1[n]$, also followed by downsampling.

For example, operation \mathcal{A} applied to a set of 8-point sequence $x[0], \dots, x[7]$ will generate a 4-point sequence containing $x[0] + x[1]$, $x[2] + x[3]$, $x[4] + x[5]$, and $x[6] + x[7]$ (all divided by $\sqrt{2}$) representing the local average (or approximation) of the signal. When operation \mathcal{D} is applied to the same input, it will generate a different 4-point sequence containing $x[0] - x[1]$, $x[2] - x[3]$, $x[4] - x[5]$, and $x[6] - x[7]$ (all divided by $\sqrt{2}$) representing the local difference (or details) of the signal.

In this filter bank algorithm, this pair of operations \mathcal{A} and \mathcal{D} is applied first to the N -point signal $x[n]$ ($n = 0, \dots, N - 1$), and then recursively to the output of operation \mathcal{A} in the previous recursion. As the data size is reduced by half after each recursion, this process can be carried out $\log_2 N$ times to generate all N transform coefficients. This is the filter bank implementation of the DHT, as illustrated on left of Fig.8.11.

The inverse transform of the fast algorithm (right half of Fig. 8.10) can also be viewed as a recursion of two operations:

- Operation \mathcal{A} : a lowpass filter implemented as $y[n] = x[n] * g_0[n]$, applied to the upsampled version of the data (with a zero inserted between every two consecutive data points, also in front of the first sample and after the last one);
- Operation \mathcal{D} : a highpass filtered by $y[n] = x[n] * g_1[n]$, applied to the upsampled input data.

For example, when operation \mathcal{A} is applied to $X[0]$, it will first be upsampled to become $0, X[0], 0$, which is then convolved with $g_0[n]$ to generate a sequence with two elements $X[0], X[0]$. Also, when operation \mathcal{D} is applied to $X[1]$, it will be upsampled to become $0, X[1], 0$, which is convolved with $g_1[n]$ to generate a sequence $X[1], -X[1]$. The corresponding elements of these two sequences are then added to generate a new sequence $X[0] + X[1], X[0] - X[1]$. In the next level of recursion, operation \mathcal{A} will be applied to this 2-point sequence, while operation \mathcal{D} is applied to the next two data points $X[2], X[3]$, and their outputs, two 4-point sequences, are added again. This recursion is also carried out $\log_2 N$ times until all N data points $x[0], \dots, x[N - 1]$ are reconstructed. This is the filter bank implementation of the IDHT, as illustrated on the right of Fig.8.11.

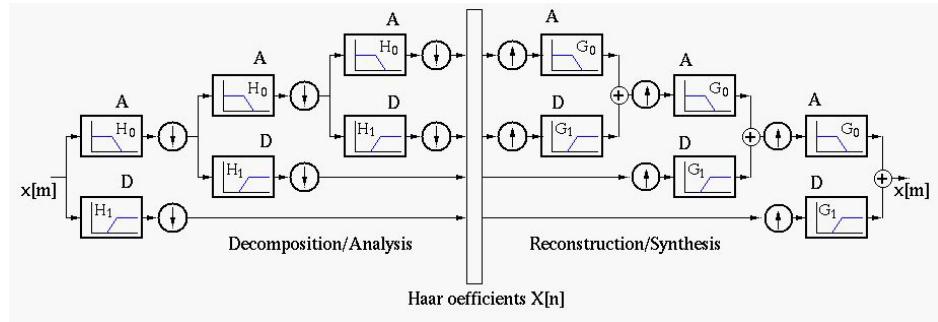


Figure 8.11 Filter bank implementation of DHT

H_0 and G_0 are lowpass filters and H_1 and G_1 are highpass filters. The up and down arrows represent upsampling and downsampling, respectively.

8.4 Two-dimensional Transforms

Same as the discrete Fourier and cosine transforms, all three of the transform methods (Walsh-Hadamard, Slant and Haar transforms) discussed above can also be applied to a 2-D signal $x[m, n]$ ($m = 0, \dots, M - 1, n = 0, \dots, N - 1$), such as an image, for purposes such as feature extraction, filtering and data compression. For convenience, in the following we will represent any of the three orthogonal matrices considered above by a generic orthogonal matrix \mathbf{A} . The forward and inverse 2-D transform of a 2-D signal are defined respectively as:

$$\begin{cases} \mathbf{X} = \mathbf{A}^T \mathbf{x} \mathbf{A} & \text{(forward)} \\ \mathbf{x} = \mathbf{A} \mathbf{X} \mathbf{A}^T & \text{(inverse)} \end{cases} \quad (8.89)$$

where both the 2-D signal \mathbf{x} and its spectrum \mathbf{X} are $M \times N$ matrices, and the pre-multiplication matrix \mathbf{A} is $M \times M$ for the column transforms, while the post-multiplication matrix \mathbf{A} is $N \times N$ for the row transforms. The inverse transform (second equation) expresses the given 2-D signal \mathbf{x} as a linear combination of a set of N^2 2-D basis functions:

$$\begin{aligned} \mathbf{x} &= [\mathbf{a}_0, \dots, \mathbf{a}_{M-1}] \begin{bmatrix} X[0, 0] & \cdots & X[0, N-1] \\ \vdots & \ddots & \vdots \\ X[M-1, 0] & \cdots & X[M-1, N-1] \end{bmatrix} \begin{bmatrix} \mathbf{a}_0^T \\ \vdots \\ \mathbf{a}_{N-1}^T \end{bmatrix} \\ &= \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} X[k, l] \mathbf{a}_k \mathbf{a}_l^T = \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} X[k, l] \mathbf{B}_{kl} \end{aligned} \quad (8.90)$$

where $\mathbf{B}_{kl} = \mathbf{a}_k \mathbf{a}_l^T$ is the kl-th 2-D ($M \times N$) basis function, weighted by the corresponding coefficient $X[k, l]$. Same as in the cases of DFT (Eq.4.262) and DCT (Eq.7.120), this coefficient can be obtained as the projection (inner product) of

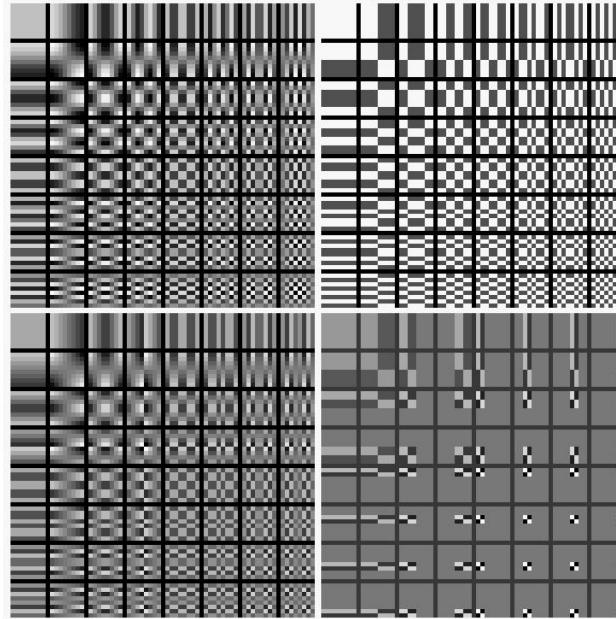


Figure 8.12 The basis functions for 2-D DCT (top-left), WHT (top-right), ST (lower-left), and DHT (lower-right)

the 2-D signal \mathbf{x} onto the kl -th 2-D basis function \mathbf{B}_{kl} :

$$\begin{aligned} X[k, l] &= \mathbf{a}_k^T \begin{bmatrix} x[0, 0] & \cdots & x[0, N-1] \\ \vdots & \ddots & \vdots \\ x[M-1, 0] & \cdots & x[M-1, N-1] \end{bmatrix} \mathbf{a}_l \\ &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m, n] B_{kl}[m, n] = \langle \mathbf{x}, \mathbf{B}_{kl} \rangle \end{aligned} \quad (8.91)$$

When $M = N = 8$, the $8 \times 8 = 64$ such 2-D basis functions corresponding to Walsh-Hadamard (WHT), slant (ST), and Haar (DHT) are shown in Fig.8.12, in comparison with those of the 2-D DCT. In all four transforms the DC component is at the top-left corner, and the farther away from the corner, the higher frequency/sequency contents or scales of details are represented. Also note that the spatial positions are represented in the Haar basis.

All of these transform methods can be used for filtering. Fig.8.13 shows both the low-pass and high-pass filtering effects in both spatial domain and spatial frequency domain for each of the transform methods. We can also see that all of these transforms have the general property of compacting the signal energy into a small number of low frequency/sequency/scale components. In the low-pass filtering examples, only about one percent of the transform coefficients are kept after filtering in the transform domain of DCT, WHT, ST and DHT, but they carry, respectively, 96.4%, 94.8%, 95.5% and 93% of the total signal energy.

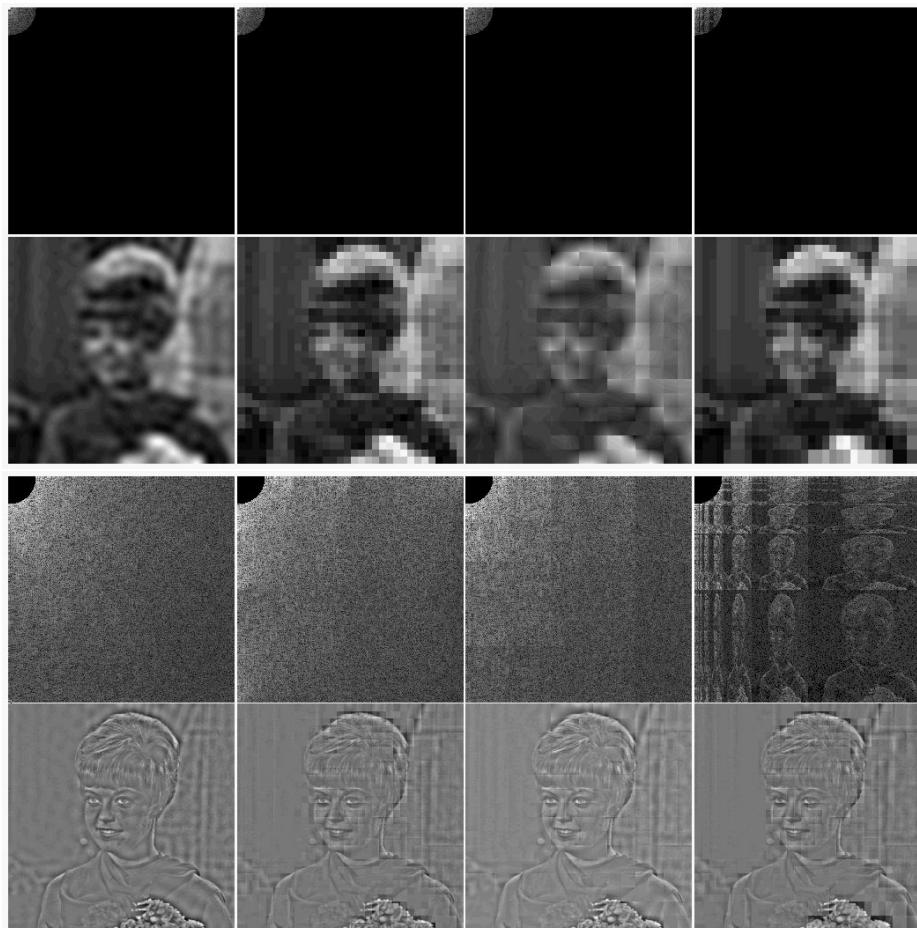


Figure 8.13 Low-pass and high-pass filtering based DCT, WHT, ST and DHT (from left to right)

The filtered spectrum is given in the first (low-pass) and third (high-pass) row, the corresponding filtered image is given directly below the spectrum.

Therefore all of these transform methods lend themselves to data compression, same as the Fourier transform.

As these transform methods are based on different basis functions, they are may be suitable for different types of signals. Most obviously, like the DFT, the DCT is based on sinusoidal basis functions and is therefore suitable for representing signals that are smooth in nature. However, it is also possible that in some specific applications other transform methods may be more suitable, as the signals of interest may be more effectively represented by a particular type of basis functions other than sinusoids. For example, the WHT may be more suitable to use if the signal is square-wave like in nature and may be most

effectively represented by a small subset of the WHT basis functions, so that the corresponding transform coefficients may contain most of the signal energy.

Also we make some special note regarding the Haar transform. Same as all other 2-D transforms, the first basis function, the top-left corner in Fig.8.12, is a constant representing the DC component of the 2-D signal. However, the rest of the basis functions are quite different. Most obviously, the last 16 basis functions in the lower-right quarter of the 2-D spectrum represent not only the same (highest) level of details in the signal, but also their spatial positions. This contrasts strongly with the spectra of all other transforms, which represent progressively higher spatial frequencies/sequencies (for signal details at different levels) without any indication in terms of their spatial positions. As noted before, this capability of position representation in the spectrum is gained with the cost of much reduced number of scale levels.

8.5 Homework Problems

1. Prove the orthogonality of the WHT matrix \mathbf{H} by mathematical induction. First, show that when $n = 1$ \mathbf{H}_1 is orthogonal, next show that if \mathbf{H}_{n-1} is orthogonal, then \mathbf{H}_n is also orthogonal.
2. Prove the orthogonality of the slant transform matrix \mathbf{S} by mathematical induction. Same as in the previous problem, First show that when $n = 1$ \mathbf{S}_1 is orthogonal. Then based on the assumption that \mathbf{S}_{n-1} is orthogonal, show \mathbf{S}_n is also orthogonal.
3. Show that computational complexity of the fast algorithm for DHT in Fig.8.10 is $O(N)$, linear to the size of the data. (Hint: follow the analysis of the FFT algorithm, and consider the number of stages in the algorithm and the complexity at each stage.)
4. Understand the C code for the WHT provided in the text and convert it into a Matlab function. Apply it to an $N=8$ sequence $\mathbf{x} = [x[0], \dots, x[7]]^T$ of your choice to obtain its N transform coefficients, then carry out the inverse transform to reconstruct the sequence from these transform coefficients.
5. Repeat the previous problem for the discrete slant transform ST.
6. Repeat the previous problem for the discrete Haar transform DHT.
7. Implement the 2-D sequency-ordered WHT of the same image used in the homework of Chapter 5 and carry out various types of filtering (low-pass, high-pass, etc.) of the image in transform domain. Then carry out the inverse transform and display the filtered image. Compare the filtering effects with those obtained by the Fourier transform obtained in Chapter 5.
8. Carry out image compression of the image used before as shown in Fig.5.21 by suppressing to zero all sequency components lower than a certain threshold. Obtain the percentage of such suppressed frequency components, and the percentage of lost energy (in terms of signal value squared). (Note that this exercise only serves to illustrate the basic idea of image compression but it

is not how image compression is practically done, where those components suppressed need to be recorded as well.)

9. Repeat the two problems above for the discrete slant transform ST.
10. Repeat the two problems above for the discrete Haar transform DHT.

9 Karhunen-Loeve Transform and Principal Component Analysis

9.1 Stochastic Process and Signal Correlation

9.1.1 Signals as Stochastic Processes

In all of our previous discussions, a time signal $x(t)$ is assumed to take a deterministic value $x(t_0)$ at any given moment $t = t_0$. However, in practice, many signals of interest are not deterministic, in the sense that multiple measurements of the same variable may be similar but not identical. While this random nature of such a signal could be caused by some inevitable measurement errors, we also realize that often a variable of certain physical process is affected by a large number of factors too complex to model in terms of how they collectively affect the variable of interest. Consequently the measured signal appears to be random.

The signal $x(t)$ of a non-deterministic variable can be considered as a *stochastic* or *random process*, of which a time sample $x(t_0)$ at $t = t_0$ is treated as a random variable with certain probability distribution. In this chapter, we will consider a special orthogonal transform that can be applied to such random signals, similar to the way all orthogonal transforms discussed previously are applied to deterministic signals.

Let us first review the following concepts of a stochastic process $x(t)$.

- The *mean function* of $x(t)$ is the expectation of the stochastic process:

$$\mu_x(t) = \int x(t)p(x_t)dx = E[x(t)] \quad (9.1)$$

where $p(x_t)$ is the *probability density function (pdf)* of the variable $x(t)$. If $\mu_x(t) = 0$ for all t , then $x(t)$ is a zero-mean or *centered* stochastic process. As any give process $x(t)$ can be turned into a zero-mean process by simply subtracting the mean from it $x(t) - \mu_x(t)$, we can always assume a given process $x(t)$ to be centered with a zero mean function without loss of generality.

- The *auto-covariance function* of $x(t)$ is defined as

$$\begin{aligned} Cov_x(t, \tau) &= \sigma_x^2(t, \tau) = \int \int (x(t) - \mu_x(t)) (\bar{x}(\tau) - \bar{\mu}_x(\tau)) p(x_t, x_\tau) dt d\tau \\ &= E[(x(t) - \mu_x(t)) (\bar{x}(\tau) - \bar{\mu}_x(\tau))] = E[x(t)\bar{x}(\tau)] - \mu_x(t)\bar{\mu}_x(\tau) \end{aligned}$$

where $p(x_t, x_\tau)$ is the *joint probability density function* of $x(t)$ and $x(\tau)$. When $t = \tau$, $\sigma^2(t, t) = \text{Var}_x(t) = E[|x(t)|^2] - \mu_x^2(t)$ becomes the variance of the signal at t . As we can always assume $x(t)$ to be centered with $\mu_x(t) = 0$, and the covariance $\sigma_x^2(t, \tau) = E[x(t)x(\tau)] = \langle x(t), x(\tau) \rangle$ can be considered as the inner product of the two variables $x(t)$ and $x(\tau)$ (Eq.2.20 in Chapter 2). In particular, if $\sigma_x^2(t, \tau) = \langle x(t), x(\tau) \rangle = 0$, the two variables are said to be orthogonal to each other.

- The *autocorrelation function* of $x(t)$ is defined as the covariance $\sigma_x^2(t, \tau)$ normalized by $\sigma_x(t)$ and $\sigma_x(\tau)$:

$$r_x(t, \tau) = \frac{\sigma_x^2(t, \tau)}{\sqrt{\sigma_x^2(t) \sigma_x^2(\tau)}} = \frac{\langle x(t), x(\tau) \rangle}{\sqrt{\langle x(t), x(t) \rangle \langle x(\tau), x(\tau) \rangle}} \quad (9.2)$$

Due to the Cauchy-Schwarz inequality (Eq.2.30) $|\langle x, y \rangle|^2 \leq \langle x, x \rangle \langle y, y \rangle$, we get $|r_x(t, \tau)| \leq 1$, and $r_x(t, \tau) = 1$ if and only if $t = \tau$. This result indicates that the similarity between any two different variables $x(t)$ and $x(\tau)$ is no greater than that of a variable $x(t)$ to itself, which is always 1 or one hundred percent.

If the joint probability density function of the random process $x(t)$ does not change over time, then $x(t)$ is a *stationary process*, and the following hold for any τ :

$$\mu_x(t) = \mu_x(t - \tau), \quad \sigma_x^2(t, \tau) = \sigma_x^2(t - \tau, 0), \quad r_x(t, \tau) = r_x(t - \tau, 0) \quad (9.3)$$

i.e., the mean function $\mu_x(t) = \mu_x$ becomes a constant, and auto-covariance and autocorrelation functions only depend on the time difference $t - \tau$ and therefore can be written as $\sigma_x^2(t - \tau)$ and $r_x(t - \tau)$, respectively. If these equations still hold even though the joint density function is not necessarily time invariant, then $x(t)$ is a *weak or wide-sense stationary (WSS)* process. Moreover, without loss of generality, we can further normalize the signal by a transformation $x'(t) = (x(t) - \mu_x)/\sigma_x^2$ so that its covariance becomes the same as its correlation. In other words, these two functions represent essentially the same characteristics of the signal.

Same as a deterministic signal, a random process $x(t)$ can also be truncated and sampled to become a finite set of N random variables $x[n] = x(nt_0)$ ($n = 0, \dots, N - 1$), where $t_0 = 1/F$ is the sampling period and $F = 1/t_0$ is the sampling rate. If the sampling rate is not a concern, we could assume $t_0 = F = 1$ for simplicity. The N signal samples can be represented by a random vector $\mathbf{x} = [x[0], \dots, x[N - 1]]^T$, correspondingly the mean and auto-covariance/autocorrelation functions for a random process become the mean vector and covariance matrix, respectively:

- The *mean vector* of a random vector \mathbf{x} is its expectation:

$$\boldsymbol{\mu}_x = E(\mathbf{x}) = [\mu[0], \dots, \mu[N - 1]]^T \quad (9.4)$$

where $\mu[n] = E(x[n])$ is the mean of $x[n]$ ($n = 0, \dots, N - 1$).

- The covariance matrix of a random vector \mathbf{x} is defined as:

$$\boldsymbol{\Sigma}_x = E[(\mathbf{x} - \boldsymbol{\mu}_x)(\mathbf{x} - \boldsymbol{\mu}_x)^*] = E[\mathbf{x}\mathbf{x}^*] - \boldsymbol{\mu}_x\boldsymbol{\mu}_x^* = \begin{bmatrix} \sigma_0^2 & \cdots & \sigma_{0(N-1)}^2 \\ \vdots & \ddots & \vdots \\ \sigma_{(N-1)0}^2 & \cdots & \sigma_{N-1}^2 \end{bmatrix} \quad (9.5)$$

where the element σ_{mn}^2 is the covariance of two $x[m]$ and $x[n]$ ($m, n = 0, \dots, N-1$):

$$\sigma_{mn}^2 = E[(x[m] - \mu[m])(\bar{x}[n] - \bar{\mu}[n])] = E(x[m]\bar{x}[n]) - \mu[m]\bar{\mu}[n] \quad (9.6)$$

As always, we can assume $\mu[n] = 0$ (by trivially subtracting the mean vector from the random vector) and get $\sigma_{mn}^2 = E(x[m]\bar{x}[n]) = \langle x[m], x[n] \rangle$. The nth component on the diagonal is the variance of the nth variable $x[n]$ representing the dynamic energy contained in $x[n]$:

$$\sigma_n^2 = E[|x[n] - \mu[n]|^2] = E(|x[n]|^2) - |\mu[n]|^2 \quad (9.7)$$

This covariance matrix $\boldsymbol{\Sigma}_x^* = \boldsymbol{\Sigma}_x$ is Hermitian and positive definite.

- The correlation matrix of a random vector \mathbf{x} is defined as:

$$\mathbf{R}_x = \begin{bmatrix} r_0 & \cdots & r_{0(N-1)} \\ \vdots & \ddots & \vdots \\ r_{(N-1)0} & \cdots & r_{N-1} \end{bmatrix} \quad (9.8)$$

where the element r_{mn} is the correlation coefficient between two random variables $x[m]$ and $x[n]$ defined as the covariance σ_{mn}^2 normalized by σ_m and σ_n :

$$r_{mn} = \frac{\sigma_{mn}^2}{\sqrt{\sigma_m^2 \sigma_n^2}} = \frac{\langle x[m], x[n] \rangle}{\sqrt{\langle x[n], x[n] \rangle \langle x[n], x[n] \rangle}}, \quad (m, n = 0, \dots, N-1) \quad (9.9)$$

where $\langle x[m], x[n] \rangle = E[x[m]\bar{x}[n]]$. r_{mn} measures the similarity between the two variables. Note that $r_{nn} = 1$ and $|r_{mn}| \leq 1$ for all $m \neq n$

The true mean vector $\boldsymbol{\mu}_x$ and covariance matrix $\boldsymbol{\Sigma}_x$ of a random vector \mathbf{x} are difficult to obtain as they depend on the joint probability density function $p(\mathbf{x})$, which is unlikely to be available in practice. However, $\boldsymbol{\mu}_x$ and $\boldsymbol{\Sigma}_x$ can be estimated if enough samples of the random vector can be obtained. Let $\{\mathbf{x}_k, (k = 1, \dots, K)\}$ be a set of K samples of the N -D random vector \mathbf{x} , then the mean vector and covariance matrix can be estimated as:

$$\hat{\boldsymbol{\mu}}_x = \frac{1}{K} \sum_{k=1}^K \mathbf{x}_k, \quad \text{and} \quad \hat{\boldsymbol{\Sigma}}_x = \frac{1}{K-1} \sum_{k=1}^K (\mathbf{x}_k - \hat{\boldsymbol{\mu}}_x)(\mathbf{x}_k - \hat{\boldsymbol{\mu}}_x)^* \quad (9.10)$$

where $\boldsymbol{\mu}_x = 0$ can always be assumed to be zero. Moreover, if we define a $K \times N$ matrix $\mathbf{D} = [\mathbf{x}_1, \dots, \mathbf{x}_K]^T$ composed of the K sample vectors of zero mean as

its row vectors, then the estimated covariance matrix can be expressed as:

$$\hat{\Sigma}_x = \frac{1}{K-1} [\mathbf{D}^T \mathbf{D}^*]_{N \times N} \quad (9.11)$$

9.1.2 Signal Correlation

Signal correlation is an important concept in signal processing in general, and in the context of the KLT transform in particular. As the measurement of a certain physical system, a signal tends to be smoothly and relatively evenly distributed in either time or space, in the sense that two samples of such a temporal or spatial signal are likely to be similar to each other if they are near to each other, but are less so if they are farther apart, i.e., they tend to be *locally correlated*. For example, given the current temperature as a signal sample $x(t)$, one could predict with reasonable confidence that the next sample $x(t + \tau)$ for the temperature in the near future with a small τ is fairly similar. However, one would be less confident when τ becomes larger. The correlation between two signal samples will eventually diminish when they are so far apart from each other that they are simply not relevant anymore. In other words, the smaller τ , the larger $r_x(t, t + \tau)$ and vice versa (e.g., the autocorrelation of the clouds in Fig. 7.10).

This common sense experience in everyday life is due to the general phenomenon that the energy associated with a system tends to be distributed smoothly and evenly over both time and space in the physical world governed by the principle of minimum energy and maximum *entropy*, which dictates that in a closed system, concentrated energy tends to disperse over time, and differences in physical quantities (temperature, pressure, density, etc.) tend to even out. Any disruption or discontinuity, typically associated with some kind of energy surge, is a relatively rare and unlikely event.

This signal characteristic of local correlation is reflected in the correlation matrix \mathbf{R}_x defined in Eq. 9.8. All elements along the diagonal take the maximum value 1 for self-correlation (always 100% correlated), while any off-diagonal element $|r_{mn}| < 1$ for the cross-correlation between two signal samples $x[m]$ and $x[n]$ always takes a smaller value. Moreover, those entries r_{mn} closer to the diagonal (small $|m - n|$) tend to take larger values (close to 1) than those farther away from the diagonal (large $|m - n|$). If the correlation matrix is thought of as a landscape, then there is a ridge along its diagonal along the NW-SE direction.

Based on this observation, a discrete signal can be modeled by a first order stationary *Markov chain* (see Appendix 2), of which the n th random sample $x[n]$ depends only on the previous sample $x[n - 1]$ with correlation $0 \leq r \leq 1$. The correlation between any two samples $x[m]$ and $x[n]$ is therefore $r_{mn} = r^{|m-n|}$, i.e., the correlation reduces exponentially as a function of the time interval between

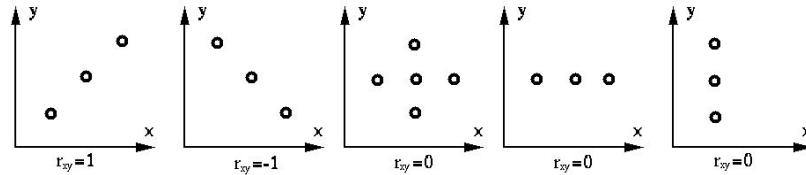


Figure 9.1 Different correlations between x and y

The two variables x and y are positively correlated (1st from left), negatively correlated (2nd), or not correlated (3rd to 5th).

them, and the correlation matrix can be written as:

$$\mathbf{R}_x = \begin{bmatrix} 1 & r & r^2 & \dots & r^{N-2} & r^{N-1} \\ r & 1 & r & \dots & r^{N-3} & r^{N-2} \\ r^2 & r & 1 & \dots & r^{N-4} & r^{N-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ r^{N-2} & r^{N-3} & r^{N-4} & \dots & 1 & r \\ r^{N-1} & r^{N-2} & r^{N-3} & \dots & r & 1 \end{bmatrix}_{N \times N} \quad (9.12)$$

This is a Toeplitz matrix with all elements along the diagonal direction being the same. This model of first-order Markov chain will be used later.

A set of simple examples to illustrate intuitively the different amount of correlation between two random variables x and y will be considered in the homework. Here we just show a few different cases in Fig.9.1, where each dot represents an outcome of a experiment in terms of $N = 2$ values of x and y , with N easily generalized to any value $N > 2$. In each of the five cases shown in the figure, the variances σ_x^2 and σ_y^2 represent respectively the dynamic energy or information contained in the two variables x and y , while the correlation $r_{xy} = \sigma_{xy}^2 / \sqrt{\sigma_x^2 \sigma_y^2}$ represents how much the two variables are correlated. Specifically if $r_{xy} > 0$, they are positively correlated as in the first case in Fig.9.1, where the two variables contain the same amount of energy $\sigma_x^2 = \sigma_y^2$. But as they are maximally correlated with $r_{xy} = 1$, they carry redundant information in the sense that given x we know y . On the other hand, if $r_{xy} < 0$ they are negatively correlated. As in the second case in the figure, $r_{xy} = -1$ and the two variables contain the same amount of energy and they carry redundant information as they are completely (and negatively) correlated. If $r_{xy} = 0$ as in the third case in the figure, the two variables are uncorrelated, each carrying its own independent information. In the last two cases $\sigma_{xy}^2 = 0$ while either $\sigma_y^2 = 0$ or $\sigma_x^2 = 0$, i.e., one of the two variables contains zero dynamic energy and can therefore be omitted. In this case the dimension of the data set can be reduced from 2 to 1 without losing any information. Moreover, we note that by a 45-degree rotation of the coordinate system (an orthogonal transform that conserves signal energy), the first two cases will be converted into the last two, where the two variables are no longer correlated and the total energy is redistributed. One of the two variables contains 100% of

the energy, while the other containing zero energy can be omitted without losing any information. This simple example of rotation illustrates the essential reason why orthogonal transforms can be used for data compression.

In general, from the view point of data compression and signal processing, we want to avoid high signal correlation and even energy distribution, it is therefore desirable to convert the given data set in such a way so that (1) the signal components are minimally correlated with least amount of redundancy, and (2) the total energy contained in the components is mostly contained in a small number of them so that those that carry little energy can be omitted. These properties are commonly desired for many data processing applications such as information extraction, noise reduction and data compression. We will next consider such a transform method that can achieve these goals optimally.

9.2 Karhunen-Loeve Transform (KLT)

9.2.1 Continuous Karhunen-Loeve Theorem

As seen before, a deterministic time signal $x(t)$ can be represented by an orthogonal transform as a linear combination of a set of orthogonal basis functions (Eq.2.107)

$$x(t) = \sum_k c[k] \phi_k(t) \quad (9.13)$$

where the coefficients $c[k]$ can be found as the projection of $x(t)$ onto each of the basis functions (Eq.2.109):

$$c[k] = \langle x(t), \phi_k(t) \rangle = \int x(t) \bar{\phi}_k(t) dt \quad (9.14)$$

On the other hand, a random signal $x(t)$ as a stochastic process can also be represented in exactly the same form (Eqs.2.377 and 2.378) according to the Karhunen-Loeve theorem (Thm. 2.15). As discussed in section 2.5, the covariance $\sigma_x^2(t, \tau)$ of a centered stochastic process $x(t)$ is a Hermitian kernel, and the associated integral operator is a self-adjoint and positive definite with real positive eigenvalues λ_k and orthogonal eigenfunctions $\phi_k(t)$ (Thm.2.358). These eigenfunctions form an orthogonal basis of the function space in which a random signal $x(t)$ resides, i.e., Eqs.9.13 and 9.14 can also be considered as the series expansion of a stochastic signal $x(t)$, by which the signal is converted into a set of coefficients $c[k]$ for the series in the transform domain.

However, we note that the two interpretations of Eqs.9.13 and 9.14 are of essential difference. When $x(t)$ is a deterministic signal, the coefficients $c[k]$ are constants; but when $x(t)$ is a random signal, the coefficients $c[k]$ are random variables. The random nature of $x(t)$, when series expanded to become $x(t) = \sum_k c[k] \phi_k(t)$, is reflected by the random coefficients $c[k]$. In either case, the orthogonal basis functions $\phi_k(t)$ of the expansion are always deterministic.

9.2.2 Discrete Karhunen-Loeve Transform

We now consider the discrete version of the Karhunen-Loeve theorem. When a stochastic process $x(t)$ is truncated and sampled, it becomes a random vector composed of N random variables $\mathbf{x} = [x[0], \dots, x[N-1]]^T$. For convenience and without loss of generality, we will always assume in the following that the signal is centered with $\mu_x = \mathbf{0}$, and its covariance matrix is $\Sigma_x = E(\mathbf{x}\mathbf{x}^*)$ with its m -th element being $\sigma_{mn}^2 = E(x[m]\bar{x}[n]) = \langle x[m], x[n] \rangle$. As Σ_x is positive definite and Hermitian, all of its eigenvalues λ_k are real and positive, and its eigenvectors ϕ_k ($k = 0, \dots, N-1$) form a set of orthogonal basis vectors that span the N-D vector space. Any given N-D random vector in the space can be represented as a linear combination of these basis vectors. This is the discrete Karhunen-Loeve theorem.

Let ϕ_k ($k = 0, \dots, N-1$) be the eigenvector corresponding to the k th eigenvalue λ_k of the covariance matrix Σ_x , i.e.,

$$\Sigma_x \phi_k = \lambda_k \phi_k \quad (k = 0, \dots, N-1) \quad (9.15)$$

As Σ_x is Hermitian and positive definite, all its eigenvalues $\lambda_k > 0$ are real and positive. Moreover, its N eigenvectors are orthogonal, $\langle \phi_k, \phi_l \rangle = \delta[k-l]$ ($k, l = 0, \dots, N-1$), and they form an $N \times N$ unitary matrix $\Phi = [\phi_0, \dots, \phi_{N-1}]$ satisfying $\Phi^{-1} = \Phi^*$, i.e., $\Phi^* \Phi = \Phi \Phi^* = \mathbf{I}$. The N eigenequations in Eq.9.15 can then be combined to become:

$$\Sigma_x \Phi = \Phi \Lambda \quad (9.16)$$

where $\Lambda = \text{diag}(\lambda_0, \dots, \lambda_{N-1})$ is a diagonal matrix. By pre-multiplying $\Phi^* = \Phi^{-1}$ on both sides, the covariance matrix Σ_x is diagonalized:

$$\Phi^* \Sigma_x \Phi = \Phi^* \Phi \Lambda = \Lambda \quad (9.17)$$

The discrete Karhunen-Loeve transform of a given random signal vector \mathbf{x} can now be defined as:

$$\mathbf{X} = \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \Phi^* \mathbf{x} = \begin{bmatrix} \phi_0^* \\ \vdots \\ \phi_{N-1}^* \end{bmatrix} \mathbf{x} \quad (9.18)$$

where the k th component $X[k]$ of the vector \mathbf{X} in transform domain is the projection of \mathbf{x} onto the k th basis vector ϕ_k :

$$X[k] = \phi_k^* \mathbf{x} = \langle \mathbf{x}, \phi_k \rangle \quad (9.19)$$

Pre-multiplying Φ on both sides of equation 9.18, we get the inverse KLT transform:

$$\mathbf{x} = \Phi \mathbf{X} = [\phi_0, \dots, \phi_{N-1}] \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \sum_{k=0}^{N-1} X[k] \phi_k \quad (9.20)$$

Eqs. 9.20 and 9.18 can be rewritten as a pair of the discrete KLT transform:

$$\begin{cases} \mathbf{X} = \Phi^* \mathbf{x} \\ \mathbf{x} = \Phi \mathbf{X} \end{cases} \quad (9.21)$$

The first equation is the forward transform that gives the random coefficient $X[k]$ as the projection of the random vector \mathbf{x} onto the k th deterministic basis vector ϕ_k ($k = 0, \dots, N - 1$), while the second equation is the inverse transform that represents the random vector \mathbf{x} as a linear combination of the N eigenvectors ϕ_k ($k = 0, \dots, N - 1$) of Σ_x weighted by the random coefficients $X[k]$. Note that Eqs. 9.20 and 9.18 for the discrete KLT correspond to Eqs. 9.13 and 9.14 for the continuous KLT.

9.2.3 Optimalities of the KLT

As discussed in previous chapters, all orthogonal transforms exhibit to various extents the properties of signal decorrelation and energy compaction. For example, in frequency domain after the Fourier transform, most of the signal energy is likely to be concentrated in a small number of low frequency components while little energy is contained in high frequency components. Moreover, while the signal is typically locally correlated in time domain, in the sense that the signal value $x[n]$ can be predicted to be similar to the previous one $x[n - 1]$, this is no longer the case in frequency domain, as knowing the value of a frequency component $X[k]$ would provide little information regarding the neighboring components. Other orthogonal transforms have the similar effects.

Now we show that among all orthogonal transforms, the KLT is optimal in terms of signal decorrelation and energy compaction, as stated in the theorem below:

Theorem 9.1. *Let $\mathbf{X} = \Phi^* \mathbf{x}$ be the KLT of an N -D random signal vector \mathbf{x} , where Φ is the eigenvector matrix associated with the covariance matrix Σ_x of \mathbf{x} (Eq. 9.17). Then by the KLT,*

1. *The total signal energy \mathcal{E}_x is conserved:*

$$\mathcal{E}_X = \sum_{k=0}^{N-1} E(|X[k]|^2) = \text{tr} \Sigma_X = \text{tr} \Sigma_x = \sum_{k=0}^{N-1} E(|x[k]|^2) = \mathcal{E}_x \quad (9.22)$$

2. *The signal is completely decorrelated, i.e., all off-diagonal components of Σ_X are zero:*

$$\sigma_{kl}^2 = 0, \quad \text{for all } k \neq l \quad (9.23)$$

3. *The signal energy is maximally compacted:*

$$\mathcal{E}_M(\Phi) \geq \mathcal{E}_M(\mathbf{A}) \quad (9.24)$$

where $\mathcal{E}_M(\mathbf{A})$ is the energy contained in the first $M < N$ signal components after an arbitrary orthogonal transform $\mathbf{X} = \mathbf{A}^* \mathbf{x}$

Proof: The first statement is true simply because the trace of the covariance matrix remains the same after any unitary transform:

$$\text{tr} \Sigma_X = \text{tr}(\Phi^* \Sigma_x \Phi) = \text{tr}(\Phi^* \Phi \Sigma_x) = \text{tr} \Sigma_x \quad (9.25)$$

where we have used Eq.12.19. This result is equivalent to Parseval's identity for the property of energy conservation of any orthogonal transform of the deterministic signals.

The second statement is true due to the definition of the KLT by which the covariance matrix Σ_X of $\mathbf{X} = \Phi^* \mathbf{x}$ is diagonalized (Eq.9.17):

$$\Sigma_X = E(\mathbf{X} \mathbf{X}^*) = E[(\Phi^* \mathbf{x})(\Phi^* \mathbf{x})^*] = \Phi^* E(\mathbf{x} \mathbf{x}^*) \Phi = \Phi^* \Sigma_x \Phi = \Lambda \quad (9.26)$$

As all off-diagonal elements $\sigma_{kl}^2 = E(X[k]\bar{X}[l]) = \langle X[k], X[l] \rangle = 0$ ($k \neq l$), any two different components $X[k]$ and $X[l]$ are indeed decorrelated. The total signal energy is the sum of all eigenvalues (real and positive):

$$\mathcal{E}_X = \text{tr} \Sigma_X = \text{tr} \Lambda = \sum_{k=0}^{N-1} \lambda_k \quad (9.27)$$

To prove the third statement, we let $\mathbf{A} = [\mathbf{a}_0, \dots, \mathbf{a}_{N-1}]$ be an arbitrary unitary matrix ($\mathbf{A}^* = \mathbf{A}^{-1}$), then the k th element of $\mathbf{X} = \mathbf{A}^* \mathbf{x}$ is $X[k] = \mathbf{a}_k^* \mathbf{x}$, and the energy contained in the first $M < N$ components after this transform is the sum of the first M elements along the diagonal of Σ_X :

$$\begin{aligned} \mathcal{E}_M(\mathbf{A}) &= \sum_{k=0}^{M-1} E(|X[k]|^2) = \sum_{k=0}^{M-1} E(|\mathbf{a}_k^* \mathbf{x}|^2) = \sum_{k=0}^{M-1} E[(\mathbf{a}_k^* \mathbf{x}) (\mathbf{x}^* \mathbf{a}_k)^*] \\ &= \sum_{k=0}^{M-1} E(\mathbf{a}_k^* \mathbf{x} \mathbf{x}^* \mathbf{a}_k) = \sum_{k=0}^{M-1} \mathbf{a}_k^* E(\mathbf{x} \mathbf{x}^*) \mathbf{a}_k = \sum_{k=0}^{M-1} \mathbf{a}_k^* \Sigma_x \mathbf{a}_k \end{aligned}$$

The task of finding the optimal matrix \mathbf{A} that maximizes $\mathcal{E}_M(\mathbf{A})$ can be formulated as a constrained optimization problem:

$$\begin{aligned} \mathcal{E}_M(\mathbf{A}) &= \sum_{k=0}^{M-1} \mathbf{a}_k^* \Sigma_x \mathbf{a}_k \rightarrow \max \\ \text{subject to: } \mathbf{a}_k^* \mathbf{a}_k &= 1 \quad (k = 0, \dots, M-1) \end{aligned} \quad (9.28)$$

Here the constraint $\mathbf{a}_k^* \mathbf{a}_k = 1$ is to guarantee that \mathbf{A} is indeed an orthogonal matrix with orthonormal column vectors. This problem can be solved by the method of Lagrange multipliers. Specifically, we set to zero the following partial

derivative of the modified objective function with respect to \mathbf{a}_l :

$$\begin{aligned} \frac{\partial}{\partial \mathbf{a}_l} [\mathcal{E}_M(\mathbf{A}) - \sum_{k=0}^{M-1} \lambda_k (\mathbf{a}_k^* \mathbf{a}_k - 1)] &= \frac{\partial}{\partial \mathbf{a}_l} \left[\sum_{k=0}^{M-1} (\mathbf{a}_k^* \boldsymbol{\Sigma}_x \mathbf{a}_k - \lambda_k \mathbf{a}_k^* \mathbf{a}_k + \lambda_k) \right] \\ &= \frac{\partial}{\partial \mathbf{a}_l} [\mathbf{a}_l^* \boldsymbol{\Sigma}_x \mathbf{a}_l - \lambda_l \mathbf{a}_l^* \mathbf{a}_l] = 2\boldsymbol{\Sigma}_x \mathbf{a}_l - 2\lambda_l \mathbf{a}_l = 0 \end{aligned} \quad (9.29)$$

The last equal sign is due to the derivative of a scalar function $f(\mathbf{a})$ with respect to its vector argument \mathbf{a} (Eq.12.67). This equation happens to be the eigenequation of matrix $\boldsymbol{\Sigma}_x$:

$$\boldsymbol{\Sigma}_x \mathbf{a}_l = \lambda_l \mathbf{a}_l, \quad (l = 0, \dots, M-1) \quad (9.30)$$

Comparing this to Eq.9.15, we see that $\mathbf{a}_l = \boldsymbol{\phi}_l$ must be the eigenvectors of $\boldsymbol{\Sigma}_x$, i.e., the optimal transform matrix must be the KLT matrix $\mathbf{A} = \boldsymbol{\Phi} = [\boldsymbol{\phi}_0, \dots, \boldsymbol{\phi}_{N-1}]$. The energy contained in the first M components is:

$$\mathcal{E}_M(\boldsymbol{\Phi}) = \sum_{k=0}^{M-1} \boldsymbol{\phi}_k^* \boldsymbol{\Sigma}_x \boldsymbol{\phi}_k = \sum_{k=0}^{M-1} \lambda_k \quad (9.31)$$

where the k th eigenvalue $\lambda_k = E(|X[k]|^2)$ is the average energy contained in the k th component $X[k]$ of $\mathbf{X} = \boldsymbol{\Phi}^* \mathbf{x}$. This energy $\mathcal{E}_M(\boldsymbol{\Phi})$ is maximized if we choose to keep the M signal components corresponding to the M largest eigenvalues. The percentage of energy kept in the M components is

$$\frac{\mathcal{E}_M(\boldsymbol{\Phi})}{\mathcal{E}_N} = \frac{\sum_{k=0}^{M-1} \lambda_k}{\sum_{k=0}^{N-1} \lambda_k} \quad (9.32)$$

Q.E.D.

The optimality of energy compaction of the KLT can also be viewed in terms of *Shannon entropy* or simply *entropy*. To understand the concept of entropy, let us first consider a random variable x representing the outcome of a random event. We assume there are in total N possible outcomes each with probability p_k ($k = 0, \dots, N-1$) and $\sum_{k=0}^{N-1} p_k = 1$. The uncertainty of a specific outcome x_k can be defined by

$$I(x_k) = \log(1/p_k) = -\log p_k, \quad (k = 0, \dots, N-1) \quad (9.33)$$

In particular, when $p_k = 1$, $I(x_k) = 0$, i.e., a necessary event has zero uncertainty. On the other hand, when $p_k = 0$, $I(x_k) = \infty$, i.e., an impossible event has infinite uncertainty. The entropy of the random event x is its uncertainty defined as the expected uncertainty $I(x_k)$ of any output x_k :

$$H(x) = E[I(x_k)] = -E[\log p_k] = -\sum_{k=0}^{N-1} p_k \log p_k \quad (9.34)$$

It is unessential what logarithmic base is used, as the entropies corresponding different bases are the same up to a scaling factor. The unit of entropy H is *bit* if the base is 2, or *nat* or *nit* if the natural logarithm is used with base e . The

two units are related by a scaling factor of $\ln 2$. In the special case of N equally likely outcomes with $p_k = 1/N$ for all N possible outcomes, the entropy reaches its maximum $H = \log N$ for maximum uncertainty. On the other hand, when $p_l = 1$ but $p_k = 0$ for all $k \neq l$, the entropy reaches its minimum $H = 0$ for zero uncertainty.

After certain amount of information regarding the outcome of a random event x is gained, its certainty may be reduced from $H(x)$ to $H'(x)$, and the reduction in uncertainty $I(x) = H(x) - H'(x)$ is a quantitative measurement of the amount of information gained. In particular, if the outcome of the event is completely known, the uncertainty is reduced from $H(x)$ to $H'(x) = 0$, i.e., the entropy $H(x)$ also represents the total amount of information contained in the random variable x .

The energy distribution among all N components $x[n]$ of a random signal \mathbf{x} can be considered as a histogram or a probability distribution of N possible outcomes of a random event, based on which the entropy defined in Eq.9.34 can be used to measure quantitatively how well the signal energy is concentrated among its N components. Typically the energy of a signal is relatively evenly distributed among all signal components, i.e., the uncertainty is large. But after certain orthogonal transform (e.g., the DFT or DCT), the energy is redistributed so that most of it is compacted into a small number of components (e.g., the low frequency components), i.e., the uncertainty is reduced. As the KLT is optimal in the sense that it maximally compacts signal energy into a small number of signal components, it minimizes the entropy defined this way.

From the data compression point of view, signal uncertainty measured by entropy is also indicative of how much the data can be reduced by certain compression method such as an entropy encoding algorithm called *Huffman coding*, which is an optimal compression algorithm that assigns variable code lengths to a set of N signal symbols according to their probabilities p_k . The optimality is achieved by always assigning shorter code to more probable symbols so that the average code length is minimized. As can be seen in one of the homework problems, the average code length is positively related to the signal entropy. Therefore it is always desirable for the purpose of data compression to carry out certain orthogonal transform by which the signal energy is compacted and its entropy reduced, so that shorter average code will result and better compression effect can be achieved.

9.2.4 Geometric Interpretation of the KLT

The property of optimal energy compaction of the KLT can also be viewed in terms of information contained in the signal. We assume a random signal vector composed of a set of N real random variables $\mathbf{x} = [x[0], \dots, x[N-1]]^T$ has a

normal joint probability density function (Eq.13.36):

$$p(\mathbf{x}) = N(\mathbf{x}, \boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x) = \frac{1}{(2\pi)^{N/2} |\boldsymbol{\Sigma}_x|^{1/2}} \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_x)^T \boldsymbol{\Sigma}_x^{-1} (\mathbf{x} - \boldsymbol{\mu}_x)\right] \quad (9.35)$$

As always, we assume $\boldsymbol{\mu}_x = \mathbf{0}$ without loss of generality. Based on this probability density function $p(\mathbf{x})$, the uncertainty, or the amount of information contained in this signal, can be measured in terms of the entropy defined in Eq.9.34:

$$H(\mathbf{x}) = -E[\ln p(\mathbf{x})] = \frac{N}{2} \ln 2\pi + \frac{1}{2} \ln |\boldsymbol{\Sigma}_x| + \frac{1}{2} E[\mathbf{x}^T \boldsymbol{\Sigma}_x^{-1} \mathbf{x}] \quad (9.36)$$

Note that this entropy based on $p(\mathbf{x})$ is different from that based on the energy distribution histogram considered previously. According to Eq.12.43, the second term can be further written as:

$$\frac{1}{2} \ln |\boldsymbol{\Sigma}_x| = \frac{1}{2} \ln \left(\prod_{k=0}^{N-1} \lambda_k \right) = \frac{1}{2} \sum_{k=0}^{N-1} \ln \lambda_k \quad (9.37)$$

and according to Eq.12.19, the last term (a scalar) can be further written as:

$$\begin{aligned} \frac{1}{2} E[\text{tr}(\mathbf{x}^T \boldsymbol{\Sigma}_x^{-1} \mathbf{x})] &= \frac{1}{2} E[\text{tr}(\boldsymbol{\Sigma}_x^{-1} \mathbf{x} \mathbf{x}^T)] \\ &= \frac{1}{2} \text{tr}(\boldsymbol{\Sigma}_x^{-1} E[\mathbf{x} \mathbf{x}^T]) = \frac{1}{2} \text{tr}(\boldsymbol{\Sigma}_x^{-1} \boldsymbol{\Sigma}_x) = \frac{1}{2} \text{tr} \mathbf{I} = \frac{N}{2} \end{aligned} \quad (9.38)$$

Substituting these two terms back into the equation above we get

$$H(\mathbf{x}) = \frac{N}{2} (\ln 2\pi + 1) + \frac{1}{2} \sum_{k=0}^{N-1} \ln \lambda_k \quad (9.39)$$

If, for the purpose of data compression, we want to keep only $M < N$ of the N variables with minimum information loss, we would need to keep the variables corresponding to the M greatest eigenvalues λ_k to maximize the entropy $H(\mathbf{x})$.

The shape of the normal distribution in the N-D space given in Eq.9.35 can be represented by an iso-value hyper-surface in the space determined by:

$$N(\mathbf{x}, \boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x) = c \quad (9.40)$$

where the constant can be so chosen so that:

$$(\mathbf{x} - \boldsymbol{\mu}_x)^T \boldsymbol{\Sigma}_x^{-1} (\mathbf{x} - \boldsymbol{\mu}_x) = \mathbf{x}^T \boldsymbol{\Sigma}_x^{-1} \mathbf{x} = 1 \quad (9.41)$$

As $\boldsymbol{\Sigma}_x$ is positive definite, this quadratic equation represents a hyper-ellipsoid in the N-D space, whose spatial orientation is totally determined by $\boldsymbol{\Sigma}_x$.

After the KLT the signal vector \mathbf{x} becomes $\mathbf{X} = \Phi^T \mathbf{x}$ which is completely decorrelated with a diagonalized covariance matrix (Eq.9.26):

$$\boldsymbol{\Sigma}_X = \boldsymbol{\Lambda} = \begin{bmatrix} \lambda_0 & 0 & \cdots & 0 \\ 0 & \lambda_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_{N-1} \end{bmatrix} = \begin{bmatrix} \sigma_X^2[0] & 0 & \cdots & 0 \\ 0 & \sigma_X^2[1] & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_X^2[N-1] \end{bmatrix} \quad (9.42)$$

Substituting $\mathbf{x} = \Phi \mathbf{X}$ into the quadratic equation Eq.9.41, we get:

$$\begin{aligned}\mathbf{x}^T \Sigma_x^{-1} \mathbf{x} &= \mathbf{X}^T \Phi^T \Sigma_x^{-1} \Phi \mathbf{X} = \mathbf{X}^T \Sigma_X^{-1} \mathbf{X} \\ &= \mathbf{X}^T \Lambda^{-1} \mathbf{X} = \sum_{k=0}^{N-1} \frac{X^2[k]}{\lambda_k} = \sum_{k=0}^{N-1} \frac{X^2[k]}{\sigma_X^2[k]} = 1\end{aligned}\quad (9.43)$$

This is the equation of a standard hyper-ellipsoid with its N semi-axes being $\sqrt{\lambda_k} = \sigma_X[k]$. We see that the KLT can be interpreted geometrically in terms of the following effects:

- The coordinate system of the N-D space is rotated in such a way that they are now aligned with the eigenvectors ϕ_k ($k = 0, \dots, N - 1$) of Σ_x .
- The semi-principal axes of the hyper-ellipsoid representing the distribution $N(\mathbf{x}, \mu_x, \Sigma_x)$ are in parallel with the new coordinates ϕ_k .
- The lengths of these semi-principal axes are the square root of the corresponding eigenvalue $\sqrt{\lambda_k}$ ($k = 0, \dots, N - 1$).

These properties can be most conveniently visualized when $N = 2$, as illustrated in Fig.9.2. Here a signal $\mathbf{x} = [x_0, x_1]^T$ is originally represented under the standard basis vectors e_0 and e_1 :

$$\mathbf{x} = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = x_0 e_0 + x_1 e_1 = x_0 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + x_1 \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (9.44)$$

The quadratic equation in Eq.9.41 representing the 2-D normal distribution of the signal \mathbf{x} can be written as:

$$\mathbf{x}^T \Sigma_x^{-1} \mathbf{x} = [x_0, x_1] \begin{bmatrix} a & b/2 \\ b/2 & c \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = ax_0^2 + bx_0x_1 + cx_1^2 = 1 \quad (9.45)$$

where we have assumed:

$$\Sigma_x^{-1} = \begin{bmatrix} a & b/2 \\ b/2 & c \end{bmatrix} \quad (9.46)$$

As Σ_x is positive definite and so is Σ_x^{-1} , we have $|\Sigma_x^{-1}| = ac - b^2/4 > 0$, i.e., the quadratic equation above represents an ellipse (instead of any other quadratic curves such as a hyperbola or parabola) centered at the origin (or at μ_x if it is not zero). As shown in Fig.9.2, the two signal components x_0 and x_1 are maximally correlated with $r_{01} = 1$ and contain equal amount of energy $\sigma_{x_0}^2 = \sigma_{x_1}^2$, i.e., the energy is evenly distributed among both components.

Then a 2-D KLT $\mathbf{y} = \Phi^T \mathbf{x}$ is carried out in three stages: (1) subtract the mean μ_x from \mathbf{x} so that it is centered, (2) carry out the rotation $\mathbf{y} = \Phi^T \mathbf{x}$, and (3) add back the mean vector in the rotated space $\mu_y = \Phi^T \mu_x$. After the KLT, the signal is represented by two new basis vectors ϕ_0 and ϕ_1 , which are just rotated version of e_0 and e_1 . In this space spanned by ϕ_0 and ϕ_1 , the ellipse representing the joint probability density $p(\mathbf{x})$ becomes standardized with major semi-axis $\sqrt{\lambda_0} = \sigma_{X_0}$ and minor semi-axes $\sqrt{\lambda_1} = \sigma_{X_1}$, in parallel with the new basis vectors ϕ_0 and ϕ_1 , respectively.

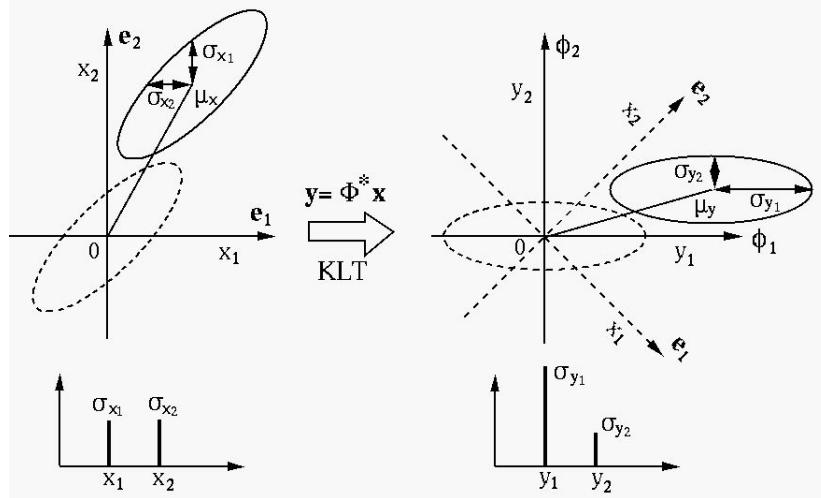


Figure 9.2 Geometric interpretation of KLT $y = \Phi^T x$

We see that after the KLT the two components y_0 and y_1 are completely decorrelated with $r_{01} = 0$, and $\lambda_0 > \lambda_1$ indicating that the energy is maximally compacted into y_0 while y_1 contains minimal energy. We also note that this KLT rotation is optimal in terms of both signal decorrelation and energy compaction, as no other rotation can do any better in these regards.

9.2.5 Principal Component Analysis (PCA)

Due to its optimality of signal decorrelation and energy compaction, the KLT can be used to reduce the dimensionality of a given data set while preserving maximum signal energy/information in various applications such as information extraction and data compression. The signal components $X[k]$ after the KLT are called the *principal components*, and the data analysis method based on the KLT transform is called *principal component analysis (PCA)*, which is widely used in a large variety of fields. Specifically the PCA can be carried out in the following steps:

1. Estimate the mean vector μ_x of the given random signal vector x . Subtract μ_x from x so that it becomes centered with zero mean.
2. Estimate the covariance matrix Σ_x of the centered signal.
3. Find all N eigenvalues and sort them in descending order:

$$\lambda_0 \geq \dots \geq \lambda_{N-1} \quad (9.47)$$

4. Determine a reduced dimensionality $M < N$ so that the percentage of energy contained $\sum_{n=0}^{M-1} \lambda_n / \sum_{n=0}^{N-1} \lambda_n$ is no less than a preset threshold (e.g., 99%).

5. Construct an $N \times M$ transform matrix composed of the M eigenvectors corresponding to the M largest eigenvalues of Σ_x :

$$\Phi_M = [\phi_0, \dots, \phi_{M-1}]_{N \times M} \quad (9.48)$$

and carry out the KLT based on this Φ_M :

$$\mathbf{X}_M = \begin{bmatrix} X[0] \\ \vdots \\ X[M-1] \end{bmatrix}_{M \times 1} = \Phi_M^* \mathbf{x} = \begin{bmatrix} \phi_0^* \\ \vdots \\ \phi_{M-1}^* \end{bmatrix}_{M \times N} \begin{bmatrix} x[0] \\ \vdots \\ x[N-1] \end{bmatrix}_{N \times 1} \quad (9.49)$$

where the k th element is $X[k] = \phi_k^* \mathbf{x} = \langle \mathbf{x}, \phi_k \rangle$. As the dimensionality M of \mathbf{X} is less than the dimensionality N of \mathbf{x} , data compression is achieved. This is a lossy compression with the error representing the percentage of information lost: $\sum_{k=M}^{N-1} \lambda_k / \sum_{k=0}^{N-1} \lambda_k$. But as these λ_k 's in the numerator summation are the smallest eigenvalues, the error is minimum (e.g., 1%).

6. Carry out analysis needed in the M -dimensional space, and inverse KLT for reconstruction if needed (e.g., for compression):

$$\hat{\mathbf{x}} = \Phi_M \mathbf{X}_M = \Phi_M \Phi_M^* \mathbf{x} \quad (9.50)$$

or in component form:

$$\hat{\mathbf{x}} = \begin{bmatrix} \hat{x}[0] \\ \vdots \\ \hat{x}[N-1] \end{bmatrix} = [\phi_0 \cdots \phi_{M-1}] \begin{bmatrix} X[0] \\ \vdots \\ X[M-1] \end{bmatrix} = \sum_{k=0}^{M-1} X[k] \phi_k \quad (9.51)$$

$$= [\phi_0 \cdots \phi_{M-1}] \begin{bmatrix} \phi_0^* \\ \vdots \\ \phi_{M-1}^* \end{bmatrix} \mathbf{x} = \left[\sum_{k=0}^{M-1} \phi_k \phi_k^* \right]_{N \times N} \mathbf{x} \quad (9.52)$$

Here Eq.9.51 indicates that $\hat{\mathbf{x}}$ is a linear combination of the first M of the N eigenvectors that span the N-D space, while Eq.9.52 indicates that $\hat{\mathbf{x}}$ is a linear transformation of \mathbf{x} by an $N \times N$ matrix formed as the sum of the M outer products $\phi_k \phi_k^*$ ($k = 0, \dots, M-1$). In particular when $M = N$, this matrix becomes $\Phi_N \Phi_N^* = \mathbf{I}_{N \times N}$ and $\hat{\mathbf{x}} = \mathbf{x}$ is a perfect reconstruction.

9.2.6 Comparison with Other Orthogonal Transforms

To illustrate the optimality of the KLT in terms of the two desirable properties of signal decorrelation and energy compaction discussed above, we compare its performance with a set of real orthogonal transforms considered in previous chapters including identity transform (IT or no transform), discrete cosine transform (DCT), Walsh-Hadamard transform (WHT), slant transform (SLT), and discrete Haar transform (DHT) in the following examples based on two $M \times N$ images shown in Fig.9.3, an image of clouds (left) and another image of sand (right).

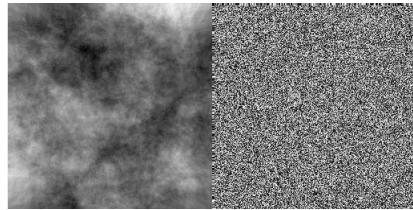


Figure 9.3 Images of clouds and sand

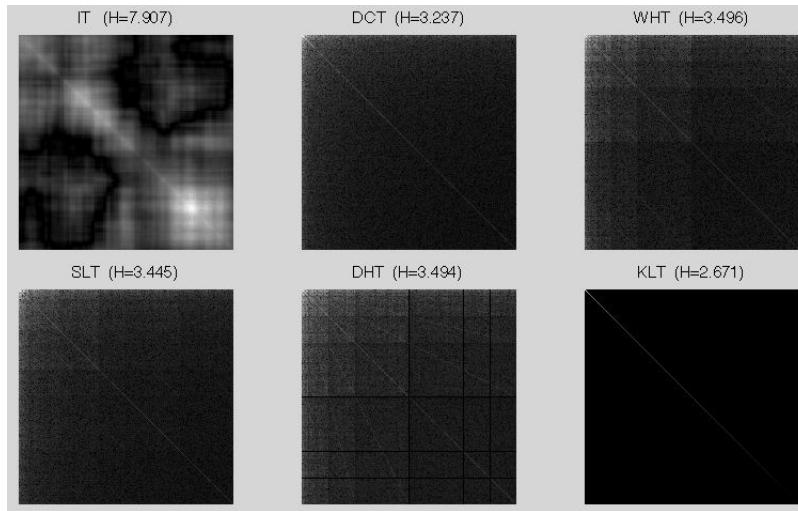


Figure 9.4 Covariance matrices of cloud image after various transforms

We carry out each of these orthogonal transforms represented as a generic transform $\mathbf{X} = \mathbf{A}^T \mathbf{x}$, where $\mathbf{A}^T = \mathbf{A}^{-1}$ is an orthogonal matrix operator applied to \mathbf{x} for each of the M rows of the image treated as M instantiations of a random vector \mathbf{x} of N components. We then compare the covariance matrix Σ_x of the original vector \mathbf{x} with the covariance matrix $\Sigma_X = \mathbf{A}^T \Sigma_x \mathbf{A}$ (Eq.13.34) after the transform to see how well each transform method decorrelates the signal and compacts its energy.

The covariance matrices of the cloud and sand images after each of the transforms are shown in image form in Figs.9.4 and 9.5, respectively. The intensities of the image pixels representing the $N \times N$ covariance matrix elements are rescaled by a non-linear mapping $y = x^{0.3}$ for the very low values to be visible as well as the high values.

In the top-left panel of Fig.9.4 showing the covariance matrix of the original signal before any transform (or IT), there exist quite a lot bright areas off the diagonal, indicating that a significant number of signal components are highly correlated. We can also observe a general trend that the elements around the diagonal are brighter than those farther away from the diagonal, indicating the

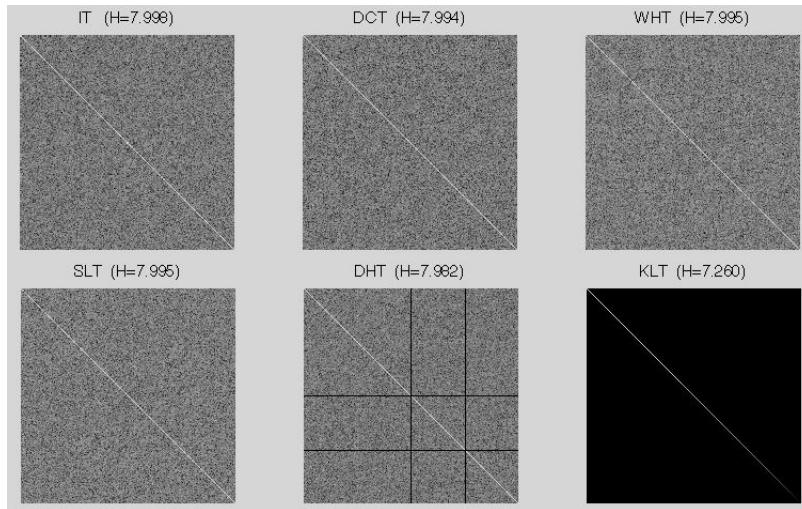


Figure 9.5 Covariance matrices of sand image after various transforms

fact that neighboring signal components tend to be more correlated than those that are farther apart. In the next few panels of the figure showing the covariance matrix after certain orthogonal transform, the values of the off-diagonal elements are much reduced, indicating that the signal components are significantly decorrelated. Finally, in the lower-right panel of the figure showing the covariance matrix after the KLT, all off-diagonal elements become zero, i.e., the signal components are completely decorrelated.

The effect of energy compaction is also represented in the figure by the brightness of the elements along the diagonal, which is reduced gradually from top-left to bottom-right. This effect can be more clearly seen in Fig.9.6 showing the profile of the diagonal of the covariance matrix, the variances of the N signal components after each of the transform methods. We note that the dashed curve representing the energy distribution before any transform (IT) is mostly flat, i.e., the signal energy is relatively evenly distributed among all signal components. The remaining curves of energy distribution after each of the transforms all show some steep descent (high on the left and low on the right), indicating that the signal energy is greatly compacted with most energy concentrated in a small number of signal components (corresponding to mainly low frequencies). In particular, the solid curve corresponding to the KLT has the steepest descent representing the optimal energy compaction. As in Fig. 9.4, here a non-linear mapping $y = x^{0.3}$ is used for low values to be visible as well as the high ones.

The same analysis process is also carried out to the image of sand shown in the right panel of Fig.9.3, which has a drastically different texture from that of the clouds shown in the left panel. This is because the color of a grain of sand is irrelevant to that of the neighboring grains, i.e., the signal components are much less correlated in comparison to the previous case of the image of clouds.

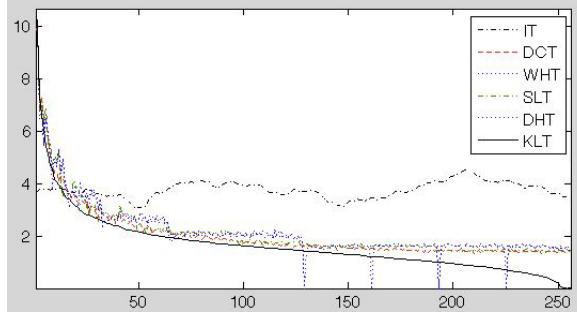


Figure 9.6 Signal energy distribution after various transforms

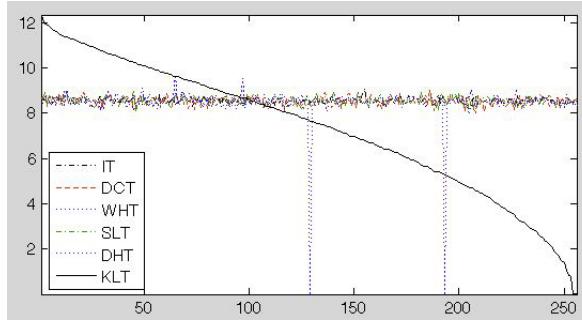


Figure 9.7 Signal energy distribution after various transforms

Consequently, in the covariance matrix of the original signal (IT) shown in the top-left panel of Fig.9.5, all off-diagonal elements look random with relatively low values, indicating that the pixels are not correlated. Also we no longer see the trend of brighter pixels around the diagonal as observed in the covariance matrix for the image of clouds. Moreover, we see that all of the covariance matrices after the transforms shown in the other panels of the figure look very similar to the first one, indicating that the signal is not decorrelated by these transforms, except for the last covariance matrix in the lower-right panel after the optimal transform of KLT, by which the signal is completely decorrelated as indicated by the diagonal covariance matrix.

Same as in the case of signal of clouds, the profiles of the diagonals of the covariance matrices for the sand signal are also plotted in Fig.9.7, showing how the signal energy is distributed among all signal components after the orthogonal transforms. We see that none of the transform methods is able to further compact signal energy, except the optimal KLT by which the signal is still maximally compacted, as shown by the solid curve high on the left low on the right.

The effect of energy compaction can also be quantitatively measured by the entropies of the energy distribution profiles in Figs.9.6 and 9.7, as listed in Table 9.1 for different transform methods applied to both the signals of clouds and sand. We see that for the signal of clouds with significant correlation, all orthog-

Table 9.1. Entropies of signal energy distribution after various transforms

Signal	IT	DCT	WHT	SLT	DHT	KLT
Clouds	7.907	3.237	3.496	3.445	3.494	2.671
Sand	7.998	7.894	7.965	7.995	7.982	7.260

Table 9.2. Number of components needed to keep certain percentage of energy

Energy Percentage:	90%	95%	99%	100%
IT:	209 (82%)	230 (90%)	250 (98%)	256 (100%)
DCT:	10 (4%)	22 (9%)	97 (38%)	256 (100%)
KLT:	7 (3%)	13 (5%)	55 (21%)	256 (100%)

onal transforms perform well in terms of energy compaction as the entropy is significantly reduced after each transform, and the optimal KLT achieves the minimum entropy slightly lower than others. However, most transform methods have very limited effect of energy compaction for the signal of sand with low correlation, except the KLT with the minimum entropy which is significantly lower than those of all other transforms.

The different energy compaction effects achieved by the IT (no transform), DCT and KLT for the cloud signal are also illustrated in Table 9.2, which lists the number (and percentage) of signal components needed in order to keep certain percentage of the total signal energy (information) for data compression. For example, if it is tolerable to lose 5% of the signal energy, out of the total $N = 256$ components we need to keep 230 signal component (90% of data) without any transform, 22 components (9% of data) after the DCT, but only 13 components (5% of data) after the KLT. In other words, using the optimal KLT, we can achieve a data compression rate of $13/256 \approx 5\%$, by keeping only 5% of the data but preserving 95% of the signal energy.

From the two examples above, several observations can be made.

- All orthogonal transforms tend to decorrelate a natural signal and compact its energy, and KLT does it optimally. Typically, after an orthogonal transform, consecutive signal components in the transform domain are much less correlated, and the signal energy tends to be compacted into a small number of signal components. For example, after DFT or DCT, two consecutive frequency components in the spectrum are not likely to be correlated, and most of the signal energy is concentrated in a small number of low frequency components as well as the DC component, while most of the high frequency components carry little energy. These are essentially the reasons why orthogonal transforms are widely used in data processing.

- The general claim that orthogonal transforms tend to reduce signal correlation and compact signal energy is based on the implicit assumption that time or spatial signals in most applications are mostly continuous and smooth due to the nature of underlying physics. However, this assumption may not be necessarily true in every single case. In fact, the effects of signal decorrelation and energy compaction depend on the nature of the specific signal at hand. These effects may not be obvious in some unlikely cases where the signal is not correlated to start with, such as the image of sands.
- The KLT is optimal among all orthogonal transforms in terms of signal decorrelation and energy compaction. However, in many cases the performance of other transforms, such as the DCT, are not too different from that of the KLT. Although suboptimal, such a transform is often used due to its fast algorithm for much reduced computational complexity.

Although the KLT is optimal among all orthogonal transforms, other orthogonal transforms are still widely used for two reasons. First, by definition the KLT transform is for random signals and it depends on the specific data being analyzed. The transform matrix $\Phi = [\phi_0, \dots, \phi_{N-1}]$ is composed of the eigenvectors of the covariance matrix Σ_x of the signal x , which can be estimated only when enough data are available. Second, the computational cost of the KLT transform is much higher than other orthogonal transforms. The computational complexity of the eigenvalue problem of the N-D covariance matrix is $O(N^3)$, while the complexity for any other orthogonal transform based on a predetermined transform matrix is no worse than $O(N^2)$. Moreover, fast algorithms with complexity $O(N \log_2 N)$ exist for most transforms such as DFT, DCT, and WHT. For these reasons, the DFT, DCT or some other transforms may be the preferred method in many applications. The KLT can be used when the covariance matrix of the data can be estimated and computational cost is not critical. Also the KLT as the optimal transform can be used to serve as a standard against which all other transform methods can be compared and evaluated.

9.2.7 Approximation of KLT by DCT

Although no fast algorithm exists for the KLT, it can be approximated by the discrete cosine transform DCT if the signal is locally correlated and therefore can be modeled as a first-order Markov process with Toeplitz correlation matrix \mathbf{R} (Eq.9.12). Specifically, we will show that when the correlation of Markov process approaches one, its KLT transform approaches the DCT. The proof is a two-step process: (1) find the KLT matrix for the Markov process by solving the eigenvalue problem of its correlation matrix \mathbf{R} , and (2), let $r \rightarrow 1$ and show the KLT matrix approaches the DCT matrix.

The KLT matrix of a first-order Markov process is the eigenvector matrix $\Phi = [\phi_0, \dots, \phi_{N-1}]$ of the Toeplitz correlation matrix \mathbf{R} :

$$\mathbf{R}\Phi = \Phi\Lambda, \quad \text{i.e.,} \quad \Phi^T \mathbf{R}\Phi = \Lambda \quad (9.53)$$

As \mathbf{R} is symmetric (self-adjoint), all λ_n are real and all ϕ_n are orthogonal. Also, it can be shown¹ that Φ and Λ of the Toeplitz correlation matrix \mathbf{R} take the following forms:

- The nth eigenvalue is:

$$\lambda_n = \frac{1-r}{1-2r\cos\omega_n+r^2}, \quad (n=0, \dots, N-1) \quad (9.54)$$

- The mth element ϕ_{mn} of the nth eigenvector $\phi_n = [\dots, \phi_{mn}, \dots]^T$ is:

$$\phi_{mn} = \left(\frac{2}{N+\lambda_n} \right)^{1/2} \sin \left(\omega_n \left(m - \frac{N-1}{2} \right) + (n+1)\frac{\pi}{2} \right) \quad (9.55)$$

- In the above, ω_n ($n=0, \dots, N-1$) are the N real roots of the following equation:

$$\tan(N\omega) = -\frac{(1-r^2)\sin\omega}{(1+r^2)\cos\omega - 2r} \quad (9.56)$$

The proof for these expressions is lengthy and therefore omitted here.

Next we consider the three expressions given above when $r \rightarrow 1$. First, Eq.9.56 simply becomes:

$$\tan(N\omega) = 0 \quad (9.57)$$

Solving this for ω we get:

$$\omega_n = n\pi/N \quad (9.58)$$

However, when $n=0$, $\omega_0=0$ and $\cos\omega_0=1$, and Eq.9.57 becomes an indeterminate form 0/0. But applying L'Hopital's rule twice yields:

$$\lim_{\omega \rightarrow 0} \tan(N\omega) = \lim_{\omega \rightarrow 0} \frac{0}{2\cos\omega} = 0 \quad (9.59)$$

i.e., $\omega_0=0$ is still a valid root for Eq.9.56. Having found $\omega_n=n\pi/N$ for all $0 \leq n \leq N-1$, we can further find the eigenvalues λ_n in Eq.9.54 when $r \rightarrow 1$. For $n > 0$, $\omega_n \neq 0$ and $\cos\omega_n \neq 1$, we have:

$$\lambda_n = \lim_{r \rightarrow 1} \frac{1-r}{1-2r\cos\omega_n+r^2} = 0, \quad (1 \leq n \leq N-1) \quad (9.60)$$

We also get $\lambda_0=N$ by noting that the second equation in Eq.9.53 is a similarity transformation of \mathbf{R} which conserves its trace:

$$tr\mathbf{R} = N = tr\Lambda = \sum_{n=0}^{N-1} \lambda_n = \lambda_0 \quad (9.61)$$

¹ Ray, W.D. and Driver, R.M., Further decomposition of the Karhunen-Loeve series representation of a stationary process, *IEEE Transaction on Information Theory*, 16(6), November 1970

We can now find the elements ϕ_{mn} in the eigenvector ϕ_n . For all $n > 0$, we have $\lambda_n = 0$ and $\omega_n = n\pi/N$, Eq.9.55 becomes:

$$\begin{aligned}\phi_{mn} &= \sqrt{\frac{2}{N}} \sin \left(\frac{n\pi}{N} \left(m - \frac{N-1}{2} \right) + (n+1) \frac{\pi}{2} \right) = \sqrt{\frac{2}{N}} \sin \left(\frac{n\pi}{2N} (2m+1) + \frac{\pi}{2} \right) \\ &= \sqrt{\frac{2}{N}} \cos \left(\frac{n\pi}{2N} (2m+1) \right), \quad (0 \leq m \leq N-1, 1 \leq n \leq N-1)\end{aligned}\quad (9.62)$$

When $n = 0$, $\omega_0 = 0$ and $\lambda_0 = N$, and Eq.9.55 becomes:

$$\phi_{m0} = \sqrt{\frac{1}{N}} \sin \left(\frac{\pi}{2} \right) = \sqrt{\frac{1}{N}}, \quad (0 \leq m \leq N-1) \quad (9.63)$$

This happens to be precisely the DCT transform matrix derived in section 7.2.3, and we can therefore conclude that the KLT of a first order Markov process approaches the DCT when $r \rightarrow 1$.

However, we note that the result above cannot be extended to the limit of $r = 1$, as when $r = 1$ all elements of \mathbf{R} become 1, and its eigenvectors are no longer unique. In fact, the column vectors of many other orthogonal transform matrix \mathbf{A} are the eigenvectors of this all-1 matrix \mathbf{R} :

$$\mathbf{A}^T \mathbf{R} \mathbf{A} = \mathbf{\Lambda} = \text{diag}[N, 0, \dots, 0] \quad (9.64)$$

i.e.,

$$\mathbf{a}_m^T \mathbf{R} \mathbf{a}_n = \begin{cases} N & m = n = 0 \\ 0 & \text{else} \end{cases} \quad (9.65)$$

To see this, we note that the first column \mathbf{a}_0 of any orthogonal transform matrix $\mathbf{A} = [\mathbf{a}_0, \dots, \mathbf{a}_{N-1}]$ (DFT, WHT, as well as DCT, except DST) is always composed of N constants $1/\sqrt{N}$ (representing the DC component), and as all other columns \mathbf{a}_n ($n > 0$) are orthogonal to \mathbf{a}_0 , they all sum up to zero:

$$\langle \mathbf{a}_n, \mathbf{a}_0 \rangle = \mathbf{a}_n^T \mathbf{a}_0 = \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} a[m, n] = 0 \quad (9.66)$$

As the result, all elements of matrix $\mathbf{A}^T \mathbf{R} \mathbf{A}$ in Eq.9.64 are zero:

$$\mathbf{a}_m^T \mathbf{R} \mathbf{a}_n = \mathbf{a}_m^T \begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{bmatrix} \mathbf{a}_n = \mathbf{a}_m^T \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix} = 0, \quad (m \neq 0 \text{ or } n \neq 0) \quad (9.67)$$

except when $m = n = 0$, the top-left element is

$$\mathbf{a}_0^T \mathbf{R} \mathbf{a}_0 = \frac{1}{N} [1, \dots, 1]^T \begin{bmatrix} 1 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} = N \quad (9.68)$$

The approximation of the KLT of a first-order Markov process by the DCT can also be seen from another point of view. It can be shown that the N DCT

basis vectors, the column vectors of the DCT matrix $\mathbf{C} = [\mathbf{c}_0, \dots, \mathbf{c}_{N-1}]$, are the eigenvectors of the tridiagonal matrix of the following form (independent of the parameter α):

$$\mathbf{Q} = \begin{bmatrix} 1 - \alpha & -\alpha & 0 & \cdots & 0 \\ -\alpha & 1 & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & 1 & -\alpha \\ 0 & \cdots & 0 & -\alpha & 1 - \alpha \end{bmatrix} \quad (9.69)$$

i.e., $\mathbf{C}^T \mathbf{Q} \mathbf{C} = \mathbf{M}$, where $\mathbf{M} = \text{diag}(\mu_0, \dots, \mu_{N-1})$ is a diagonal matrix composed of N eigenvalues of \mathbf{Q} .

On the other hand, it can also be shown that the inverse of the correlation matrix \mathbf{R} of a first order Markov process given in Eq.9.12 takes the following form:

$$\mathbf{R}^{-1} = \frac{1}{\beta} \begin{bmatrix} 1 - r\alpha & -\alpha & 0 & \cdots & 0 \\ -\alpha & 1 & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & 1 & -\alpha \\ 0 & \cdots & 0 & -\alpha & 1 - r\alpha \end{bmatrix} \quad (9.70)$$

where

$$\alpha = \frac{r}{1 + r^2}, \quad \beta = \frac{1 - r^2}{1 + r^2} \quad (9.71)$$

i.e., $\Phi^T \mathbf{R}^{-1} \Phi = \mathbf{M}^{-1}$ or $\Phi^T \mathbf{R} \Phi = \Lambda$, where $\Lambda = \text{diag}(\lambda_0, \dots, \lambda_{N-1})$ is a diagonal matrix composed of N eigenvalues of \mathbf{R} .

It is therefore clear that the N KLT basis vectors, the column vectors of the eigenvector matrix $\Phi = [\phi_0, \dots, \phi_{N-1}]$ of \mathbf{R} (same as that of \mathbf{R}^{-1}), can be approximated by the DCT basis vectors if $r \rightarrow 1$. Note, again, that the approximation breaks down when $r = 1$ and $\beta = 0$.

As an example, Fig.9.8 shows the first 8 of the $N = 128$ basis vectors of the KLT of a Markov process with $r = 0.9$, in comparison to the corresponding DCT basis vectors. Note that the KLT basis vectors match those of the DCT very closely and the similarity will increase when r approaches 1. Also note that as an eigenvector of \mathbf{R} , a KLT vector ϕ_n , can be either positive or negative, i.e., the corresponding transform coefficients of the KLT and DCT may have opposite polarity. However, this does not affect the transform as the reconstructed signal will still be the same. Also shown in the left and right panels of Fig.9.9 are the 3-D plots of the covariance matrices after the KLT and DCT of a Markov process. We see that the two transforms are very similar in terms of the energy compaction and signal decorrelation. The performances of the DCT are almost as good as the optimal KLT.

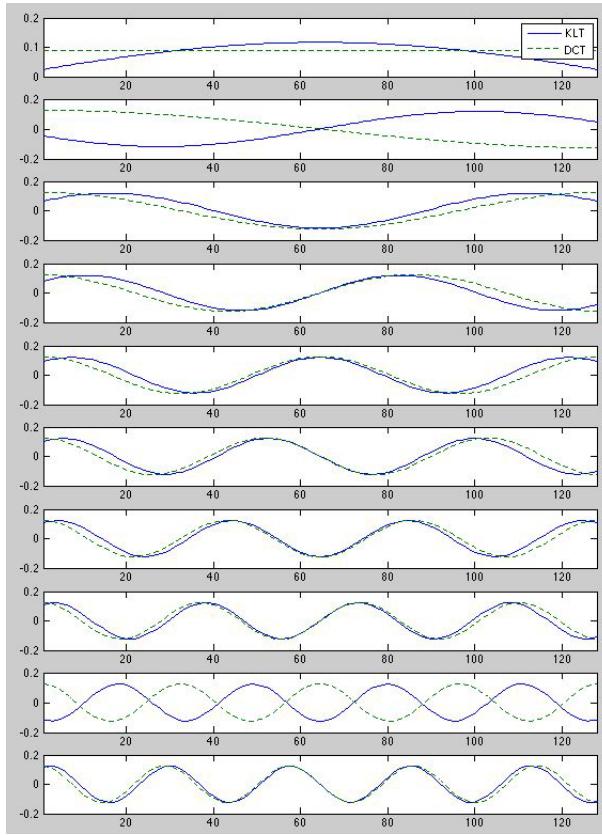


Figure 9.8 Comparison of the first 8 basis vectors of the DCT and KLT of 1st order Markov process

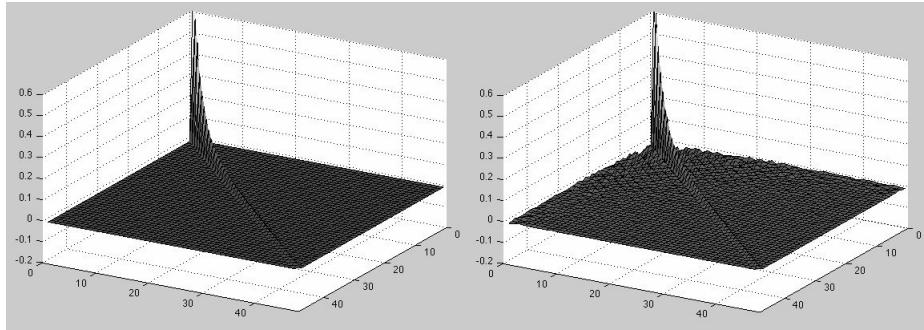


Figure 9.9 3-D plots of the covariance matrices after the KLT (left) and DCT (right)

The result above has important significance. As most signals of interest in practice are likely to be locally correlated and can therefore be modeled by a first order Markov process, we can always expect the results of the DCT transform are close to the optimal transform of KLT. Furthermore, as the basis vectors

of the KLT are the eigenvectors of the signal covariance Σ_x corresponding to the eigenvalues arranged in descending order, they are actually arranged according the energy contained in signal components (represented by the eigenvalues). Consequently, as the KLT is approximated by the DCT, its first principal component corresponding to the DC component contains the largest amount of energy, and the subsequent components corresponding to progressively higher frequencies in the DCT contain progressively lower energy. This approximation is valid in general for all locally correlated signals.

To illustrate this fact, we consider a dataset of annual temperatures in Los Angeles area collected over the period of 1878-1997, shown in the top panel of Fig.9.10. To obtain the covariance of a sequence of $n = 8$ samples of the data, we truncate the signal into a set of segments of n samples each, and treat these segments as random samples from a stochastic process. We next obtain the n by n covariance matrix of this data, as shown in the lower left panel of the figure. We see that the elements around the diagonal of the matrix have high values, indicating that the signal samples are highly correlated when they are close to each other (taken within a short duration), but the values of the elements farther away from the diagonal are much reduced, indicating that the signal samples are much less correlated when they are far apart (separated by a long period of time). This kind of behavior can be modeled by a first-order Markov chain of n points whose covariance is shown in the lower right panel of the same figure (correlation between two consecutive samples assumed to be $r = 0.5$), which looks similar to the covariance of the actual signal, in the sense that the correlation is gradually reduced between signal samples when they are farther apart.

Next we further consider the KLT transform of the signal and its approximation by the DCT. The KLT transform matrix is composed of the n eigenvectors of the signal covariance, shown in the panels on the left of Fig.9.11, which are compared to the eigenvectors of the covariance of the Markov model (solid curves) shown in the panels on the right of the figure. These two sets of curves look similar in terms of the general wave forms and their frequencies (not necessarily in the same order). Moreover, comparing the eigenvectors based on the Markov model with the rows of the DCT transform matrix, also shown in the panels on the right (dashed curves), we see that they match very closely (except certain phase shifts).

We can make the following observations based on this example:

- The temperature time function, as one of the weather parameters representing a natural process, confirms the general assumption that the correlation between signal samples tends to decay as they are farther apart.
- The signal correlation can be indeed closely modeled by a first order Markov chain model with a correlation r and the only parameter.
- The eigenvectors of the covariance matrix above can be closely matched by the row vectors of the DCT transform matrix.

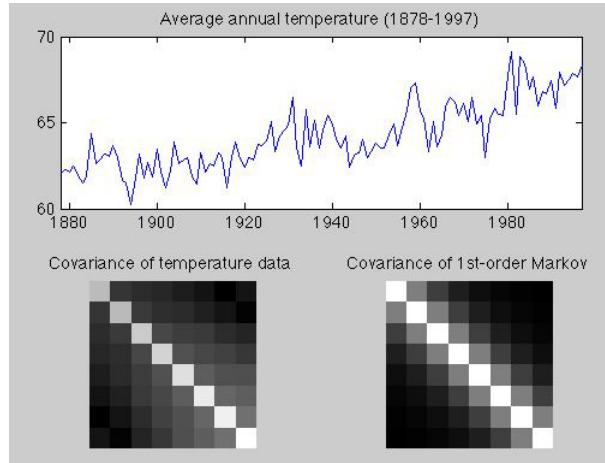


Figure 9.10 Covariances of natural signal and 1st order Markov chain

- Based on the observations above, we conclude that the KLT transform of a typical natural signals can be approximately carried out as a DCT transform.
- In particular, the first eigenvector ϕ_0 corresponding to the largest eigenvalue is approximated by the first row of the DCT matrix composed of all constants, representing the first principal component $y_0 = \langle \mathbf{x}, \phi_0 \rangle = \phi_0^* \mathbf{x}$ is the average (DC component) of all elements in signal \mathbf{x} .

9.3 Applications of the KLT

9.3.1 Image processing and analysis

The KLT can be carried out on a set of N images for various purposes such as feature extraction and data compression. There are two alternative ways to carry out the KLT on the N images, depending on how a random signal is defined. First, an N -D vector can be formed by N pixels each taken at the same position (e.g., i th row and j th column) from one of the N images. The number of such vectors is obviously the total number of pixels in each image, assumed to be K , and they form a K by N matrix \mathbf{D} , whose covariance matrix can be estimated as (Eq.9.11):

$$\hat{\Sigma}_x = \frac{1}{K-1} [\mathbf{D}^T \mathbf{D}]_{N \times N} \quad (9.72)$$

Alternatively, a K -dimensional vector can be formed by concatenating the rows (or columns) of each of the N image, and each of these vectors from the N images can be treated as a sample of a K -D random vector, represented by a column of \mathbf{D} defined above, or a row of \mathbf{D}^T , and the covariance matrix can be estimated

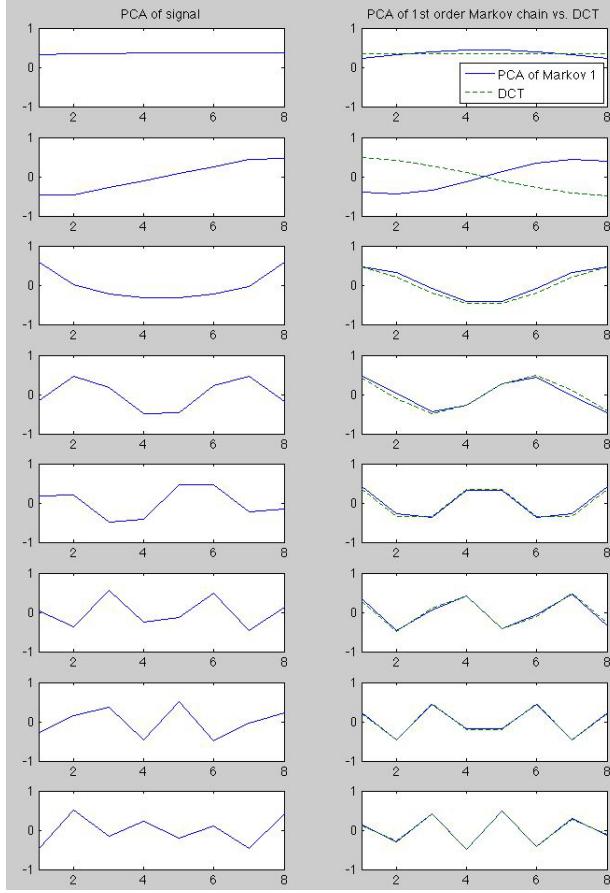


Figure 9.11 KLT of signal (left) compared with KLT of Markov model and DCT (right)

as:

$$\hat{\Sigma}'_x = \frac{1}{N-1} [\mathbf{D}\mathbf{D}^T]_{K \times K} \quad (9.73)$$

We can show the eigenvalue problems of these two different covariance matrices are equivalent. First assume the eigenequations for $\mathbf{D}^T\mathbf{D}$ and $\mathbf{D}^T\mathbf{D}$ are:

$$\mathbf{D}^T\mathbf{D}\phi = \lambda\phi, \quad \mathbf{D}\mathbf{D}^T\psi = \mu\psi \quad (9.74)$$

Pre-multiplying \mathbf{D}^T on both sides of the second equation we get:

$$\mathbf{D}^T\mathbf{D}[\mathbf{D}^T\psi] = \mu[\mathbf{D}^T\psi] \quad (9.75)$$

This is actually the first eigenequation with the same eigenvalue $\mu = \lambda$ and eigenvector $\mathbf{D}^T\psi$, which is the same as ϕ , when both are normalized. The two covariance matrices Σ_x and Σ'_x have the same rank $R = \min(N, K)$ (if D is not degenerate) and therefore the same number of non-zero eigenvalues. Consequently, the

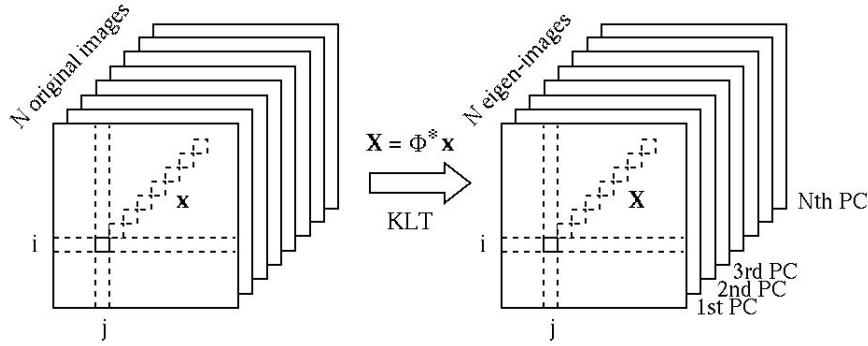


Figure 9.12 KLT of a set of images

KLT can be carried out based on either matrix with the same effects in terms of the signal decorrelation and energy compaction. In the likely case where the number of image pixels is greater than the number of images, $K > N$, we will take the first approach above to treat the same pixels from all N images as a sample of an N -D random signal vector and carry out the KLT based on the $N \times N$ covariance matrix $\hat{\Sigma}_x = \mathbf{D}^T \mathbf{D}/(K - 1)$. Each of the K pixels represented by a vector x can then be transformed to a new vector $X = \Phi^* x$ for the same pixel of a set of N *eigen-images*, as illustrated in Fig. 9.12. Due to the nature of the KLT, most of the energy/information contained in the N images, representing the variations among all N images, is now concentrated in the first M eigen-images ($M < N$), so that the remaining $N - M$ eigen-images can be omitted without losing much energy/information. This is the foundation for various KLT-based image feature extraction/classification and image compression algorithms, all of which could be carried out in a much lower dimensional space.

Example 9.1: In *remote sensing*, the images of the surface of Earth (or other planets) are taken by orbiting satellites, for various studies in fields such as geology, geography and agriculture. The camera system on the satellite has a set of N sensors each sensitive to a different wavelength band in the visible and infrared range of the electromagnetic spectrum. Depending on the number of sensors N , the image data collected are either multi-spectral ($N < 10$) or hyperspectral (N is up to 200 or more). For instance, the $N = 210$ bands of the HYDICE (Hyperspectral Digital Imagery Collection Experiment) data cover the wavelength range from 400 to 2500 nm (10^{-9} meter) with 10 nm separation between two neighboring bands. In this example, we choose 20 bands separated by 100 nm from a set of HYDICE image data (Lincoln Memorial, Washington DC)², as shown in Fig. 9.13 (showing only 10 of the 20 images). We see that the images corre-

² Credit to the School of Electrical and Computer Engineering, ITaP and LARS, Purdue University

Table 9.3. Energy distribution (percentage) among 20 signal components

Component	0	1	2	3	4	5	6	7	8	9
Before KLT	3.7	3.7	3.7	3.9	5.0	5.4	6.8	6.9	7.5	8.3
After KLT	70.6	23.3	4.6	0.6	0.5	0.1	0.1	0.1	0.1	.04
Cont'd	10	11	12	13	14	15	16	17	18	19
	7.5	5.7	6.4	4.9	2.9	3.8	3.7	3.3	2.3	4.7
	.02	.02	.02	.01	.01	.01	.00	.00	.00	

sponding to neighboring wavelength bands are often similar to each other and therefore redundant, i.e., they are highly correlated. (Obviously in the complete HYDICE data, the 210 bands separated by 10 nm are even more highly correlated.) When the KLT is carried out on these $N = 20$ dimensional vectors (each for a pixel in the images), 20 PCA images can be obtained as shown in Fig.9.14 (again showing only 10). Two observations can be made. First, after the KLT the images are completely decorrelated, as the PCA images all look different, each carrying its independent information. Second, the signal energy is highly compacted into the first few PCAs, as also seen in Fig.9.15 and Table 9.1 for the energy distributions before and after the KLT. The data can be compressed by keeping only the first three PCA components (15% of data) containing 98.5% of the total energy/information.

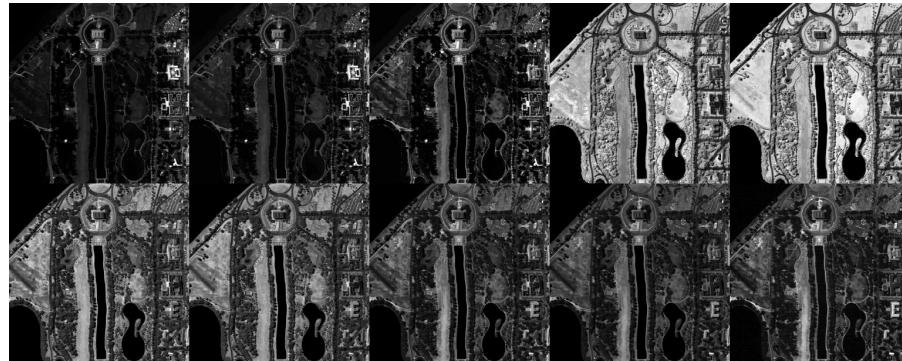


Figure 9.13 Ten spectral bands of the HYDICE image data (Lincoln Memorial, Washington DC)

Example 9.2: Consider a sequence of $N=8$ frames of a video of a moving escalator and their eigen-images shown respectively in the upper and lower parts of Fig. 9.16. The covariance matrix and the energy distribution among the 8 components plot both before and after the KLT are shown in Fig.9.17. We see that due to the local correlation, the covariance matrix before the KLT (left) does indeed

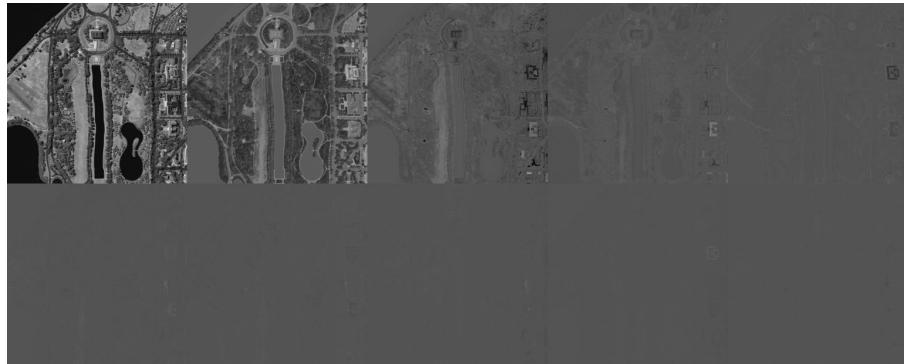


Figure 9.14 Ten PCA images after the KLT

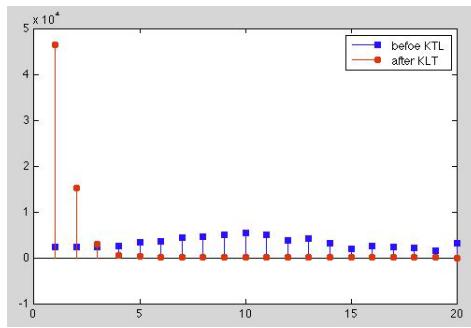


Figure 9.15 Signal energy distributions among 20 signal components before and after the KLT

resemble the correlation matrix \mathbf{R} of a first order Markov model (bottom right in Fig.9.11), and the covariance matrix after the KLT (middle) is completely decorrelated and its energy highly compacted, as also clearly shown in the comparison of the energy distribution before and after the KLT (right). Also as shown in Eq.9.18, the KLT basis is very much similar to the DCT basis, indicating that the DCT with a fast algorithm would generate almost the same results as the KLT. Moreover, it is interesting to observe that the first eigen-image (left panel of the 3rd row of Fig.9.16 represents mostly the static scene of the image frames corresponding to the mail variations in the image (carrying most of the energy), while the subsequent eigen-images represent mostly the motion in the video, the variation between the frames. For example, the motion of the people riding on the escalator is mostly reflected by the first few eigen-images following the first one, while the motion of the escalator stairs is mostly reflected in the subsequent eigen-images.

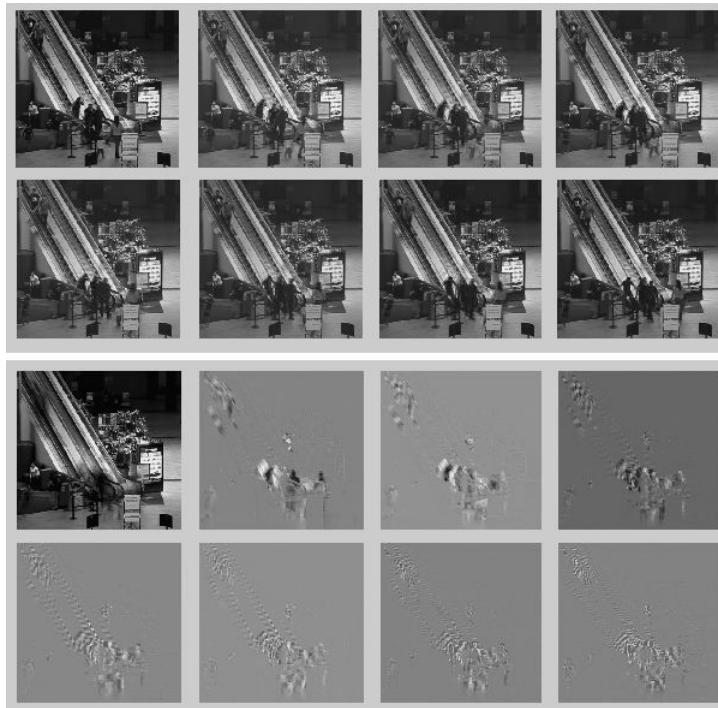


Figure 9.16 Video frames (top) and the eigen-images (bottom)

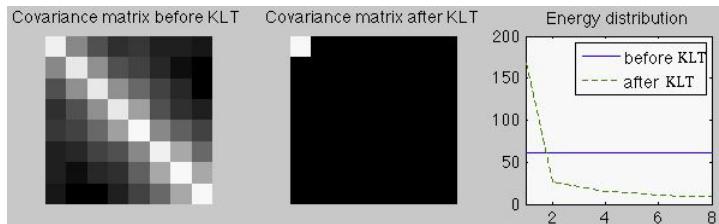


Figure 9.17 Covariance matrix before and after KLT

The covariance matrices before and after the KLT are shown in image form (left and middle), while the energy distributions among the N components before and after the transform are also plotted (right).

Example 9.3: A set of $N=20$ face images are shown in the top panel of Fig.9.19³. The KLT is carried out on these images to obtain the eigen-images, called in this case *eigen-faces* (middle panel). It can be seen that the first few *eigenfaces* capture the most essential common features shared by all faces. Specifically, the first eigen-face represents a generic face in the dark background, while the second eigen-face represents the darker hair versus the brighter face. The rest of

³ Credit to AT&T Laboratories Cambridge

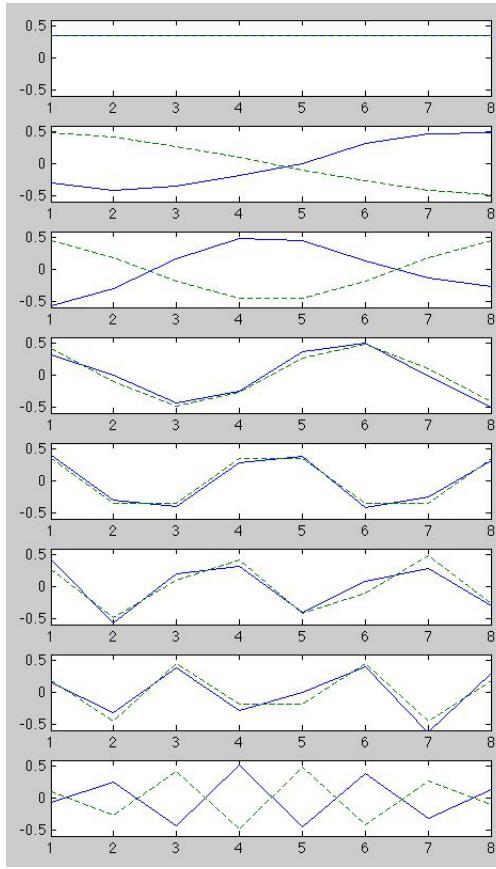


Figure 9.18 KLT basis vectors compared with DCT basis

The basis vectors of the KLT of the video frames closely resemble the DCT basis vectors (may be with opposite polarities).

the eigenfaces represent some other features with progressively less significance. The table below shows the percentage of energy contained in each component:

The faces are then reconstructed based on 95% of the total information as shown in the bottom panel in Fig.9.19. The method of eigenfaces is used in facial recognition and classification.

9.3.2 Feature Extraction for Pattern Classification

In the field of machine learning, pattern classification/recognition is a general method that classifies a set of objects of interest into different *categories* or *classes* and recognizes any given object as a member of one of these classes. Specifically, each object is represented as an N-D vector, known as a *pattern*, based on a set of *N features* that can be observed and measured to characterize the object. Then



Figure 9.19 Original faces (top), Eigenfaces (middle), and Reconstructed faces (bottom)

# of components	1	2	3	4	5	6	7	8
% energy contained	48.5	11.6	6.1	4.6	3.8	3.7	2.6	2.5
accumulative	48.5	60.1	66.2	70.8	74.6	78.3	81.0	83.5
Cont'd	9	10	11	12	13	14	15	16
	1.9	1.9	1.8	1.6	1.5	1.4	1.3	1.2
	85.4	87.3	89.0	90.7	92.2	93.6	94.9	96.1
Cont'd	17	18	19	20				
	1.1	1.1	0.9	0.8				
	97.2	98.2	99.2	100.0				

a pattern classification algorithm can be carried out in this N-D *feature space* to classify all pattern vectors in it. The classification is therefore essentially the partitioning of the feature space into a set of regions each corresponding to one particular class. A given pattern is classified to the class corresponding to the region in which it resides. There are in general two types of pattern classification algorithms, depending on whether certain *a priori* knowledge or information regarding the classes is available. An algorithm is *supervised* if it is based on the assumed availability of a set of patterns with known classes, called *training samples*. When such training samples can not be obtained, an *unsupervised* algorithm has to be used. If the number of features N is large, especially if the

N features are not all pertinent to the representation of the classes of interest, a process called *feature extraction* indexfeature extraction is needed to find a set of $M < N$ features to form a much lower dimensional feature space in which the classification can be more effectively and efficiently carried out. These M features can be either directly chosen from the N original features, or they can be generated based on N original features.

For example, in the hyperspectral remote sensing image data , at each pixel position a set of N values corresponding to N bands of wavelengths form a pattern vector in the N-D feature space, representing the spectral signature of the surface material covered by the pixel. All pixels in the image data can therefore be classified according to their spectral signatures into different classes of the surface materials of interest, such as vegetations (e.g., different types of crops and forests), water bodies (e.g., oceans, lakes, rivers), soil and rocks (e.g., different types of minerals), snow and ice, desert, man-made objects (e.g., pavement, roads, and buildings). For example, the spectral signatures ($N = 191$ of four different ground cover types, water, grass, trees and building roofs, of the HYDICE image data used in Example 9.1 are shown in Fig.9.20.

For another example, In image recognition, some objects given in image form, e.g., one of the 26 letters or the 10 digits from 0 to 9, may need to be recognized. Extracting from the image a set of relevant features representing the patterns may be difficult as it requires specific knowledge regarding the objects of interest. A more straightforward way of representing such image objects is to simply use all N pixels in the image (e.g., $N = 256$ for 16×16 images), arranged as an N-D vector pattern obtained by concatenating its rows or columns of the image.

A challenge in both examples above is that the number of features N is large ($N = 191$ or $N = 256$), and not all of them are necessarily pertinent to the classification of the specific classes of interest. In such cases, we need to carry out the feature extraction as a pre-processing process to find a set of $M < N$ features most relevant to the subsequent classification. Due to the property of optimal energy compaction stated in Theorem 9.1, the KLT $\mathbf{X} = \Phi^* \mathbf{x}$ can be applied to generate such a set of M new features as the linear combination of the N original feature. However, the KLT matrix Φ can no longer be based on the covariance matrix Σ_x (Eq.9.16), which represents the variations among all pattern vectors in the data. Instead, here the KLT matrix Φ needs to be based on some different matrix that reflects the differences between the classes to be distinguished.

Let $\{\mathbf{x}_i^{(k)}, (i = 1, \dots, n_k)\}$ be a set of n_k N-D vectors for the training samples known to belong to class k , where $k = 1, \dots, K$ for all classes. Based on these training samples we can define the following *scatter matrices*:

- *Scatter or covariance matrix* of class k for the variation or scatteredness within the class:

$$\mathbf{S}_k = \frac{1}{n_k} \sum_{i=1}^{n_k} (\mathbf{x}_i^{(k)} - \mathbf{m}_k)(\mathbf{x}_i^{(k)} - \mathbf{m}_k)^T, \quad (k = 1, \dots, K) \quad (9.76)$$

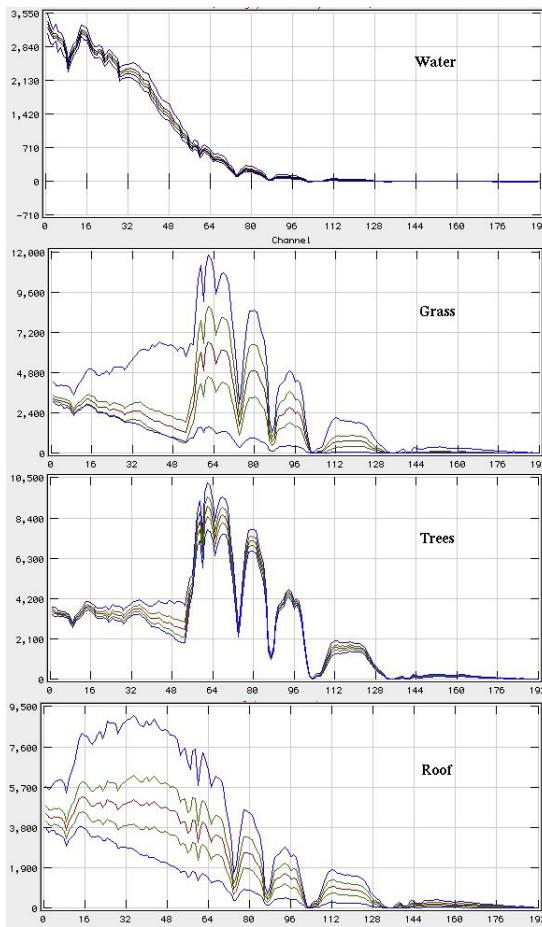


Figure 9.20 Spectral signatures of four ground cover types

The five curves are for the maximum, mean+standard deviation, mean, mean-standard deviation and minimum, respectively.

where \mathbf{m}_k is the mean vector of the k th class:

$$\mathbf{m}_k = \frac{1}{n_k} \sum_{i=1}^{n_k} \mathbf{x}_i^{(k)}, \quad (k = 1, \dots, K) \quad (9.77)$$

- *Within-class scatter matrix* for the average within-class scatteredness of all K classes:

$$\mathbf{S}_w = \sum_{k=1}^K p_k \mathbf{S}_k = \frac{1}{n} \sum_{k=1}^K n_k \mathbf{S}_k, \quad (9.78)$$

where $n = \sum_{k=1}^K n_k$ is the total number of training samples of all K classes, and $p_k = n_k/n$.

- *Between-class scatter matrix* for the separability, or the variation between all K classes:

$$\mathbf{S}_b = \sum_{k=1}^K p_k (\mathbf{m}_k - \mathbf{m})(\mathbf{m}_k - \mathbf{m})^T, \quad (9.79)$$

where \mathbf{m} is the mean vector of all n training samples of all K classes:

$$\mathbf{m} = \frac{1}{n} \sum_{\mathbf{x}} \mathbf{x} = \frac{1}{n} \sum_{k=1}^K n_k \frac{1}{n_k} \sum_{i=1}^{n_k} \mathbf{x}_i^{(k)} = \sum_{k=1}^K p_k \mathbf{m}_k \quad (9.80)$$

- *Total scatter or covariance matrix* for the total variation among all n samples of the K classes:

$$\begin{aligned} \mathbf{S}_t &= \frac{1}{n} \sum_{\mathbf{x}} (\mathbf{x} - \mathbf{m})(\mathbf{x} - \mathbf{m})^T \\ &= \frac{1}{n} \sum_{k=1}^K \sum_{i=1}^{n_k} (\mathbf{x}_i^{(k)} - \mathbf{m}_k + \mathbf{m}_k - \mathbf{m})(\mathbf{x}_i^{(k)} - \mathbf{m}_k + \mathbf{m}_k - \mathbf{m})^T \\ &= \frac{1}{n} \sum_{k=1}^K \sum_{i=1}^{n_k} (\mathbf{x}_i^{(k)} - \mathbf{m}_k)(\mathbf{x}_i^{(k)} - \mathbf{m}_k)^T + \frac{1}{n} \sum_{k=1}^K \sum_{i=1}^{n_k} (\mathbf{m}_k - \mathbf{m})(\mathbf{m}_k - \mathbf{m})^T \\ &= \mathbf{S}_w + \mathbf{S}_b \end{aligned} \quad (9.81)$$

The second to the last equal sign is due to the fact that

$$\sum_{k=1}^K \sum_{i=1}^{n_k} (\mathbf{x}_i^{(k)} - \mathbf{m}_k)(\mathbf{m}_k - \mathbf{m})^T = 0 \quad (9.82)$$

The equation $\mathbf{S}_t = \mathbf{S}_w + \mathbf{S}_b$ indicates that the total scatteredness \mathbf{S}_t of the n samples is due to the contributions of the total within-class scatteredness \mathbf{S}_w and the total between-class scatteredness \mathbf{S}_b , as one would intuitively expect.

Now we can carry out the KLT based on the between-class scatter matrix \mathbf{S}_b , so that most of the information specifically representing the separability of the K classes (for different surface materials in remote sensing or letters/digits in character recognition) will be compacted into a small number of M components after the transform. The classification/recognition can then be carried out in the resulting M-D feature space containing most of the information relevant to the classification (separability) with much reduced computational complexity, by certain classification algorithm. As a simple example, we could classify a given pattern \mathbf{x} to the class with minimum distance $D(\mathbf{x}, \mathbf{m}_k)$ between its mean and the pattern \mathbf{x} :

$$\mathbf{x} \text{ belongs to class } k \text{ iff } D(\mathbf{x}, \mathbf{m}_k) \leq D(\mathbf{x}, \mathbf{m}_l), \quad (l = 1, \dots, K) \quad (9.83)$$

Another application of the KLT is data visualization. it may be desirable to be able to intuitively assess the data by visualizing how the data points are distributed in the N-D feature space. However, visualization is obviously impossible when $N > 3$. In such cases the KLT transform based on the overall covariance

matrix of the data can be used to project the data points from the original N-D space to a 2 or 3-D space in which most of the information characterizing the distribution of the data points in the feature space is conserved for visualization.

Example 9.4: Here we consider the classification of the 10 digits from 0 to 9, each written multiple times by students in a class, in the form of a 16×16 image, as shown in top-left panel of Fig.9.21. Each pattern can be simply represented by the $N = 256 = 16 \times 16$ pixels in the image, which can be converted to an N-D vectors obtained by concatenating the rows of its image. Based on \mathbf{S}_b representing the separability of the 10 classes, the KLT can be carried out. The energy distribution plots both before and after the KLT are shown in the two right panels in Fig.9.21. Different from the KLT based on the covariance matrix of the data as discussed previously, here the KLT is based on the between-class scatter matrix \mathbf{S}_b , and consequently the energy in question represents specifically the separability information most pertinent to the classification of the 10 digits. From the distribution plots we see that before the KLT, the energy is relatively evenly distributed through out most of the 256 pixels with high local correlation in the same row (each corresponding to one of the 16 peaks in the plot), but after the KLT, the energy is highly compacted into the first 9 principal components, while the remaining $256 - 9 = 247$ components contain little energy and therefore can be omitted. The classification can then be carried out in the M=9 dimensional feature space with much reduced computational cost. Also, in order to visualize the information contained in the 9-D space used in the classification, we can carry out the inverse KLT to reconstruct the images based on these 9 components (Eq.9.50), as shown in the bottom-left panel of the figure. We see that these images contain most of the information pertinent to the classification, in that the within-class variation is minimized while the between-class variation is maximized.

9.4 Singular Value Decomposition Transform

9.4.1 Singular Value Decomposition (SVD)

The *singular value decomposition (SVD)* of an $M \times N$ matrix \mathbf{A} of rank $R \leq \min(M, N)$ is based on the following eigenvalue problems of an $M \times M$ matrix \mathbf{AA}^* and an $N \times N$ matrix $\mathbf{A}^*\mathbf{A}$:

$$\begin{aligned}\mathbf{AA}^*\mathbf{u}_n &= \lambda_n \mathbf{u}_n, & (n = 1, \dots, M) \\ \mathbf{A}^*\mathbf{A}\mathbf{v}_n &= \lambda_n \mathbf{v}_n, & (n = 1, \dots, N)\end{aligned}\tag{9.84}$$

As the rank of \mathbf{A} is R , there exist only R non-zero eigenvalues. As both \mathbf{AA}^* and $\mathbf{A}^*\mathbf{A}$ are symmetric (self-adjoint), their eigenvalues λ_n are real and their

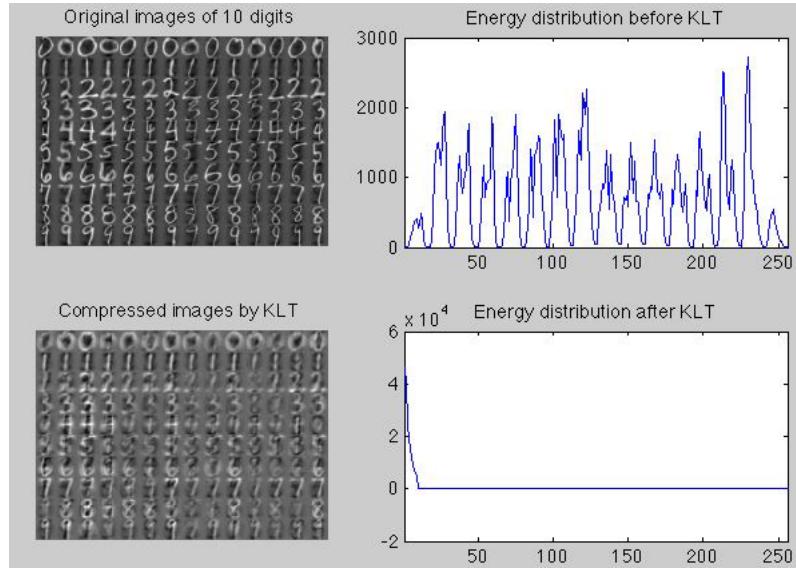


Figure 9.21 KLT of image pattern classification based on between-class scatter matrix

eigenvector \mathbf{u}_n and \mathbf{v}_n are orthogonal:

$$\mathbf{u}_m^* \mathbf{u}_n = \mathbf{v}_m^* \mathbf{v}_n = \delta[m - n] \quad (9.85)$$

and they form two orthogonal matrices $\mathbf{U} = [\mathbf{u}_1, \dots, \mathbf{u}_N]_{M \times M}$ and $\mathbf{V} = [\mathbf{v}_1, \dots, \mathbf{v}_N]_{N \times N}$ that satisfy:

$$\begin{aligned} \mathbf{U}\mathbf{U}^* &= \mathbf{U}^*\mathbf{U} = \mathbf{I}_{M \times M} \\ \mathbf{V}\mathbf{V}^* &= \mathbf{V}^*\mathbf{V} = \mathbf{I}_{N \times N} \end{aligned} \quad (9.86)$$

Both $\mathbf{A}\mathbf{A}^*$ and $\mathbf{A}^*\mathbf{A}$ can be diagonalized by \mathbf{U} and \mathbf{V} respectively:

$$\begin{aligned} \mathbf{U}^*(\mathbf{A}\mathbf{A}^*)\mathbf{U} &= \mathbf{\Lambda}_{M \times M} = \text{diag}[\lambda_1, \dots, \lambda_R, 0, \dots, 0] \\ \mathbf{V}^*(\mathbf{A}^*\mathbf{A})\mathbf{V} &= \mathbf{\Lambda}_{N \times N} = \text{diag}[\lambda_1, \dots, \lambda_R, 0, \dots, 0] \end{aligned} \quad (9.87)$$

The *singular value decomposition theorem* states that the $M \times N$ matrix \mathbf{A} can be diagonalized by \mathbf{U} and \mathbf{V} :

$$\mathbf{U}^*\mathbf{A}\mathbf{V} = \mathbf{\Lambda}^{1/2} = \text{diag}[\sqrt{\lambda_1}, \dots, \sqrt{\lambda_R}, 0, \dots, 0] = \text{diag}[s_1, \dots, s_R, 0, \dots, 0] \quad (9.88)$$

where $\mathbf{\Lambda}$ is an $M \times N$ matrix with R non-zero elements $s_n = \sqrt{\lambda_n}$, ($n = 1, \dots, R$), called *singular values* of \mathbf{A} along the diagonal (starting with the top-left element of the matrix), and the column vectors \mathbf{u}_k and \mathbf{v}_k in \mathbf{U} and \mathbf{V} are called respectively the *left-singular vectors* and *right-singular vectors* corresponding to singular value s_k . This equation can be considered as the forward SVD transform. Pre-multiplying \mathbf{U} and post-multiplying \mathbf{V}^* on both sides of the

equation above, we get inverse transform:

$$\mathbf{A} = \mathbf{U}\Lambda^{1/2}\mathbf{V}^* = \sum_{k=1}^R \sqrt{\lambda_k} [\mathbf{u}_k \mathbf{v}_k^*] = \sum_{k=1}^R s_k [\mathbf{u}_k \mathbf{v}_k^*] \quad (9.89)$$

by which the original matrix \mathbf{A} is represented as a linear combination of R matrices $[\mathbf{u}_k \mathbf{v}_k^*]$ weighted by the singular values $\sqrt{\lambda_k}$ ($k = 1, \dots, R$). We can rewrite both the forward and inverse SVD transform as a pair of forward and inverse transforms:

$$\begin{cases} \Lambda^{1/2} = \mathbf{U}^* \mathbf{A} \mathbf{V} \\ \mathbf{A} = \mathbf{U} \Lambda^{1/2} \mathbf{V}^* \end{cases} \quad (9.90)$$

The matrix \mathbf{A} can be considered as a 2-D signal, such as an image, which can be forward SVD transformed to be represented by a set of coefficients $s_k = \sqrt{\lambda_k}$ for the components $[\mathbf{u}_k \mathbf{v}_k^*]$ ($k = 1, \dots, R$), which can also be called eigen-images, and 2-D signal \mathbf{A} can also be expressed by the inverse SVD transform as a linear combination of R SVD components weighted by the singular values.

Same as all orthogonal transforms discussed previously, the SVD transform also conserves the signal energy. The total energy contained in \mathbf{A} is simply the sum of the energy contained in each of its $M \times N$ elements, which is equal to the trace of either $\mathbf{A}\mathbf{A}^*$ and $\mathbf{A}^*\mathbf{A}$:

$$\mathcal{E} = \sum_{m=1}^M \sum_{n=1}^N |a_{mn}|^2 = \text{tr}(\mathbf{A}\mathbf{A}^*) = \text{tr}(\mathbf{A}^*\mathbf{A}) \quad (9.91)$$

Moreover, as trace is conserved by an orthogonal transform, we take trace on both sides of Eq.9.87 to get:

$$\begin{aligned} \text{tr}[\mathbf{U}^*(\mathbf{A}\mathbf{A}^*)\mathbf{U}] &= \text{tr}(\mathbf{A}\mathbf{A}^*) = \text{tr}\Lambda = \sum_{k=1}^R \lambda_k \\ \text{tr}[\mathbf{V}^*(\mathbf{A}^*\mathbf{A})\mathbf{V}] &= \text{tr}(\mathbf{A}^*\mathbf{A}) = \text{tr}\Lambda = \sum_{k=1}^R \lambda_k \end{aligned} \quad (9.92)$$

This result indicates that the energy contained in the signal \mathbf{A} is the same as the sum of all singular value squared representing the signal energy in transform domain after the SVD transform.

We can further show that the *degrees of freedom (DOF)*, the number of independent variables in the representation of the signal, is also conserved by the SVD transform, indicating that the signal information is conserved. For simplicity, we assume $M = N = R$ and the DOF of $\mathbf{A}_{N \times N}$ is N^2 . After the transform, the signal is represented in terms of \mathbf{U} , \mathbf{V} , and Λ . We first show the DOF of both \mathbf{U} and \mathbf{V} is $(N^2 - N)/2$. The DOF of the first column with N elements is $N - 1$ due to the constraint of normalization, and the DOF of the second column is $N - 2$ due to the constraints of being orthogonal to the first one as well as being normalized. In general, the DOF of a column is always one less than that

of the previous one, and the total DOF of all N vectors of \mathbf{U} is:

$$(N - 1) + (N - 2) + \cdots + 1 = N(N - 1)/2 = (N^2 - N)/2 \quad (9.93)$$

The same is true for \mathbf{V} . Together with the DOF of N for Λ , the total DOF in the transform domain is $2(N^2 - N)/2 + N = N^2$, same as that of \mathbf{A} before the SVD transform.

SVD can be used to find the pseudo-inverse of matrix $\mathbf{A} = \mathbf{U}\Lambda^{1/2}\mathbf{V}^*$:

$$\mathbf{A}^- = \mathbf{V}\Lambda^{-1/2}\mathbf{U}^* \quad (9.94)$$

where both \mathbf{A}^- and $\Lambda^{-1/2}$ are $N \times M$ matrices, and $\Lambda^{-1/2}$ is the pseudo-inverse of Λ composed of the reciprocals $1/s_k = 1/\sqrt{\lambda_k}$ of the R singular values along the diagonal.

9.4.2 Application in Image Compression

The SVD transform has various applications including image processing and analysis. We now consider how it can be used for data compression. For simplicity we consider an N by N real image matrix $\mathbf{A}_{N \times N} = [\mathbf{a}_1, \dots, \mathbf{a}_N]$ where \mathbf{a}_k is the k th column vector of \mathbf{A} . Image compression can be achieved by using only the first $K < N$ eigen-images of \mathbf{A} :

$$\mathbf{A}_K = \sum_{k=1}^K \sqrt{\lambda_k} \mathbf{u}_k \mathbf{v}_k^T \quad (9.95)$$

The energy contained in \mathbf{A}_K is:

$$\begin{aligned} \text{tr}[\mathbf{A}_K^T \mathbf{A}_K] &= \text{tr}\left[\sum_{k=1}^K \sqrt{\lambda_k} \mathbf{v}_k \mathbf{u}_k^T\right] \left[\sum_{l=1}^K \sqrt{\lambda_l} \mathbf{u}_l \mathbf{v}_l^T\right] \\ &= \text{tr}\left[\sum_{k=1}^K \left(\sum_{l=1}^K \sqrt{\lambda_k} \sqrt{\lambda_l} \mathbf{v}_k \mathbf{u}_k^T \mathbf{u}_l \mathbf{v}_l^T\right)\right] = \text{tr}\left[\sum_{k=1}^K \lambda_k \mathbf{v}_k \mathbf{v}_k^T\right] \\ &= \sum_{k=1}^K \lambda_k \text{tr}[\mathbf{v}_k \mathbf{v}_k^T] = \sum_{k=1}^K \lambda_k \mathbf{v}_k^T \mathbf{v}_k = \sum_{k=1}^K \lambda_k \end{aligned}$$

The percentage of energy contained in the compressed image \mathbf{A}_K is $\sum_{k=1}^K \lambda_k / \sum_{k=1}^R \lambda_k$. Obviously if we use the K components corresponding to the K largest eigenvalues, the energy contained in \mathbf{A}_K is maximized.

Next we consider the compression rate in terms of the DOF of \mathbf{A}_K . The DOF in the K orthogonal vectors $\{\mathbf{u}_i \ i = 1, \dots, K\}$ is:

$$(N - 1) + (N - 2) + \cdots + (N - K) = NK - K(K + 1)/2 \quad (9.96)$$

The same is true for $\{\mathbf{v}_i \ i = 1, \dots, K\}$. Including the DOF of K in $\{\lambda_k, k = 1, \dots, K\}$, we get the total DOF:

$$2NK - K(K + 1) + k = 2NK - K^2 \quad (9.97)$$

and the compression ratio is

$$\frac{2NK - K^2}{N^2} = \frac{2K}{N} - \frac{K^2}{N^2} \approx \frac{2K}{N} \quad (9.98)$$

We consider a specific example of the image of Lenna ($M = N = R = 128$) shown in Fig.9.22 (left) together with its SVD matrices \mathbf{U} and \mathbf{V} (middle and right). The singular values $s_i = \sqrt{\lambda_i}$ in descending order and the energy λ_i contained are also plotted respectively in the top and bottom panels of Fig.9.23. The reconstructed images using different K of the SVD eigen-images are shown in Fig.9.24. The top two rows show the SVD eigen-images (1st row) corresponding to 10 largest singular values, and the corresponding partial sums (2nd row) for the reconstruction. The bottom two rows show the rest of the eigen-images and the corresponding reconstructions, when the number K is increased by 10 each time ($K = 10, 20, 30, \dots$).

We see that the reconstructed images approximate the original image progressively closely as K is increased to include more eigen-images in the partial sum. This effect can be quantitatively explained by the energy distribution over the total 128 SVD components, shown in the lower panel of Fig.9.23. The distribution curve is obtained by simply squaring the singular value curve in the top panel so that it represents the energy contained in each of the eigen-images. As most of the signal energy is contained in the first few SVD components, all eigen-images for $K > 20$ in the 3rd row contain little information, correspondingly, the reconstructed images in the 4th row closely approximate the original image, which is perfectly reconstructed only if all $M = N = 128$ eigen-images are used.



Figure 9.22 Original image (left), matrices \mathbf{U} (middle) and \mathbf{V} (right)

9.5 Homework Problems

1. An experiment concerning two random variables x and y is carried out $K = 3$ times with different outcomes as listed in the tables given below. Calculate their correlation r_{xy} based on the estimated means and covariances:

$$\hat{\mu}_x = \frac{1}{K} \sum_{k=1}^K x^{(k)}, \quad \hat{\mu}_y = \frac{1}{K} \sum_{k=1}^K y^{(k)} \quad (9.99)$$

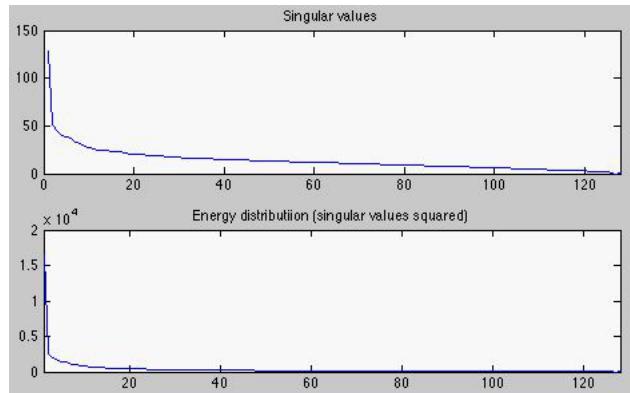


Figure 9.23 Singular values $s_i = \sqrt{\lambda_i}$ (top) and their energy distribution λ_i (bottom)



Figure 9.24 SVD components (top) and the corresponding partial reconstructions (bottom)

$$\hat{\sigma}_{xy}^2 = \frac{1}{K-1} \sum_{k=1}^K x^{(k)} y^{(k)} - \hat{\mu}_x \hat{\mu}_y, \quad \hat{r}_{xy} = \frac{\hat{\sigma}_{xy}^2}{\sqrt{\hat{\sigma}_x^2 \hat{\sigma}_y^2}} \quad (9.100)$$

a.

k	1st	2nd	3rd
$x^{(k)}$	1	2	3
$y^{(k)}$	1	2	3

(9.101)

b.

k	1st	2nd	3rd
$x^{(k)}$	2	4	6
$y^{(k)}$	3	6	9

(9.102)

c.

k	1st	2nd	3rd
$x^{(k)}$	1	2	3
$y^{(k)}$	3	2	1

(9.103)

d.

k	1st	2nd	3rd
$x^{(k)}$	1	2	3
$y^{(k)}$	2	2	2

(9.104)

e.

k	1st	2nd	3rd
$x^{(k)}$	2	2	2
$y^{(k)}$	1	2	3

(9.105)

f.

k	1st	2nd	3rd	4th	5th
$x^{(k)}$	1	2	2	2	3
$y^{(k)}$	2	1	2	3	2

(9.106)

2. In the 2-D normal distribution in Eq.9.45, let $a = c = 5$ and $b = 8$.
- Find the two eigenvalues λ_0 and λ_1 and their corresponding eigenvectors ϕ_0 and ϕ_1 .
 - Find the KLT matrix $\Phi = [\phi_0 \phi_1]$. What kind of rotation does it represent? Carry out the KLT rotation $\mathbf{y} = \Phi^T \mathbf{x}$ so that $\mathbf{y} = [y[0], y[1]]^T$ can be expressed in terms of $\mathbf{x} = [x[0], x[1]]^T$. Find Σ_y .
 - Give the quadratic equation associated with a 2-D normal distribution of \mathbf{y} after the KLT. Confirm this is an equation of an ellipse and find the major and minor semi-axes.
3. Consider a set of $K = 9$ data points in an $N = 2$ dimensional space:

$$\mathbf{x} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 2 & 3 & 3 & 4 \\ 1 & 2 & 3 & 4 & 5 & 3 & 2 & 4 & 3 \end{bmatrix} \quad (9.107)$$

Do the following by hand or using Matlab (or any other computer tools).

- Plot the data points on a 2-D plane to visualize the data.
- Find the mean vector and covariance matrix of these $K = 9$ data points;
- Find Σ_x 's eigenvalues λ_i and corresponding normalized eigenvectors ϕ_i ($i = 0, 1$), form an orthogonal KLT matrix $\Phi = [\phi_0 \phi_1]$ by the two eigenvectors;
- Carry out KLT of the original data $\mathbf{X} = \Phi^T \mathbf{x}$;
- Find the mean vector and covariance matrix of \mathbf{X} in KLT transform domain;
- Verify that the total signal energy (trace of covariance matrix) is conserved. If one of the two dimensions of \mathbf{X} corresponding the smaller eigenvalue is dropped, what is the percentage of energy remaining?

- g. Re-plot the $K = 9$ data points \mathbf{X} in the KLT transform domain spanned by ϕ_0 and ϕ_1 .
4. Repeat the problem above with the same data set augmented with four additional points

$$\mathbf{x} = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 2 & 3 & 3 & 4 & 2 & 4 & 1 & 5 \\ 1 & 2 & 3 & 4 & 5 & 3 & 2 & 4 & 3 & 4 & 2 & 5 & 1 \end{bmatrix} \quad (9.108)$$

5. Carry out SVD in Matlab (or any other programming language) of the following matrix:

$$\mathbf{A} = \begin{bmatrix} 2 & 1 & 2 & 4 \\ 4 & 3 & 3 & 1 \end{bmatrix} \quad (9.109)$$

- a. Find \mathbf{U} , \mathbf{V} and the singular values;
- b. Verify that $\mathbf{A} = \mathbf{U}\Sigma^{1/2}\mathbf{V}^T$
- c. Find the pseudo-inverse $\mathbf{A}^- = \mathbf{V}\Sigma^{-1/2}\mathbf{U}^T$
- d. Verify that $\mathbf{A}\mathbf{A}^- = \mathbf{I}$.
6. Develop code in Matlab or any other programming language to implement the following:
- a. Use Matlab function “rand” or “normrnd” to generate a set of $K = 1000$ samples of an $N = 8$ dimensional random vector \mathbf{x} . Find the mean vector \mathbf{m}_x and covariance matrix Σ_x . Observe the diagonal and off-diagonal elements of the covariance matrix and explain what you have observed. Justify that Σ_x can be modeled by an identity matrix $c\mathbf{I}$ with some constant c .
- b. Generate the N by N covariance matrix \mathbf{R} of a first order Markov process as given in Eq.9.12 with some r such as $r = 0.9$, and then design an N by N transform matrix \mathbf{A} (not orthogonal) so that the signal after the transform $\mathbf{y} = \mathbf{Ax}$ becomes Markov in the sense that its covariance given below can also be modeled by Eq.9.12:

$$\Sigma_y = E(\mathbf{yy}^T) = E(\mathbf{Axx}^T\mathbf{A}^T) = \mathbf{A}E(\mathbf{xx}^T)\mathbf{A}^T = \mathbf{A}\Sigma_x\mathbf{A}^T \quad (9.110)$$

Hint: Assume $\Sigma_x = \mathbf{I}$ and consider using SVD as given in Eq.9.90.

- c. Carry out transform $\mathbf{y} = \mathbf{Ax}$ and verify that Σ_y is indeed Toeplitz-like. Then carry out both KLT and DCT to \mathbf{y} :

$$\mathbf{z}_{KLT} = \Phi^T \mathbf{y}, \quad \text{and} \quad \mathbf{z}_{DCT} = \mathbf{C}^T \mathbf{y} \quad (9.111)$$

- d. Compare the 3-D plots of the covariance matrices of both \mathbf{z}_{KLT} and \mathbf{z}_{DCT} to convince yourself that they are very similar to each other. Plot each of the N columns of Φ and those of \mathbf{C} to convince yourself that they also look very similar to each other.
- e. The steps above can be repeated for larger values of N .
7. A signal composed of N symbols (e.g., values of signal samples before or after an orthogonal transform) can be encoded by the optimal Huffman coding with minimum total code length, which can be carried out in the following algorithm:

- Estimate the probability p_k of the k th outcome ($k = 0, \dots, N - 1$) and sort them in descending order. Here we assume $N = 2^n$ for convenience. Set $M = N$.
- The forward path (left to right): Replace the two lowest probabilities by their sum. Set $M = M - 1$. Resort the M probabilities. Repeat this step until $M = 2$.
- Backward path (right to left): Add a bit (0 or 1) to the binary code of the two probabilities newly emerging. Set $M = M + 1$. Repeat this step until $M = N$.

For example: consider $N = 2^2 = 4$ symbols A, B, C and D with probabilities $p_A = 0.4$, $p_B = 0.3$, $p_C = 0.2$, and $p_D = 0.1$:

	p_k	code	p_k	code	p_k	code
A	0.4	1	0.4	1	0.6	0
B	0.3	00	0.3	00	0.4	1
C	0.2	010	0.3	01		
D	0.1	011				

The average code length (number of bits) is:

$$L = 0.4 \times 1 + 0.3 \times 2 + 0.2 \times 3 + 0.1 \times 3 = 1.9 \text{ bits}$$

and the uncertainty is

$$H = -0.4 \log_2 0.4 - 0.3 \log_2 0.3 - 0.2 \log_2 0.2 - 0.1 \log_2 0.1 = 1.846 \text{ bits}$$

Now carry out the Huffman encoding to each of the following cases and compare the average code length with the uncertainty.

- $p_A = 0.5$, $p_B = 0.5$, $p_C = 0.0$, and $p_D = 0.0$
 - $p_A = 0.9$, $p_B = 0.1$, $p_C = 0.0$, and $p_D = 0.0$
 - $p_A = 0.8$, $p_B = 0.1$, $p_C = 0.06$, and $p_D = 0.04$
 - $p_A = 0.25$, $p_B = 0.25$, $p_C = 0.25$, and $p_D = 0.25$
- The KLT is widely used in *pattern classification* by which each object of interest is treated as a *pattern* described by an N-D vector $\mathbf{x} = [x_1, \dots, x_N]^T$, a point in an N-D *feature space*, which is then classified to one of C different categories or classes ω_c ($c = 1, \dots, C$). Here each element x_n in pattern \mathbf{x} represents one of a set of N *features*. For example, the height and weight can be treated as $N = 2$ features of human body, which can be classified into one of $C = 2$ gender classes.
 - Based on the assumed average weight and height of 65 Kg and 162 cm respectively for a group of K_f female students, and 80 Kg and 175 cm for a group of K_m male students on a co-ed campus, generate two sets of normally distributed 2-D data points $\mathbf{x} = [x_1, x_2]^T$ (two components are for weight and height, respectively) with standard deviation of 8 for

both features and both genders. (Hint: you can use Matlab function normrnd(mu,sigma) to generate a set of random numbers of normal distribution with mean mu and variance sigma.)

- b. Feature selection is one way to reduce the dimensionality of the feature space from N to $M < N$. For example, the $N=2$ dimensional pattern vectors (for height and weight) can be converted to a 1-D space, by one of the following methods:
 - * Use 1st feature x_1 for weight only (drop x_2 for height).
 - * Use 2nd feature x_2 for height only (drop x_1 for weight).
 - * Generate a new feature $y = ax_1 + bx_2$ as a linear combination of both x_1 and x_2 .

Obviously the third approach is potentially better as both weight and height information will be used. Implement this approach to generate a 1-D feature with maximal separability between the two classes and carry it out to the simulated data generated above. (Hint consider the KLT of the dataset.)

- c. In the feature space, different classification algorithm can be carried out, such as the following:
 - * Minimum distance: a pattern \mathbf{x} is classified to class ω_i if

$$\|\mathbf{x} - \mathbf{m}_i\|^2 \leq \|\mathbf{x} - \mathbf{m}_j\|^2, \quad \text{for all } j = 1, \dots, C \quad (9.112)$$

- * Bayes method: a pattern \mathbf{x} is classified to class ω_i if it is most likely to belong to the class, i.e.,

$$P(\omega_i/\mathbf{x}) \geq P(\omega_j/\mathbf{x}), \quad \text{for all } j = 1, \dots, C \quad (9.113)$$

where the likelihood is defined below according to Bayes formula:

$$P(\omega_c/\mathbf{x}) = \frac{p(\mathbf{x}/\omega_c)P(\omega_c)}{p(\mathbf{x})} \propto p(\mathbf{x}/\omega_c)P(\omega_c) \quad (9.114)$$

and $P(\omega_c)$ is the *a priori* probability for any randomly chosen pattern to belong to class ω_c , and $P(\omega_c/\mathbf{x})$ is the *posteriori* probability for a specific pattern \mathbf{x} to belong to the class. The denominator $p(\mathbf{x})$ is a distribution of all patterns independent of their classes which can be dropped as it is the same for all classes.

Specifically in this problem, we have $P(\omega_f) = K_f/(K_f + K_m)$, $P(\omega_m) = K_m/(K_f + K_m)$, and $p(\mathbf{x}/\omega_f)$ and $p(\mathbf{x}/\omega_m)$ can be assumed to be normal and their means and variances can be estimated respectively by the K_f patterns $\mathbf{x} \in \omega_f$ and the K_m patterns $\mathbf{x} \in \omega_m$ of known class, called *training samples* in practice.

Apply either minimum distance or Bayes classification to each of the three 1-D feature spaces specified in previous section, compare the three classification error rates to find the best 1-D feature space.

- 9. Based on the provided $N = 20$ (filename “DC0” through “DC19”) out of the 210 bands of the HYDICE image data (Lincoln Memorial, Washington DC),

carry out the supervised classification based on the spectral signatures of a set of $K = 4$ typical ground cover material types in the region: water surface, lawn areas, trees, and building roofs, in the following two steps:

- Training:** For each of the K classes of interest, e.g., water surface, pick a set of pixels in the image known to belong to the class (called *training samples*). Find the mean vector \mathbf{m}_k ($k = 1, \dots, K$) of the training samples for each class. You could use the following areas for the four training classes:

Ground Types	Area 1		Area 2	
	rows	Columns	rows	Columns
Water	230-390	10-40	360-400	10-50
Grass	400-420	150-170	390-410	290-300
Trees	200-230	80-110	240-270	85-105
Roofs	512-517	13-49	700-710	207-233

- Classification:** For each pixel \mathbf{x} in the image, find all K N-D Euclidean distances $D(\mathbf{x}, \mathbf{m}_l)$ ($l = 1, \dots, K$), and classify the pixel to the k th class if $D(\mathbf{x}, \mathbf{m}_k)$ is minimum among all K distances.

Next, use the KLT-based feature extraction method discussed in Subsection 9.3.2 to generate a set of M new features that conserve 99% of the information (now in terms of the separability between the four specific classes of interest), and then carry out the supervised classification in this M-D feature space. Finally, compare the classification results of the two parts.

10. Ten handwritten digits from 0 to 9 are provided in an image DigitsClasse.gif, which is composed of 10 sets of 225 subimages for each of the 10 digits. As each digit is represented as a 16×16 image, we can consider these patterns as vectors in an $N = 256$ feature space. Now carry out the KLT-based feature extraction as discussed in Example 9.4 to obtain an $M = 9$ dimensional feature space. Then carry out the classification of these $225 \times 10 = 2250$ patterns using minimum distance method used in previous problem.

10 Continuous and Discrete-time Wavelet Transforms

10.1 Why Wavelet?

10.1.1 Short-time Fourier Transform and Gabor Transform

In Chapters 3, we learned that a signal can be represented as either a time function $x(t)$ as the amplitude of the signal at any given moment t , or, alternatively and equivalently, a spectrum $X(f) = \mathcal{F}[x(t)]$ representing the magnitude and phase of the frequency component at any given frequency f . However, no information in terms of the frequency contents is explicitly available in time domain, and no information in terms of the temporal characteristics of the signal is explicitly available in frequency domain. In this sense, neither $x(t)$ in time domain nor $X(f)$ in frequency domain provides complete description of the signal. In other words, we can have either temporal or spectral locality regarding the information contained in the signal, but never both at the same time.

To address this dilemma, the *short-time Fourier transform (STFT)*, also called *windowed Fourier transform*, can be used. The signal $x(t)$ to be analyzed is first truncated by a window function $w(t)$ before it is Fourier transformed to the frequency domain. As all frequency components in the spectrum are known to be contained in the signal segment inside this particular time window, certain temporal locality in frequency domain is achieved.

We first consider a simple rectangular window with width T :

$$w_r(t) = \begin{cases} 1 & 0 < t < T \\ 0 & \text{otherwise} \end{cases} \quad (10.1)$$

If a particular segment $\tau < t < \tau + T$ of the signal $x(t)$ is of interest, the signal is first multiplied by the window $w_r(t)$ shifted by τ , and then Fourier transformed to get:

$$X_r(f, \tau) = \mathcal{F}[x(t)w_r(t - \tau)] = \int_{-\infty}^{\infty} x(t)w_r(t - \tau)e^{-j2\pi ft} dt = \int_{\tau}^{\tau+T} x(t)e^{-j2\pi ft} dt \quad (10.2)$$

Based on the time-shift and frequency convolution properties of the Fourier transform, the spectrum of this windowed signal can also be expressed as:

$$X_r(f, \tau) = X(f) * [W_r(f)e^{-j2\pi f\tau}] \quad (10.3)$$

where $W_r(f) = \mathcal{F}[w_r(t)]$ is the Fourier transform of the rectangular window $w_r(t)$. We see that the temporal locality in frequency domain is gained at the expenses of the severe distortion of the STFT spectrum $X_r(f)$ due to the convolution with the ringing sinc function $W_r(f) = \mathcal{F}[w_r(t)]$ of the rectangular window. This distortion could be reduced if a smooth window such as a bell-shaped Gaussian function is used:

$$w_g(t) = e^{-\pi(t/\sigma)^2} \quad (10.4)$$

where the parameter σ controls the width of the window. The spectrum of the Gaussian window is also a Gaussian function (Eq. 3.161):

$$W_g(f) = \mathcal{F}[w_g(t)] = \sigma e^{-\pi(\sigma f)^2} \quad (10.5)$$

Now the spectrum of the signal windowed by a Gaussian (shifted by τ) is:

$$X_g(f, \tau) = \mathcal{F}[x(t)w_g(t - \tau)] = \int_{-\infty}^{\infty} x(t)e^{-(t-\tau)^2/\sigma^2} e^{-j2\pi ft} dt \quad (10.6)$$

This Fourier transform of the Gaussian windowed signal is called the *Gabor transform* of the signal. The original time signal can be obtained by the inverse Gabor transform. Multiplying $e^{j2\pi f\tau}$ on both sides of the equation, and then integrating with respect to f , we get:

$$\begin{aligned} \int_{-\infty}^{\infty} X_g(f, \tau) e^{j2\pi f\tau} df &= \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} x(t)e^{-(t-\tau)^2/\sigma^2} e^{-j2\pi ft} dt \right] e^{j2\pi f\tau} df \\ &= \int_{-\infty}^{\infty} x(t)e^{-(t-\tau)^2/\sigma^2} \left[\int_{-\infty}^{\infty} e^{-j2\pi ft} e^{j2\pi f\tau} df \right] dt = \int_{-\infty}^{\infty} x(t)e^{-(t-\tau)^2/\sigma^2} \delta(t - \tau) dt \\ &= x(\tau) \end{aligned} \quad (10.7)$$

Similar to the case of rectangular windowing in Eq.10.3, the Gabor spectrum in Eq.10.6 can also be written as:

$$X_g(f, \tau) = [W_g(f)e^{-j2\pi f\tau}] * X(f) \quad (10.8)$$

As before, the Gabor spectrum $X_g(f, \tau)$ in Eq.10.8 is a blurred version of the true Fourier spectrum $X(f)$, although the ringing artifact caused by the rectangular window is avoided.

10.1.2 The Heisenberg Uncertainty

In general the STFT method, based on either rectangular or Gaussian windowing, suffers from a profound difficulty, namely, the increased time locality results necessarily in a decreased frequency locality, as the resolution of the STFT spectrum, a blurred version of the true Fourier spectrum $X(f)$, is much reduced due to the convolution in Eqs.10.3 or 10.8. For example, in the case of the Gabor transform, as the width $1/\sigma$ of $W_g(f)$ in frequency domain is inversely proportional to the width σ of $w_g(t)$ in time domain, a narrower time window $w_g(t)$

for higher temporal resolution will necessarily cause a wider $W_g(f)$ and thereby a more blurred Gabor spectrum $X_g(f)$.

This issue could also be illustrated if we further assume the truncated signal repeats itself outside a finite window of width T , i.e., the signal $x(t+T) = x(t)$ becomes periodic. Correspondingly, its spectrum becomes discrete, composed of an infinite set of coefficients $X[k]$ each for one of the frequency components $e^{j2\pi kt/T}$ ($k = 0, \pm 1, \pm 2, \dots$). Obviously this discrete spectrum contains no information in the gap of $f_0 = 1/T$ between any two consecutive components $X[k]$ and $X[k+1]$. Moreover, the higher temporal resolution we achieve by reducing T , the lower frequency resolution will result due to the larger gap $f_0 = 1/T$ in frequency domain. We see that it is fundamentally impossible to have the complete information of a given signal in both time and frequency domains at the same time, as increasing the resolution in one domain will necessarily reduce that in the other, due to the *Heisenberg uncertainty* discussed in Chapter3 (Eq.3.175).

The STFT approach also has another drawback. The window width is fixed through out the analysis, even though there may be a variety of different signal characteristics of interest with varying time scales. For example, the signal may contain some random, irregular and sparse spikes, or bursts of rapid oscillation, which can be localized only if a very narrow time window is used. On the other hand, there may be some totally different features in the signal, such as slow changing drifts and trends, which can be captured only if the time window has much wider width. It would be impossible for the STFT method with a fixed window width to detect and represent all of these different types of signal characteristics of interest.

In summary, if the signal is stationary and its characteristics of interest do not change much over time, then the Fourier transform may be sufficient for the analysis of the signal in terms of characterizing these features in frequency domain. However, in many applications it is the transitory or non-stationary aspects of the signal such as drifts, trends, and abrupt changes that are of most concern and interest, but the Fourier analysis is unable to detect and characterize such features in frequency domain.

In order to overcome these limitations of the Fourier analysis and to gain localized information in both frequency and time domains, a different kind of transform, called the *wavelet transform*, can be used. This method can be viewed as a trade-off between time and frequency domains. Unlike the Fourier transform which converts a signal between time (or space) and frequency domains, the coefficients of the wavelet transform represent signal details of different scales (corresponding to different frequencies in the Fourier analysis), and also their temporal (or spatial) locations. Information contained in different scale levels reflects the signal characteristics of different scales.

The discussion above can be summarized by the *Heisenberg Box (or Heisenberg cell)* illustrated in Fig.10.1, which illustrates the issue of resolution and locality in

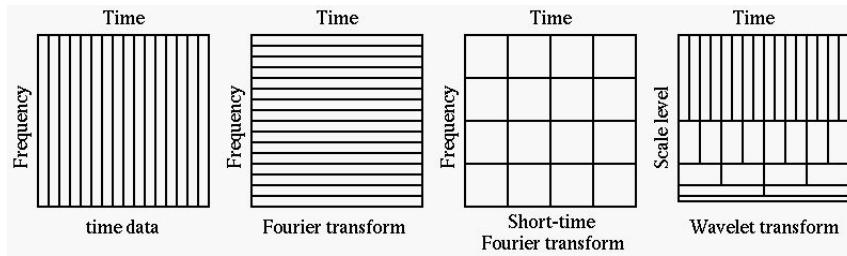


Figure 10.1 Heisenberg box: comparisons of temporal and frequency locality in Fourier and wavelet transforms

both time and frequency in the Fourier transform, short-time Fourier transform and wavelet transform.

The panel on the left is the time signal with full time resolution (temporal locality) but zero frequency resolution (frequency locality). The second panel represents its Fourier spectrum with full frequency resolution but zero temporal resolution. The third panel is the short-time Fourier transform whose temporal and frequency localities are inversely proportional to each other. In fact, this STFT method can be considered as a trade-off between the first two cases depending on the fixed window width. The last panel on the right represents the wavelet transform with varying scale levels and the corresponding time resolutions. At a low scale level (less detailed information corresponding to low frequencies) the window size is large, while at a high scale level (more signal details corresponding to high frequencies) the window size is small. In other words, local information in both time and frequency domains can be represented in this transform scheme.

10.2 Continuous-Time Wavelet Transform (CTWT)

10.2.1 Mother and Daughter Wavelets

All continuous orthogonal transforms previously discussed, such as the Fourier transform, are integral transforms that can be expressed as an inner product of the signal $x(t)$ and a transform kernel function $\phi_f(t)$:

$$X(f) = \langle x(t), \phi_f(t) \rangle = \int x(t) \overline{\phi}_f(t) dt \quad (10.9)$$

Here the family of the kernel functions $\phi_f(t)$ corresponding to different f form an orthogonal basis that span the vector space in which the signal $x(t)$ resides. For example, in the case of the Fourier transform, a member of the kernel function family is a complex exponential $\phi_f(t) = e^{j2\pi ft}$ corresponding to a parameter f representing a specific frequency.

Similarly, the *continuous-time wavelet transform (CTWT)* is also an integral transform based on a set of kernel functions, sometimes referred to as the *daughter*

ter wavelets, all derived from a *mother wavelet* $\psi(t)$ that should satisfy the following conditions:

- $\psi(t)$ has a compact support, i.e., $\psi(t) \neq 0$ only inside a bounded range $a < t < b$.
- $\psi(t)$ has a zero mean:

$$\int_{-\infty}^{\infty} \psi(t) dt = 0, \quad \text{i.e.} \quad \Psi(f)|_{f=0} = \Psi(0) = 0 \quad (10.10)$$

where $\Psi(f) = \mathcal{F}[\psi(t)]$ is the Fourier spectrum of $\psi(t)$. In other words, the DC component of the mother wavelet is zero.

- $\psi(t) \in \mathcal{L}^2$ is square integrable, i.e.,

$$\int_{-\infty}^{\infty} |\psi(t)|^2 dt < \infty \quad (10.11)$$

- $\psi(t)$ can be normalized so that:

$$\|\psi(t)\|^2 = \int_{-\infty}^{\infty} |\psi(t)|^2 dt = 1 \quad (10.12)$$

Intuitively, a mother wavelet $\psi(t)$ needs to satisfy two conditions. First, it is non-zero only within a finite range (first condition), i.e., it is “small”. Second, it has a zero mean (second condition), i.e., it is a “wave” that takes both positive and negative values around zero. In other words, $\psi(t)$ is a small wave, therefore the name “wavelet”. Obviously this is essentially different from all other continuous orthogonal transforms such as the Fourier and cosine transforms whose kernel functions are sinusoidal waves (infinite waves) defined over the entire time axis.

Based on the mother wavelet, a family of kernel functions $\psi_{s,\tau}(t)$, the daughter wavelets, can be generated by scaling and translating the mother wavelet by s and τ , respectively:

$$\psi_{s,\tau}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-\tau}{s}\right) \quad (10.13)$$

where τ is the time translation ($\tau > 0$ for right shift and $\tau < 0$ for left shift) and $s > 0$ is a scaling factor ($s > 1$ for expansion and $s < 1$ for compression). Unlike the kernel function $\phi_f(t) = e^{j2\pi f t}$ of the Fourier transform with only one parameter f for frequency, the CTWT kernel $\psi_{s,\tau}(t)$ has two parameters τ and s for translation and scaling, respectively. This is the reason why the wavelet transform is capable of representing localized information in time domain as well as in different scale levels (corresponding to different frequencies), while the Fourier transform is only capable of representing localized frequency information.

The factor $1/\sqrt{s}$ is included in the wavelet $\psi_{s,\tau}(t)$ so that it is also normalized as the mother wavelet, independent of the scaling factor s :

$$\begin{aligned} \|\psi_{s,\tau}(t)\|^2 &= \langle \psi_{s,\tau}(t), \psi_{s,\tau}(t) \rangle = \frac{1}{s} \int_{-\infty}^{\infty} \left| \psi\left(\frac{t-\tau}{s}\right) \right|^2 dt \\ &= \frac{1}{s} \int_{-\infty}^{\infty} |\psi(t')|^2 s dt' = \|\psi(t)\|^2 = 1 \end{aligned} \quad (10.14)$$

Here we have assumed $t' = (t - \tau)/s$ and therefore $dt' = dt/s$.

10.2.2 The Forward and Inverse Wavelet Transforms

Given a mother wavelet $\psi(t)$, we can derive all of her daughter wavelets $\psi_{s,\tau}(t)$ for different s and τ , and then define the continuous-time wavelet transform of a time signal $x(t)$ as an integral transform:¹

$$\begin{aligned} X(s, \tau) = \mathcal{W}[x(t)] &= \langle x(t), \psi_{s,\tau}(t) \rangle = \int_{-\infty}^{\infty} x(t) \overline{\psi}_{s,\tau}(t) dt \\ &= \frac{1}{\sqrt{s}} \int_{-\infty}^{\infty} x(t) \overline{\psi}\left(\frac{t-\tau}{s}\right) dt = x(\tau) \star \psi_{s,0}(\tau) \end{aligned} \quad (10.15)$$

We see that the CTWT of $x(t)$ is actually the correlation of the signal $x(t)$ and the wavelet function $\psi_{s,0}(t) = \psi(t/s)/\sqrt{s}$. If we take the Fourier transform on both sides of the CTWT $X(s, \tau)$ above, while treating τ as the time variable and s as a parameter, we get the Fourier spectrum of the CTWT of $x(t)$ (correlation property of the Fourier transform Eq.3.107):

$$\hat{X}(s, f) = \mathcal{F}[X(s, \tau)] = \mathcal{F}[\mathcal{W}[x(t)]] = X(f) \overline{\Psi}_{s,0}(f) \quad (10.16)$$

where $X(f) = \mathcal{F}[x(t)]$ and $\Psi_{s,0}(f) = \mathcal{F}[\psi_{s,0}(t)]$ are the Fourier spectra of the signal $x(t)$ and the wavelet $\psi_{s,0}(t)$, respectively. Note that here we have to use a hat in addition to a capital letter X to denote the result obtained by applying two different transforms (CTWT followed by CTFT) consecutively to a signal $x(t)$. This will be the only deviation from our convention of representing the transform of a signal $x(t)$ by a capital letter $X(f)$.

Given the Fourier spectrum of the mother wavelet $\Psi(f) = \mathcal{F}[\psi(t)]$, we can find the spectrum of a daughter wavelet $\psi_{s,\tau}(t)$ (time-shift and scaling properties of the Fourier transform Eqs.3.102 and 3.98):

$$\Psi_{s,\tau}(f) = \mathcal{F}[\psi_{s,\tau}(t)] = \mathcal{F}\left[\frac{1}{\sqrt{s}}\psi\left(\frac{t-\tau}{s}\right)\right] = \sqrt{s}\Psi(sf)e^{-j2\pi f\tau} \quad (10.17)$$

¹ In wavelet literatures different notations have been used for the CTWT of a signal $x(t)$, such $CWT_x(s, \tau)$ and $Wx(s, \tau)$. However, here we simply use the capitalized letter $X(s, \tau) = \mathcal{W}[x(t)]$ to represent the CTWT of $x(t)$, in consistence with the convention used for all orthogonal transforms considered in previous chapters, such as $X(f) = \mathcal{F}[x(t)]$ for the Fourier transform of $x(t)$.

In particular when $\tau = 0$, we have:

$$\Psi_{s,0}(f) = \sqrt{s}\Psi(sf)e^{-j2\pi f\tau}|_{\tau=0} = \sqrt{s}\Psi(sf) \quad (10.18)$$

Now we see that the CTWT can also be obtained by taking the inverse Fourier transform of $\hat{X}(s, f)$ in Eq.10.16:

$$X(s, \tau) = \mathcal{F}^{-1}[\hat{X}(s, f)] = \mathcal{F}^{-1}[X(f)\overline{\Psi}_{s,0}(f)] = \sqrt{s} \int_{-\infty}^{\infty} X(f)\overline{\Psi}(sf)e^{j2\pi f\tau}df \quad (10.19)$$

The time signal $x(t)$ can be reconstructed from its CTWT transform $X(s, \tau)$ by the inverse wavelet transform:

$$\begin{aligned} x(t) &= \mathcal{W}^{-1}[X(s, \tau)] = \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty X(s, \tau)\psi_{s,\tau}(t)d\tau \frac{ds}{s^2} \\ &= \frac{1}{C_\psi} \int_0^\infty \frac{1}{\sqrt{s}} \int_{-\infty}^\infty X(s, \tau)\psi\left(\frac{t-\tau}{s}\right)d\tau \frac{ds}{s^2} \end{aligned} \quad (10.20)$$

where C_ψ is defined as

$$C_\psi = \int_0^\infty \frac{|\Psi(s)|^2}{s} ds < \infty \quad (10.21)$$

This inequality, referred to as the *admissibility condition*, is necessary for the inverse CTWT to exist. Note that for this condition to hold, we must have $\Psi(f)|_{f=0} = \Psi(0) = 0$, one of the conditions specified before (Eq.10.10). Consequently, Eq.10.19 will produce the same result for different $X(0)$, as it is always multiplied by $\Psi(0) = 0$. In other words, the CTWT is insensitive to the DC component $X(0)$ of the signal $x(t)$.

Now we prove that the signal $x(t)$ can indeed be reconstructed by the inverse CTWT given in Eq.10.20. We first multiply $\Psi_{s,0}(f)/s^2$ on both sides of Eq.10.16 and integrate with respect to s to get:

$$\int_0^\infty \hat{X}(s, f)\Psi_{s,0}(f)\frac{ds}{s^2} = X(f) \int_0^\infty |\Psi_{s,0}(f)|^2 \frac{ds}{s^2} = X(f) \int_0^\infty \frac{|\Psi(sf)|^2}{s} ds \quad (10.22)$$

The last equal sign is due to Eq.10.18. The integral on the right-hand side can be further written as:

$$\int_0^\infty \frac{|\Psi(sf)|^2}{s} ds = \int_0^\infty \frac{|\Psi(sf)|^2}{sf} d(sf) = \int_0^\infty \frac{|\Psi(s')|^2}{s'} ds' = C_\psi \quad (10.23)$$

where we have assumed $s' = sf$, and the last equal sign is due to the definition of C_ψ in Eq.10.21. Now we can solve Eq.10.22 for $X(f)$ to get:

$$X(f) = \frac{1}{C_\psi} \int_0^\infty \hat{X}(s, f) \Psi_{s,0}(f) \frac{ds}{s^2} \quad (10.24)$$

Taking the inverse Fourier transform on both sides we get the inverse DTWT in Eq.10.20:

$$\begin{aligned} x(t) &= \mathcal{F}^{-1}[X(f)] = \frac{1}{C_\psi} \int_0^\infty \mathcal{F}^{-1}[\hat{X}(s, f) \Psi_{s,0}(f)] \frac{ds}{s^2} \\ &= \frac{1}{C_\psi} \int_0^\infty X(s, t) * \psi_{s,0}(t) \frac{ds}{s^2} \\ &= \frac{1}{C_\psi} \int_0^\infty \frac{1}{\sqrt{s}} \int_{-\infty}^\infty X(s, \tau) \psi\left(\frac{t-\tau}{s}\right) d\tau \frac{ds}{s^2} \end{aligned} \quad (10.25)$$

Here we have used the convolution theorem of the Fourier transform (Eq.3.112).

The result of Eq.10.23 also indicates an interesting fact as a side product. For any given function $f(x)$, in this case $|\Psi(f)|^2$, the integral of its scaled version $f(sx)/s$ over all scale s is a constant independent of x , i.e., a constant function over the entire domain of x . This result has some important significance, as we will see later in the future discussion of the discrete-time wavelet transform.

In summary, both the forward and inverse CTWTs in Eqs.10.15 and 10.20 can be written as the following CTWT pair:

$$\begin{aligned} x(s, \tau) &= \mathcal{W}[x(t)] = \frac{1}{\sqrt{s}} \int_{-\infty}^\infty x(t) \overline{\psi\left(\frac{t-\tau}{s}\right)} dt \\ x(t) &= \mathcal{W}^{-1}[X(s, \tau)] = \frac{1}{C_\psi} \int_0^\infty \frac{1}{\sqrt{s}} \int_{-\infty}^\infty X(s, \tau) \psi\left(\frac{t-\tau}{s}\right) d\tau \frac{ds}{s^2} \end{aligned} \quad (10.26)$$

The forward CTWT in the first equation converts a 1-D signal $x(t)$ into a 2-D function $X(s, \tau)$ of s for scale and τ for translation, while the inverse CTWT in the second equation reconstructs the signal from $X(s, \tau)$. The CTWT transform has some essential differences compared to all previously considered orthogonal transforms such as the Fourier transform. First, the Fourier spectrum $X(f) = \mathcal{F}[x(t)]$ is a 1-D function of frequency f , but the CTWT $X(s, \tau) = \mathcal{W}[x(t)]$ is a 2-D function of two variables s and τ . Second, the CTWT is not an orthogonal transform, as its kernel functions, the daughter wavelets, $\psi_{s,\tau}(t)$ are not orthogonal to each other. Due to such differences, the CTWT representation of a 1-D signal is necessarily redundant. It can be used for signal filtering, as to be seen later, but it is not suitable for data compression.

10.3 Properties of the CTWT

In the discussion below we will always assume $X(s, \tau) = \mathcal{W}[x(t)]$ and $Y(s, \tau) = \mathcal{W}[y(t)]$.

- **Linearity:**

$$\mathcal{W}[ax(t) + by(t)] = a\mathcal{W}[x(t)] + b\mathcal{W}[y(t)] = aX(s, \tau) + bY(s, \tau) \quad (10.27)$$

The wavelet transform of a function $x(t)$ is simply an inner product of the function with a kernel function $\psi_{s,\tau}(t)$ (Eq. 10.15). Therefore due to the lin-

earity of the inner product in the first variable, the wavelet transform is also linear.

- **Time shift:**

$$\mathcal{W}[x(t - t')] = X(s, \tau - t') \quad (10.28)$$

The proof is left for the reader as a homework problem.

- **Time scaling:**

$$\mathcal{W}[x(t/a)] = \sqrt{a}X(s/a, \tau/a) \quad (10.29)$$

The proof is left for the reader as a homework problem.

- **Localization:**

Let the center and width of a mother wavelet $\psi(t)$ be $t = t_0$ and Δt in time domain and those of its spectrum be $\Psi(f)$, f_0 and Δf in frequency domain, respectively. Then the center and width of a scaled and translated daughter wavelet $\psi_{s,\tau}(t) = \psi((t - \tau)/s)/\sqrt{s}$ are:

$$t_{0,s,\tau} = st_0 + \tau, \quad \Delta t_{s,\tau} = s\Delta t \quad (10.30)$$

and, according to the time/frequency scaling property (Eq.3.98), the center and width of its spectrum $\Psi_{s,\tau}(f) = \sqrt{s}\Psi(sf)e^{-j2\pi f\tau}$ (Eq.10.17) are:

$$f_{0,s,\tau} = \frac{1}{s}f_0, \quad \Delta f_{s,\tau} = \frac{1}{s}\Delta f \quad (10.31)$$

We can now make two observations:

- The product of the widths of the wavelet function $\psi_{s,\tau}(t)$ in time domain and its spectrum $\Psi_{s,\tau}(f)$ in frequency domain is constant, independent of s and τ :

$$\Delta t_{s,\tau} \Delta f_{s,\tau} = s\Delta t \frac{1}{s}\Delta f = \Delta t \Delta f \quad (10.32)$$

- The spectrum $\Psi_{s,\tau}(f)$ of the wavelet function can be considered as a band-pass filter with a *quality factor* Q defined as the ratio of its bandwidth and the center frequency (a concept used to describe the behavior of analog filters):

$$Q = \frac{\Delta f_{s,\tau}}{f_{0,s,\tau}} = \frac{\Delta f}{f_0} \quad (10.33)$$

i.e., the quality factor Q of the filter is constant, independent of the scaling factor s .

- **Multiplication theorem:**

Corresponding to the multiplication theorem (Eq.3.96) for the Fourier transform $\langle x(t), y(t) \rangle = \langle X(f), Y(f) \rangle$, where $X(f) = \mathcal{F}[x(t)]$ and $Y(f) = \mathcal{F}[y(t)]$, similar theorem also exists for the wavelet transform. However, as the CTWT $X(s, \tau)$ is a function of two variables s and τ , we first need to

define the inner product of two CTWTS as:

$$\langle X(s, \tau), Y(s, \tau) \rangle = \int_0^\infty \int_{-\infty}^\infty X(s, \tau) \bar{Y}(s, \tau) d\tau \frac{ds}{s^2} \quad (10.34)$$

The multiplication theorem states:

$$\langle x(t), y(t) \rangle = \frac{1}{C_\psi} \langle X(s, \tau), Y(s, \tau) \rangle \quad (10.35)$$

To prove this theorem, we substitute the CTWTS of two functions $x(t)$ and $y(t)$ (Eq.10.19)

$$\begin{aligned} X(s, \tau) &= \mathcal{W}[x(t)] = \sqrt{s} \int_{-\infty}^\infty X(f) \bar{\Psi}(sf) e^{j2\pi f\tau} df \\ Y(s, \tau) &= \mathcal{W}[y(t)] = \sqrt{s} \int_{-\infty}^\infty Y(f) \bar{\Psi}(sf) e^{j2\pi f\tau} df \end{aligned} \quad (10.36)$$

into the inner product defined above and get:

$$\begin{aligned} \langle X(s, \tau), Y(s, \tau) \rangle &= \int_0^\infty \int_{-\infty}^\infty X(s, \tau) \bar{Y}(s, \tau) d\tau \frac{ds}{s^2} \\ &= \int_0^\infty \int_{-\infty}^\infty \left[\int_{-\infty}^\infty X(f) \bar{\Psi}(sf) e^{j2\pi f\tau} df \right] \left[\int_{-\infty}^\infty \bar{Y}(f') \Psi(sf') e^{-j2\pi f'\tau} df' \right] d\tau \frac{ds}{s} \\ &= \int_0^\infty \int_{-\infty}^\infty \int_{-\infty}^\infty \left[X(f) \bar{Y}(f') \bar{\Psi}(sf) \Psi(sf') \int_{-\infty}^\infty e^{j2\pi(f-f')\tau} d\tau \right] df' df \frac{ds}{s} \\ &= \int_0^\infty \int_{-\infty}^\infty \int_{-\infty}^\infty X(f) \bar{Y}(f') \bar{\Psi}(sf) \Psi(sf') \delta(f - f') df' df \frac{ds}{s} \\ &= \int_{-\infty}^\infty X(f) \bar{Y}(f) \left[\int_0^\infty \frac{|\Psi(sf)|^2}{s} ds \right] df = C_\psi \int_{-\infty}^\infty X(f) \bar{Y}(f) df \\ &= C_\psi \langle X(f), Y(f) \rangle = C_\psi \langle x(t), y(t) \rangle \end{aligned} \quad (10.37)$$

where, again, C_ψ is given in Eq.10.21. In particular, when $y(t) = x(t)$, we have:

$$\|x(t)\|^2 = \|X(f)\|^2 = \frac{1}{C_\psi} \langle X(s, \tau), X(s, \tau) \rangle = \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty |X(s, \tau)|^2 d\tau \frac{ds}{s^2} \quad (10.38)$$

This is Parseval's theorem of the CTWT, where $|X(s, \tau)|^2$ is the signal energy distribution in the 2-D wavelet transform domain spanned by s and τ .

- **Non-orthogonality:**

All previously considered orthogonal transforms represent a given signal in terms of a set of orthogonal basis functions or vectors that span the vector space in which the signal resides. For example, in the Fourier transform $X(f) = \mathcal{F}[x(t)] = \langle x(t), \phi_f(t) \rangle$, the basis functions $\phi_f(t) = e^{j2\pi ft}$ (for all f) are orthogonal:

$$\langle \phi_f(t), \phi_{f'}(t) \rangle = \int_{-\infty}^\infty \phi_f(t) \bar{\phi}_{f'}(t) dt = 0, \quad (f \neq f') \quad (10.39)$$

indicating that they are uncorrelated with zero redundancy. In other words, the kernel function $\phi_f(t)$ at every single point f in the transform domain makes its unique contribution to the representation of the time signal in the inverse transform $x(t) = \int X(f)\phi_f(t)df = \int \langle x(t), \phi_f(t) \rangle \phi_f(t)df$.

However, this is no longer the case for the CTWT, which converts a 1-D time signal $x(t)$ to a 2-D function $X(s, \tau) = \mathcal{W}[x(t)] = \langle x(t), \psi_{s,\tau}(t) \rangle$ defined over the half plane $-\infty < \tau < \infty$ and $s > 0$. Redundancy exists in this 2-D transform domain (s, τ) in terms of the information needed for the reconstruction of the time signal $x(t)$. The redundancy between any two points (s, τ) and (s', τ') in the transform domain can be measured by the *reproducing kernel*, defined as the inner product of the two kernel functions (basis functions) $\psi_{s,\tau}(t)$ and $\psi_{s',\tau'}(t)$:

$$K(s, \tau, s', \tau') = \langle \psi_{s,\tau}(t), \psi_{s',\tau'}(t) \rangle = \int_{-\infty}^{\infty} \psi_{s,\tau}(t) \overline{\psi}_{s',\tau'}(t) dt \quad (10.40)$$

Unlike Eq.10.39 for any orthogonal transform, the inner product above is not zero in general. This is a major difference between the non-orthogonal CTWT and all orthogonal transforms. This reproducing kernel can be considered as the correlation between two kernel functions $\psi_{s,\tau}(t)$ and $\psi_{s',\tau'}(t)$, representing the redundancy between them.

Let $X(s, \tau) = \mathcal{W}[x(t)] = \langle x(t), \psi_{s,\tau}(t) \rangle \neq 0$ be the CTWT at point (s, τ) . Then the CTWT at another point (s', τ') is:

$$X(s', \tau') = \langle x(t), \psi_{s',\tau'}(t) \rangle = \int_{-\infty}^{\infty} x(t) \overline{\psi}_{s',\tau'}(t) dt \quad (10.41)$$

Substituting the reconstructed $x(t)$ by the inverse CTWT (Eq.10.20) into this equation, we get:

$$\begin{aligned} X(s', \tau') &= \int_{-\infty}^{\infty} \left[\frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty X(s, \tau) \psi_{s,\tau}(t) d\tau \frac{ds}{s^2} \right] \overline{\psi}_{s',\tau'}(t) dt \\ &= \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty X(s, \tau) \left[\int_{-\infty}^\infty \psi_{s,\tau}(t) \overline{\psi}_{s',\tau'}(t) dt \right] d\tau \frac{ds}{s^2} \\ &= \frac{1}{C_\psi} \int_0^\infty \int_{-\infty}^\infty K(s, \tau, s', \tau') X(s, \tau) d\tau \frac{ds}{s^2} \end{aligned} \quad (10.42)$$

Consider two cases. First, if $K(s, \tau, s', \tau') = 0$ for all points (s, τ) , i.e., $\psi_{s',\tau'}(t)$ at point (s', τ') is not correlated with $\psi_{s,\tau}(t)$ at any other point (s, τ) , then $X(s', \tau') = 0$, i.e., it does not contribute to the representation of the signal in the inverse CTWT (Eq.10.20). Second, if $K(s, \tau, s', \tau') \neq 0$ for some points (s, τ) , then $X(s', \tau') \neq 0$ does contribute to the representation of the signal. However, as it is a linear combination of all other $X(s, \tau) \neq 0$ (weighted by $K(s, \tau, s', \tau')$), its contribution is redundant.

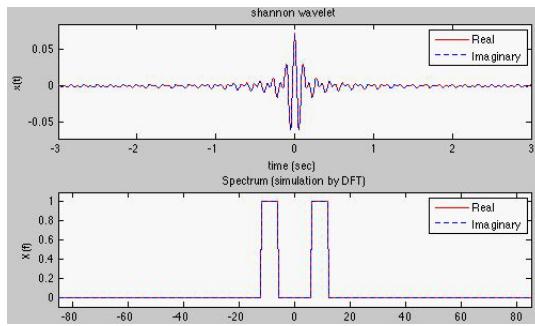


Figure 10.2 Shannon wavelet (top) and its spectrum (bottom)

10.4 Typical Mother Wavelet Functions

Through out the previous discussion of the wavelet transform, the mother wavelet function is not specifically defined. Here we consider some commonly used mother wavelets.

- **Shannon wavelet**

The Shannon wavelet can be more conveniently defined in frequency domain as an ideal band-pass filter:

$$\Psi(f) = \begin{cases} 1 & f_1 < |f| < f_2 \\ 0 & \text{otherwise} \end{cases} \quad (10.43)$$

By inverse Fourier transform we get the Shannon wavelet in time domain:

$$\begin{aligned} \psi(t) &= \mathcal{F}^{-1}[\Psi(f)] = \int_{-\infty}^{\infty} \Psi(f) e^{j2\pi ft} df = \int_{-f_2}^{-f_1} e^{j2\pi ft} df + \int_{f_1}^{f_2} e^{j2\pi ft} df \\ &= \frac{1}{\pi t} [\sin(2\pi f_2 t) - \sin(2\pi f_1 t)] \end{aligned} \quad (10.44)$$

The Shannon wavelet and its spectrum are shown in Fig.10.2. Obviously this wavelet has very good frequency locality but poor temporal locality. However, this wavelet has some significance in the discussion of an algorithm for the reconstruction of the time signal, to be considered later.

- **Morlet wavelet**

The Morlet wavelet is a complex exponential $e^{j\omega_0 t}$ modulated by a normalized Gaussian function $e^{-t^2/2}/\sqrt{2\pi}$:

$$\psi(t) = \frac{1}{\sqrt{2\pi}} e^{j\omega_0 t} e^{-t^2/2} = \frac{1}{\sqrt{2\pi}} [\cos(\omega_0 t) e^{-t^2/2} + j \sin(\omega_0 t) e^{-t^2/2}] \quad (10.45)$$

According to the frequency shift property of the Fourier transform (Eq.3.103), the spectrum of the Morlet wave is another Gaussian function shifted by $-\omega_0$:

$$\begin{aligned} \Psi(\omega) &= \mathcal{F}[\psi(t)] = \int_{-\infty}^{\infty} \psi(t) e^{-j\omega t} dt = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-t^2/2} e^{j(\omega-\omega_0)t} dt \\ &= e^{-(\omega-\omega_0)^2/2} = e^{-(2\pi(f-f_0))^2/2} \end{aligned} \quad (10.46)$$

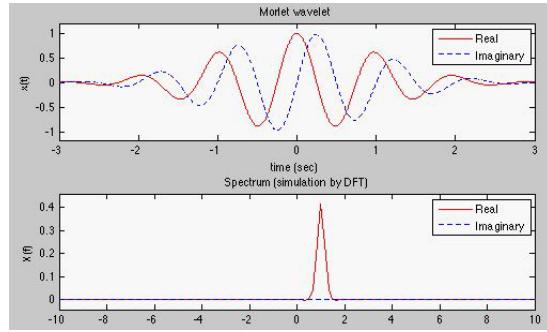


Figure 10.3 Morlet wavelet (top) and its Spectrum (bottom)

The Morlet wavelet and its spectrum are shown in Fig.10.3. Note that when $\omega_0 = 0$, $\Psi(0) = e^{-\omega_0^2/2} > 0$, violating the admissibility condition. However, if ω_0 is large enough, e.g., $f_0 = 1$ Hz or $\omega_0 = 2\pi$, $\Psi(0) = e^{-6.28^2/2} = 2.7 \times 10^{-9}$ is small enough to be neglected. As the Fourier spectrum $\Psi(\omega)$ of the Morlet wavelet is zero when $\omega < 0$, it is an analytic signal according to the definition discussed in Chapter 3.

- **Derivative of Gaussian (DoG)**

This wavelet is the first order derivative of a normalized Gaussian function $g(t) = e^{-\pi(t/a)^2}/a$:

$$\psi(t) = \frac{d}{dt}g(t) = \frac{d}{dt}\left[\frac{1}{a}e^{-\pi(t/a)^2}\right] = -\frac{2\pi t}{a^3}e^{-\pi(t/a)^2} \quad (10.47)$$

Note that the Gaussian function is normalized

$$\int_{-\infty}^{\infty} g(t)dt = 1 \quad (10.48)$$

and the parameter a is related to the standard deviation σ by $a = \sqrt{2\pi\sigma^2}$. The Fourier transform of this derivative of Gaussian can be easily found according to the time derivative property of the Fourier transform (Eq. 3.117) to be

$$\Psi(f) = \mathcal{F}[\psi(t)] = j2\pi fte^{-\pi(af)^2} \quad (10.49)$$

- **Marr wavelet (Mexican hat)**

The Marr wavelet is the negative version of the second derivative of the Gaussian function $g(t) = e^{-\pi(t/a)^2}/a$:

$$\psi(t) = -\frac{d^2}{dt^2}g(t) = -\frac{d}{dt}\left[-\frac{2\pi t}{a^3}e^{-\pi(t/a)^2}\right] = \frac{2\pi}{a^3}(1 - \frac{2\pi t}{a^2})e^{-\pi(t/a)^2} \quad (10.50)$$

If we let $a = \sqrt{2\pi\sigma^2}$, the Gaussian function $g(t)$ is normalized $\int g(t)dt = 1$, and the Marr wavelet becomes:

$$\psi(t) = \frac{1}{\sqrt{2\pi\sigma^3}}(1 - \frac{t^2}{\sigma^2})e^{-t^2/2\sigma^2} \quad (10.51)$$

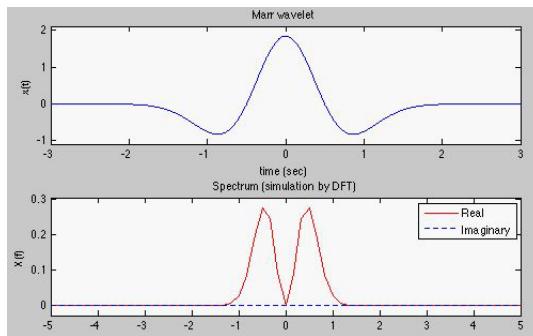


Figure 10.4 Marr wavelets of different scale levels and their spectra

The Marr wavelet function is also referred to as the Mexican hat function due to its waveform. The Fourier transform of the Gaussian function is also Gaussian (Eq.3.160):

$$\mathcal{F}\left[\frac{1}{a}e^{-\pi(t/a)^2}\right] = e^{-\pi(af)^2} \quad (10.52)$$

and according to the time derivative property of the Fourier transform (Eq.3.117), we get the spectrum of the Marr wavelet

$$\Psi(f) = \mathcal{F}[\psi(t)] = -(j2\pi ft)^2 e^{-\pi(af)^2} = 4\pi^2 f^2 e^{-\pi(af)^2} \quad (10.53)$$

The Marr wavelet and its Fourier transform are shown in Fig.10.4.

- **Difference of Gaussians**

As the name suggests, this wavelet is simply the difference between two Gaussian functions with different parameters $a_1 > a_2$ (representing the variance):

$$\psi(t) = g_1(t) - g_2(t) = \frac{1}{a_1}e^{-\pi(t/a_1)^2} - \frac{1}{a_2}e^{-\pi(t/a_2)^2} \quad (10.54)$$

The spectrum of this function is the difference between the spectra of the two Gaussian functions, which are also Gaussian:

$$\Psi(f) = G_1(f) - G_2(f) = e^{-\pi(a_1 f)^2} - e^{-\pi(a_2 f)^2} \quad (10.55)$$

Note that $\Psi(0) = 0$ as required. As can be seen in Fig.10.5, the difference of Gaussians looks very much like the second derivative of Gaussian (Marr) wavelet, and both functions could be abbreviated as DoG. But note that they are two different types of functions.

10.5 Discrete-Time Wavelet Transform (DTWT)

10.5.1 Discretization of Wavelet Functions

In order to actually obtain the wavelet transform of a real time signal in practice, we need to discretize both the signal and the wavelet functions and the resulting

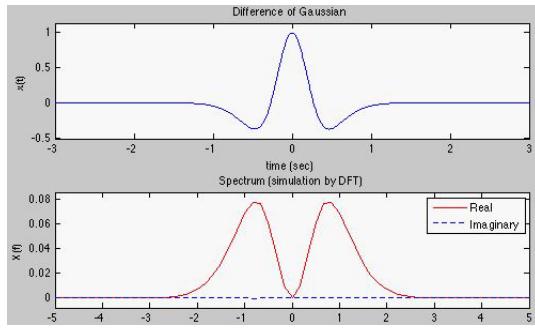


Figure 10.5 Difference of Gaussians and its spectrum

discrete version of the wavelet transform is the discrete-time wavelet transform (DTWT), which can be carried out numerically. Specifically, we need not only sample both the time signal $x(t)$ and the mother wavelet function $\psi(t)$ to get a set of finite samples $x[n]$ and $\psi[n]$ ($n = 0, \dots, N - 1$), but also discretize the scale factor s to get a set of finite daughter wavelet functions of different scales $\psi_{s_l,0}[n] = \psi[n/s_l]$ ($l = 1, \dots, S$). Here the scale factor s_l is defined as an exponential function of the scale index l :

$$s_l = s_0 2^{l/r} = s_0 (2^{1/r})^l \quad (10.56)$$

where s_0 is the base scale and r is a parameter that controls the total number of scale levels $S = r \log_2(N/s_0)$.

Having discretized the time signal and the mother wavelet, we can also obtain their DFT coefficients $X[k] = \mathcal{F}[x[n]]$ and $\Psi_{s_l,0}[k] = \mathcal{F}[\psi_{s_l,0}[n]]$ (with s_l treated as a parameter) ($k = 0, \dots, N - 1$). When the mother wavelet function $\psi[n]$ is scaled by $s_l > 1$, it is expanded in time domain to becomes $\psi_{s_l,0}[n] = \psi[n/s_l]$, and its spectrum is compressed in frequency domain to become $\Psi_{s_l,0}[k] = \Psi[s_l k]$. When $l = 1$, the mother wavelet is scaled minimally by a factor $s_0 r^{1/r}$, but when $l = S$, it is maximally expanded by a factor of $s_l = s_0 2^{S/r} = s_0 2^{\log_2(N/s_0)} = N$, and its N-point Fourier spectrum $\Psi_{s_l,0}[k] = \Psi[Nk]$ is maximally compressed to become a single point. Moreover, if $r > 1$, the base of the exponent is reduced from 2 to $2^{1/r} < 2$ for a finer scale resolution with a smaller step size between two consecutive scale levels. For example, when $r = 2$, the base of the exponent in Eq.10.56 is reduced from 2 to $\sqrt{2} = 1.442$, and the total number of scale levels is correspondingly doubled and the scale resolution is increased. Particularly when $s_0 = r = 1$, we have $s_l = 2^l$, and the corresponding transform is called *dyadic wavelet transform*.

The exponentially scaled Shannon, Morlet and Marr wavelets are shown in Figs.10.6, 10.7 and 10.8.

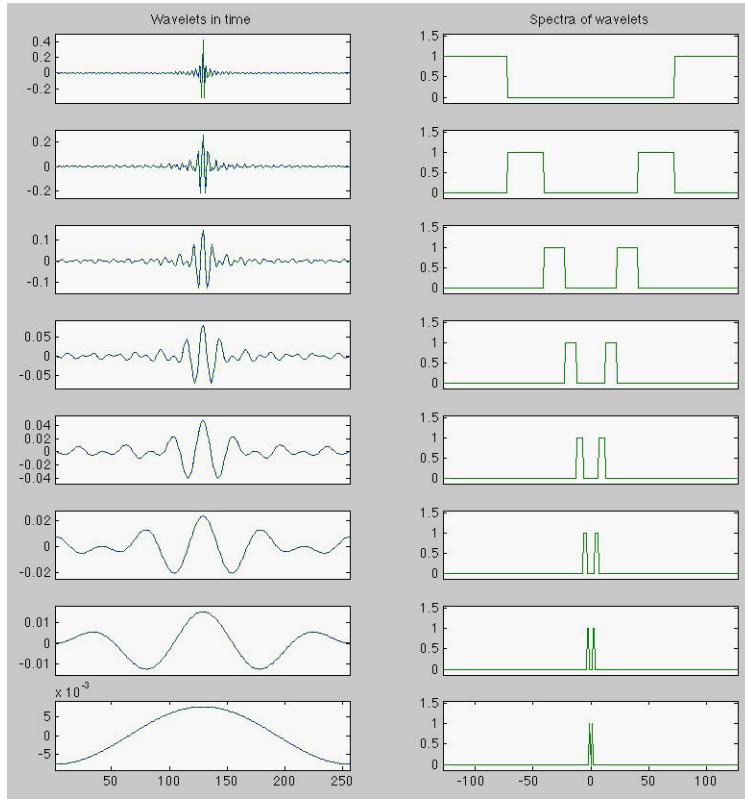


Figure 10.6 The Shannon wavelets (left) and their spectra (right)

10.5.2 The Forward and Inverse Transform

Following Eq.10.15, we can obtain the DTWT coefficients of a discrete signal $x[n]$ at scale level l as a correlation of the signal and the wavelet function $\psi_{s_l,0}[n]$:

$$X[l, n] = \mathcal{W}[x[n]] = \sum_{n=0}^{N-1} x[n] \overline{\psi}_{s_l,l}[n] = \sum_{n=0}^{N-1} x[n] \overline{\psi}_{s_l,0}[n-l] = x[n] * \psi_{s_l,0}[n] \quad (10.57)$$

Same as Eq.10.16 in the continuous case, the DTWT can also be carried out as a multiplication in frequency domain (with scale index l treated as a parameter):

$$\hat{X}[l, k] = \mathcal{F}[X[l, n]] = \mathcal{F}[\mathcal{W}[x[n]]] = X[k] \overline{\Psi}_{s_l,0}[k] \quad (10.58)$$

where $\hat{X}[l, k]$ is the DFT of the DTWT $X[l, n]$ of the signal $x[n]$. Taking the inverse DFT on both sides of the equation above we get the DTWT in time domain:

$$X[l, n] = \mathcal{F}^{-1} [\hat{X}[l, k]] = \mathcal{F}^{-1} [X[k] \overline{\Psi}_{s_l,0}[k]] \quad (10.59)$$

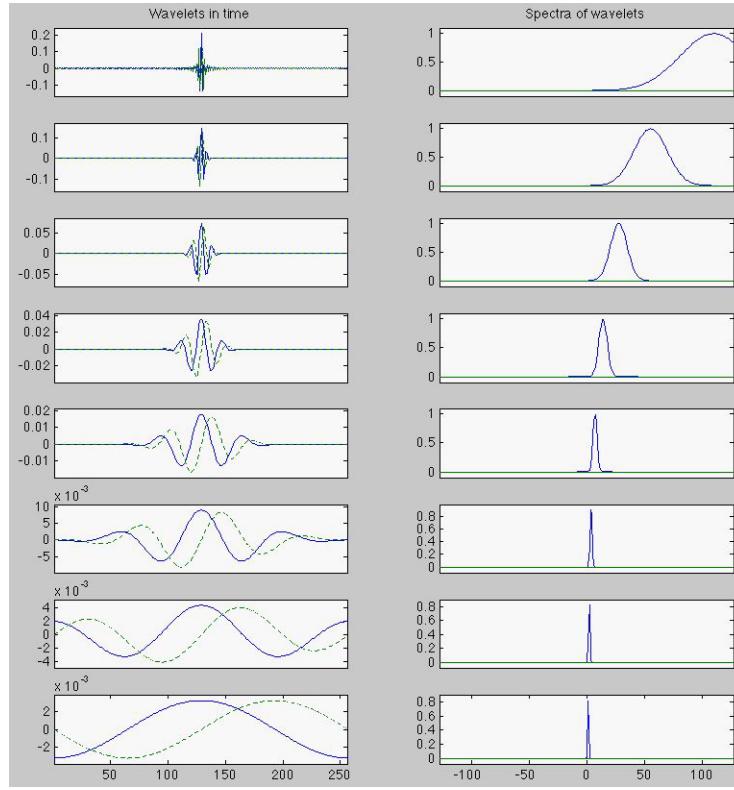


Figure 10.7 The Morlet wavelets (left) and their Spectra (right)

The inverse DTWT can also be more conveniently obtained in frequency domain, similar to the derivation of the inverse transform in Eq. 10.25 for the continuous case. We first multiply both sides of Eq.10.58 by $\Psi_{s_l,0}[k]$ and then sum both sides over all scale levels to get:

$$\sum_{l=1}^S \hat{X}[l,k] \Psi_{s_l,0}[k] = \sum_{l=1}^S [X[k] \bar{\Psi}_{s_l,0}[k]] \Psi_{s_l,0}[k] = X[k] \sum_{l=1}^S |\Psi_{s_l,0}[k]|^2 \quad (10.60)$$

But according to Eq.10.23, the summation of the daughter wavelets squared over all scales is a constant, i.e., in discrete case we have:

$$\sum_{l=1}^S |\Psi_{s_l,0}[k]|^2 = C \quad (10.61)$$

Now the above equation becomes:

$$X[k] = \frac{1}{C} \sum_{l=1}^S \hat{X}[l,k] \Psi_{s_l,0}[k] \quad (10.62)$$

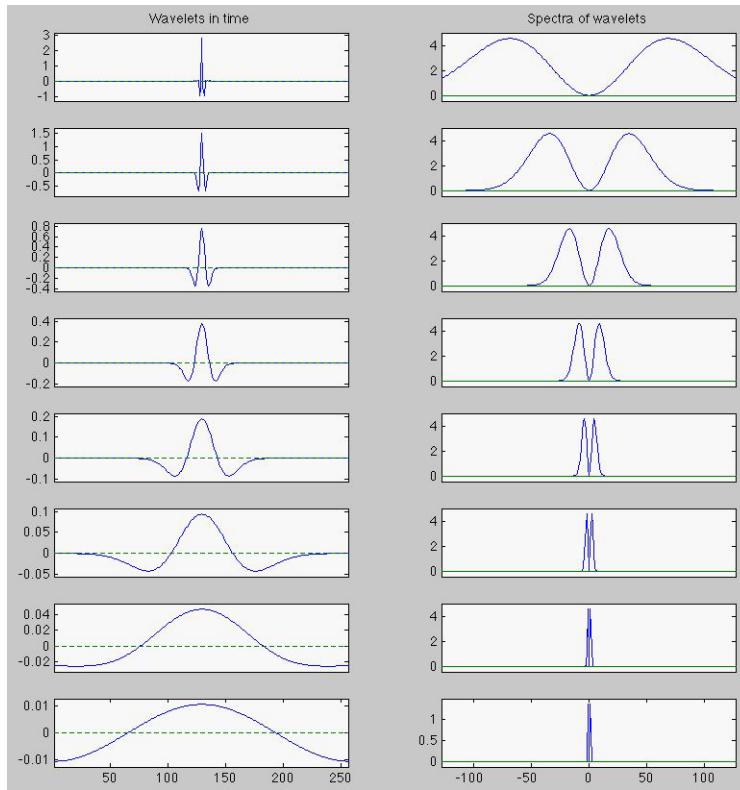


Figure 10.8 The Marr wavelets (left) and their Spectra (right)

Taking inverse DFT on both sides we get the inverse DTWT by which the original time signal $x[n]$ is reconstructed:

$$x[n] = \mathcal{F}^{-1}[X[k]] = \mathcal{F}^{-1} \left[\frac{1}{C} \sum_{l=1}^S \hat{X}[l, k] \Psi_{s_l, 0}[k] \right] \quad (10.63)$$

10.5.3 A Fast Inverse Transform Algorithm

We will now show that the inverse DTWT can be more conveniently obtained by a fast algorithm without actually carrying out Eq.10.63. To do so, we first show that the sum of the DFT coefficients $\Psi_{s_l, 0}[k] = \mathcal{F}[\psi_{s_l, 0}[n]]$ of the daughter wavelets over all exponential scales $s_l = s_0(2^{1/r})^l$ (Eq.10.56) is a constant:

$$\sum_{l=1}^S \Psi_{s_l, 0}[k] = \sum_{l=1}^S \Psi[s_l k] = \sum_{l=1}^S \Psi[s_0 2^{l/r} k] = C, \quad (\text{for all } n \neq 0) \quad (10.64)$$

where the constant C is in general not the same as that in Eq.10.61. This equation holds for all k for different frequency components, independent of the specific waveform of the mother wavelet.

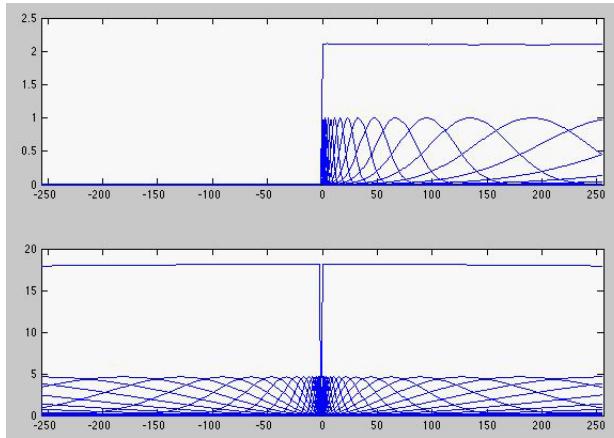


Figure 10.9 Summations of the spectra of Morlet wavelets (top) and Marr (Mexican hat) wavelets (bottom)

To prove Eq.10.64, we first consider the corresponding situation in the continuous case, the integral of an arbitrary function $\Psi(f)$ scaled exponentially by a factor $s = b^u$:

$$\begin{aligned} \int_{-\infty}^{\infty} \Psi(b^u f) du &= \int_{-\infty}^{\infty} \Psi(sf) d(\log_b s) = \frac{1}{\ln b} \int_0^{\infty} \Psi(sf) \frac{ds}{s} \\ &= \frac{1}{\ln b} \int_0^{\infty} \frac{\Psi(sf)}{sf} d(sf) = \frac{1}{\ln b} \int_0^{\infty} \frac{\Psi(s')}{s'} ds' = C \end{aligned} \quad (10.65)$$

Here we have assumed $s' = sf$, and that the integral converges to some constant. Note that this result is independent of the variable f , i.e. the integral of all exponentially scaled versions of any function $\Psi(f)$ is a constant over the entire domain f of the function, irrelevant to the specific waveform of the function. As a discrete approximation of the integral in Eq.10.65, the summation in Eq.10.64 should also converge to a constant, so long as the resolution of the different scales is high enough (large enough value for parameter r). For example, as shown Fig.10.9, the spectra of the exponentially scaled Morlet and Marr wavelets do indeed sum up to a constant over the frequency f .

Eq.10.64 still holds if we take the complex conjugate on both sides, i.e., $\bar{\Psi}_{s_l,0}[k]$ also add up to a constant $\sum_{l=1}^S \bar{\Psi}_{s_l,0}[k] = C$. Also note that In fact the DFTs of most typical wavelets are real $\bar{\Psi}_{s_l,0}[k] = \Psi_{s_l,0}[k]$.

We are now ready to consider the fast algorithm for the inverse DTWT. Specifically we will show that the inverse DTWT can be carried out simply by summing

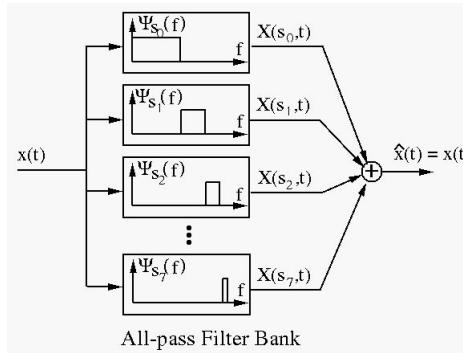


Figure 10.10 All-pass filter bank composed of band-pass wavelets

all the DTWT coefficients obtained by Eq.10.59:

$$\begin{aligned} \sum_{l=1}^S X[l, n] &= \sum_{s=1}^S \mathcal{F}^{-1} [X[k] \bar{\Psi}_{s_l, 0}[k]] = \sum_{l=1}^S \left[\sum_{k=0}^{N-1} X[k] \bar{\Psi}_{s_l, 0}[k] e^{j2\pi nk/N} \right] \\ &= \sum_{k=0}^{N-1} X[k] \left[\sum_{l=1}^S \bar{\Psi}_{s_l, 0}[k] \right] e^{j2\pi nk/N} = C \sum_{k=0}^{N-1} X[k] e^{j2\pi nk/N} = C x[n] \quad (10.66) \end{aligned}$$

where $C = \sum_{l=1}^S \bar{\Psi}_{s_l, 0}[k]$ according to Eq.10.64, and we note the last sign above is due to the inverse DFT. Now the original time signal can be trivially obtained from its DTWT coefficients:

$$x[n] = \frac{1}{C} \sum_{l=1}^S X[l, n] = \frac{\sum_{l=1}^S X[l, n]}{\sum_{l=1}^S \bar{\Psi}_{s_l, 0}[k]} \quad (10.67)$$

This fast algorithm for the inverse DTWT can be considered as an all-pass filter bank illustrated in Fig.10.10. We first consider the DTWT based on the Shannon dyadic wavelet, which is an ideal band-pass filter in frequency domain (Eq.10.43) that preserves all information of the signal inside the passing band $\Delta f = f_2 - f_1$, while suppresses all frequency components outside to zero. Moreover, as shown in Fig.10.6, the Shannon wavelets $\Psi_{s_l}(f)$ corresponding to all dyadic scales form a filter bank that completely covers the frequency range without any overlap or gap, i.e., Eq.10.64 is indeed satisfied. Collectively these ideal band-pass filters form an all-pass filter bank with a constant frequency response through out all frequencies except at $f = 0$ where $\Psi_{s_l, 0}[0] = 0$ for all $l = 1, \dots, S$ (Eq.10.10), as required by the admissibility condition. The outputs of these band-pass filters are simply the DTWT coefficients $X[l, n]$ carrying all signal information. Obviously the signal can be perfectly reconstructed as the sum of the outputs from all filters in the filter bank, as indicated in Eq.10.67.

The wavelet transform can therefore represented by the all-pass filter bank shown in Fig.10.10. The forward transform is implemented as the band-pass filtering process by which the DTWT coefficients $X(s_l, \tau)$ for different scales s_l and translation τ are produced, and the inverse transform is implemented as the

summation of the outputs of these band-pass filters by which the time signal is perfectly reconstructed.

The Shannon wavelets assumed in the discussion above can be generalized to any other wavelet function, such as the Morlet and Marr wavelets. Although as band-pass filters they are overlapped, they still form an all-pass filter bank with constant gain over the entire frequency range due to Eq.10.64, as shown in Fig.10.9. The information contained in the signal is preserved collectively by all band-pass filters in the filter bank, and the signal can be reconstructed simply by summing their outputs.

Example 10.1: The wavelet transform of a sawtooth time signal of $N = 128$ samples is shown in Fig.10.11. Here we choose to use the Morlet wavelets of $S = 8$ different scale levels, corresponding to the same number of band-pass filters. These wavelets $\psi_{s_l}(t)$ in time domain and their spectra $\Psi_{s_l}(f)$ in frequency domain have already been shown in Fig. 10.7. The DTWT coefficients $X[l, n]$ corresponding to different scale levels s_l are shown on the left of Fig.10.11, and their partial sums are shown on the right, where the l th panel is the partial sum of the first l scale levels. We see that the partial sums as the approximation of the original sawtooth signal $x[n]$ improves progressively as more scale levels are included, until eventually a perfect reconstruction of the signal is obtained when all S scale levels are included.

10.6 Wavelet Transform Computation

Here we give a few segments of C code for the implementation of the DTWT algorithm discussed above.

- Generation of S scale levels:

```
r=2;                                // scale resolution
s0=1;                                // base scale
S=r*log2((float)N/s0);      // number of scale levels
scale=alloc1df(S);                  // allocate memory for S scales
for (l=0; l<S; l++) {
    scale[l]=s0*pow(2.0,(l+1)/r); // lth scale s_l
}
```

The scales corresponding to three different sets of parameters are plotted in Fig.10.12 to show how the resolution r and base scale s_0 affect the scales s_l .

- Generation of wavelet functions:

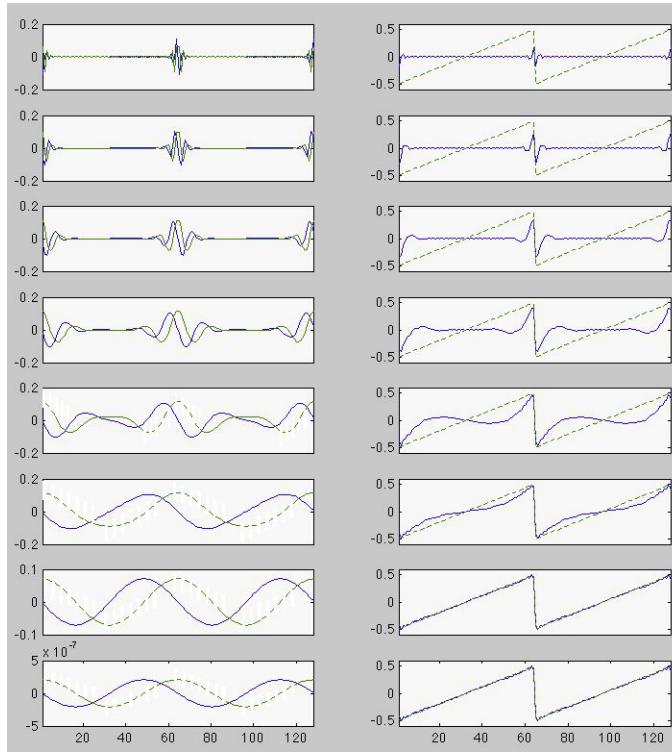


Figure 10.11 The reconstruction of a sawtooth signal (right) as the sum of its DTWT (left)

The DTWT coefficients over $S = 8$ scale levels are shown on the left (solid and dashed curves for the real and imaginary parts), while the partial sums of the DTWT coefficients of l scale levels are shown on the right (solid curves), compared with the original signal (dashed curves).

As both forward and inverse DTWT are more conveniently carried out in frequency domain, the spectra of the wavelet functions will be specified and used in the code. First we show the code for generating Morlet wavelets of S scales:

```
f0=0.6;                                // wavelet parameter
for (l=0; l<S; l++) {                  // for all S scale levels
    for (n=0; n<N; n++) {            // for all N frequencies
        v=2*Pi*(scale[l]*((float)(n-N/2)/N)-f0); // DC in middle
        waver[l][n]=exp(-v*v/2); // spectrum (real)
        wavei[l][n]=0;           // spectrum (imaginary)
    }
}
```

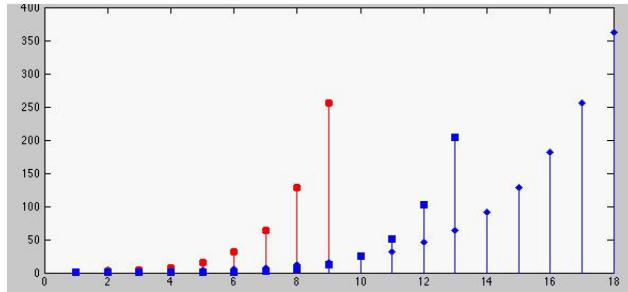


Figure 10.12 Scales s_l versus $l = 1, \dots, S$ corresponding to different parameters r and s_0 for DTWT of a signal with $N = 512$ samples

The circles represent $S = 9$ scales corresponding to $r = 1$ and $s_0 = 1$; the squares represent $S = 13$ scales corresponding to $r = 1$ and $s_0 = 0.05$; and the diamonds represent $S = 18$ scales corresponding to $r = 2$ and $s_0 = 1$.

In the code “waver” and “wavei” are two 2-D arrays for the real and imaginary parts of the wavelet spectrum for N samples (frequencies) and S scales. Also shown below is the code for generating Mexican hat wavelets of S scales:

```
a=2;                                // wavelet parameter
for (l=0; l<S; l++) {                // for all S scale levels
    for (n=0; n<N; n++) {            // for all N frequencies
        v=a*scale[l]*(n-N/2)/N;     // DC in middle
        waver[1][n]=4*Pi*Pi*v*v*exp(-Pi*v*v); // spectrum (real)
        wavei[1][n]=0;                // spectrum (imaginary)
    }
}
```

- The forward DTWT:

Here we assume the real and imaginary parts of the time signal are stored in two $N \times 1$ arrays xr and xi , respectively, and the real and imaginary parts of the DTWT of the time signal are stored in two $S \times N$ arrays Xr and Xi for wavelet coefficients of S scales and N time translations:

```
dft(xr,xi,N,0);                  // DFT of signal
for (l=0; l<S; l++) {            // for all S scale levels
    for (n=0; n<N; n++) {        // for all N frequencies
        Xr[1][n]=xr[n]*waver[1][n]+xi[n]*wavei[1][n];
        Xi[1][n]=xi[n]*waver[1][n]-xr[n]*wavei[1][n];
    }
    dft(Xr[1],Xi[1],N,1);       // inverse DFT, back to time
}
```

- The inverse DTWT:

Listed below is the code for the inverse DTWT algorithm based on Eq.10.63. Again, the real and imaginary parts of the DTWT coefficients are stored in the two $S \times N$ arrays Xr and Xi , and the real and imaginary parts of the reconstructed time signal are in two $N \times 1$ arrays yr and yi , respectively.

```

for (n=0; n<N; n++)
    yr[n]=yi[n]=0;                      // initialization
    for (l=0; l<S; l++) {                // for all S scale levels
        dft(Xr[l],Xi[l],N,0);           // DFT of DTWT coefficients
        for (n=0; n<N; n++) {
            yr[n]=yr[n]+Xr[l][n]*waver[l][n]-Xi[l][n]*wavei[l][n];
            yi[n]=yi[n]+Xr[l][n]*wavei[l][n]+Xi[l][n]*waver[l][n];
        }
        dft(yr,yi,N,1);                 // inverse DFT back to time
    }
}

```

The code based on Eq.10.67 is trivial and not listed.

A set of typical signals as well as their DTWT transforms based on both the Marr and Morlet wavelets are shown in Fig.10.13 in image forms. These signals include sinusoids and their combinations, a chirp signal (a sinusoid with continuously changing frequency), square, sawtooth, and triangle waves, impulse train and random noise.

10.7 Filtering Based on Wavelet Transform

Similar to Fourier filtering (LP, HP, BP, etc.) that takes place in frequency domain, various wavelet filtering can also be carried out in the transform domain where the wavelet coefficients $X[l, n]$ are modified to achieve certain desired effects for purposes such as noise reduction and information extraction. Here we consider a set of examples that illustrate the filtering effects based on the wavelet transform in comparison with those based on the Fourier transform.

Example 10.2: The monthly Dow Jones Industrial Average (DJIA) from 1999 to 2008 as a time function and its Fourier spectrum are plotted in the top two panels of Fig.10.14. The low-pass (LP) filtered Fourier spectrum is plotted in panel 3. Similar LP-filtering is also carried out based on the wavelet transform (Morlet), as shown in Fig.10.15. The LP-filtered data obtained by both the Fourier and wavelet transforms are re-plotted as the solid and dashed curves respectively in panel 4, in comparison with the original one as the dotted curve. We see that the LP-filtered curves by both transform methods are very similar to each other, and, as expected, they are both much smoother than the original one.

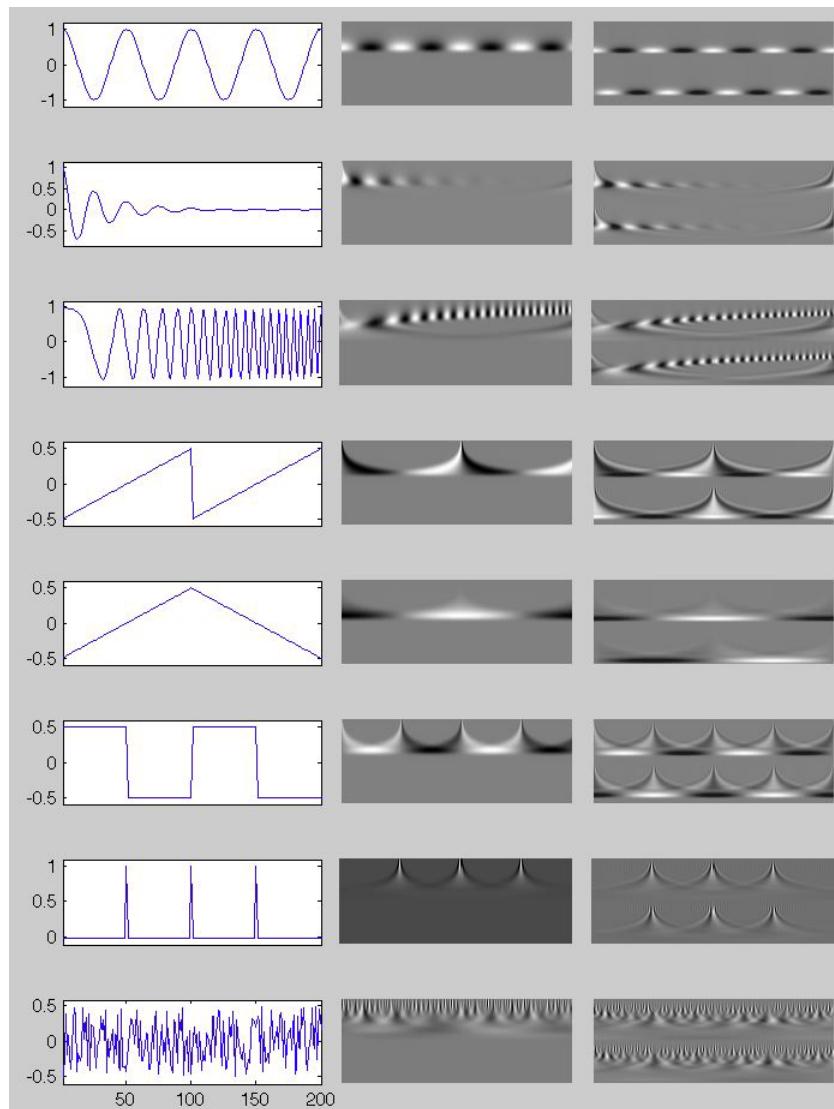


Figure 10.13 Typical signals (left) and their real DTWT based on the Marr wavelets (middle) and complex DTWT based on Morlet wavelets. The real and imaginary parts of the Morlet DTWT are shown in the upper and lower parts, respectively.

Example 10.3: A chirp is a sinusoidal signal whose frequency is monotonically and continuously changing, either linearly or exponentially. Here we compare the filtering effects of an exponential chirp based on both the Fourier transform and the wavelet transform. As the frequency changes over time, it may seem that

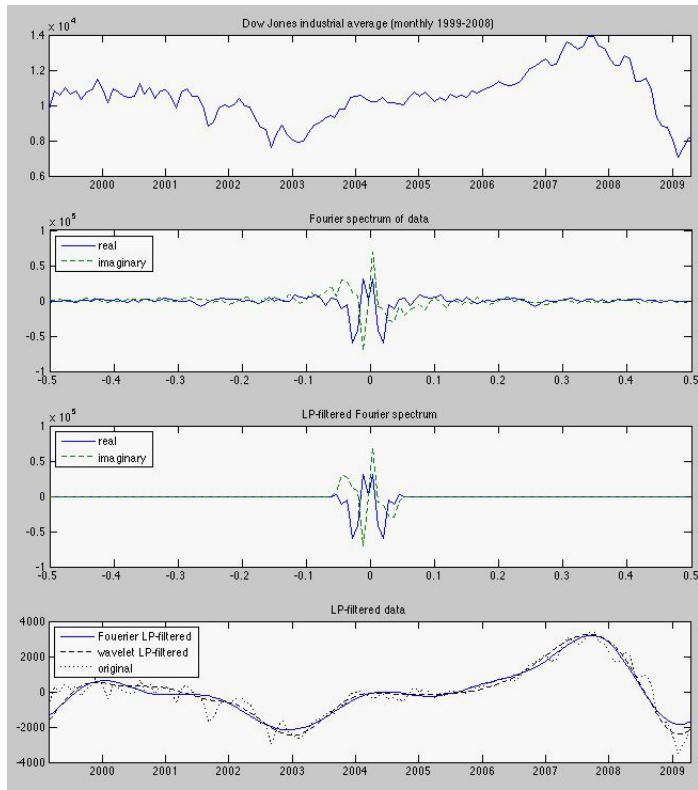


Figure 10.14 Monthly Dow Jones Industrial Average (DJIA) from 1999 to 2008

The four panels are, respectively, the DJIA index, its Fourier spectrum, the LP-filtered spectrum, and the LP-filtered data by both the Fourier and wavelet transforms.

filtering out certain frequency should only affect the signal locally in the time segment corresponding to the frequency removed. However, this is not actually the case if the filtering is carried out in Fourier domain.

A chirp and its Fourier spectrum are shown respectively in the first and second panels of Fig.10.16. Then certain frequency components in the spectrum are suppressed to zero by an ideal band-pass filter, as shown in the 3rd panel. The signal is then reconstructed by the inverse Fourier transform as shown in the bottom panel. Note that although only the frequency components within a relatively narrow band are suppressed, the entire time signal is affected, including the slow changing portion of the signal on the very left, as well as the time interval (roughly from 150 to 250) corresponding to the frequencies suppressed. This is due to the nature of the Fourier transform that the frequency information is extracted from the entire time span of the signal, and those frequency components that are suppressed also contribute to the slow changing portion of the signal as well.

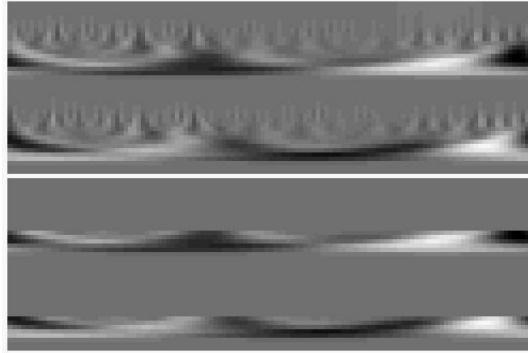


Figure 10.15 LP filtering of DJIA data based on Morlet wavelet transform

The DTWT coefficients before and after LP-filtering are shown respectively in the top and bottom panels. After filtering, the coefficients at higher scale levels are suppressed to zero (gray in the image).

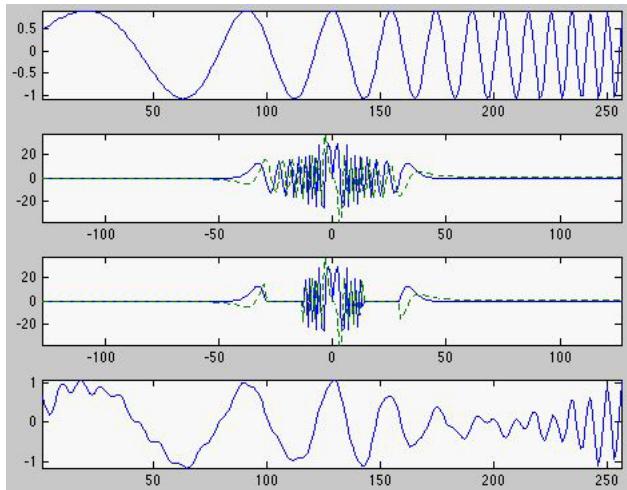


Figure 10.16 FT filtering of chirp signal

On the other hand, the filtering based on the wavelet transform demonstrates some different effect, as shown in Fig.10.17, where the same chirp and its DTWT coefficients are shown in the top two panels, and the filtering in transform domain and the reconstructed signal are shown respectively in the bottom two panels. Similar to the Fourier filtering, here the DTWT coefficients inside a certain band of scale levels are suppressed to zero. However, different from the Fourier filtering, only a local portion (also roughly from 150 to 250) of the reconstructed signal corresponding to the suppressed scale levels is significantly affected, while the waveforms of the signal outside the interval remain mostly the same. This very

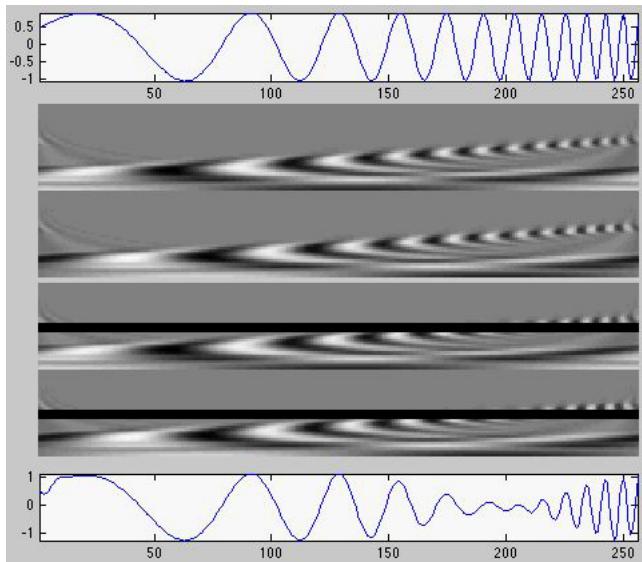


Figure 10.17 CTWT filtering of chirp signal

different filtering effect reflects the fact that the wavelet transform possesses the temporal locality as well as frequency (scale levels) locality.

Example 10.4: One weakness of the Fourier transform is that it is insensitive to non-stationary characteristics in the signal because the frequency information is extracted from the entire signal duration without temporal locality. Here we consider a signal before and after it is contaminated by some spiky noise, as shown on the top and bottom panels on the left of Fig.10.18, and the corresponding Fourier spectra shown on the right. As we can see, the spiky noise has a very wide energy distribution spreading over the entire spectrum, i.e., all frequency components of the signal are affected by the noise. In particular, some of the weaker frequency components in the signal are completely overwhelmed by the noise, and it is obvious that separating the noise from the signal by Fourier filtering is extremely difficult.

This problem of noise removal can be addressed by the wavelet filtering, as shown in Fig.10.19. The original signal and its reconstructions after high-pass and low-pass filtering are shown respectively in the top, middle and bottom panels on the left, while the corresponding wavelet coefficients are shown on the right. We see that it is now possible to separate the noise from the signal by wavelet filtering, due obviously to the temporal locality of the wavelet transform. The spiky noise is separated out by high-pass filtering (middle left), while the signal is reasonably recovered after low-pass filtering (lower left).

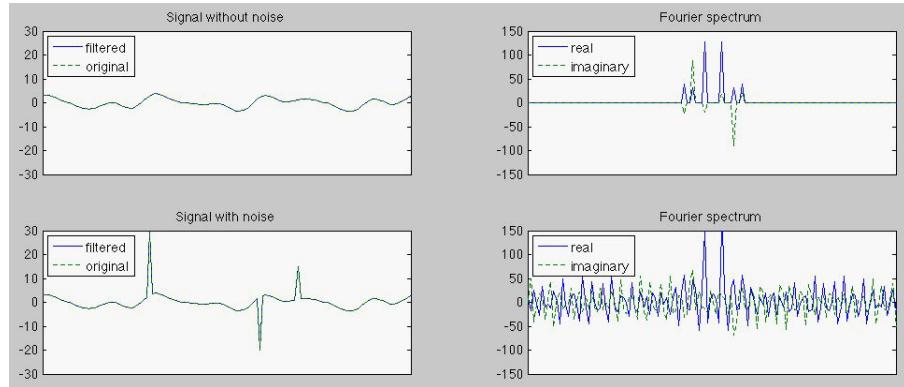


Figure 10.18 A noise-contaminated signal and its Fourier spectrum

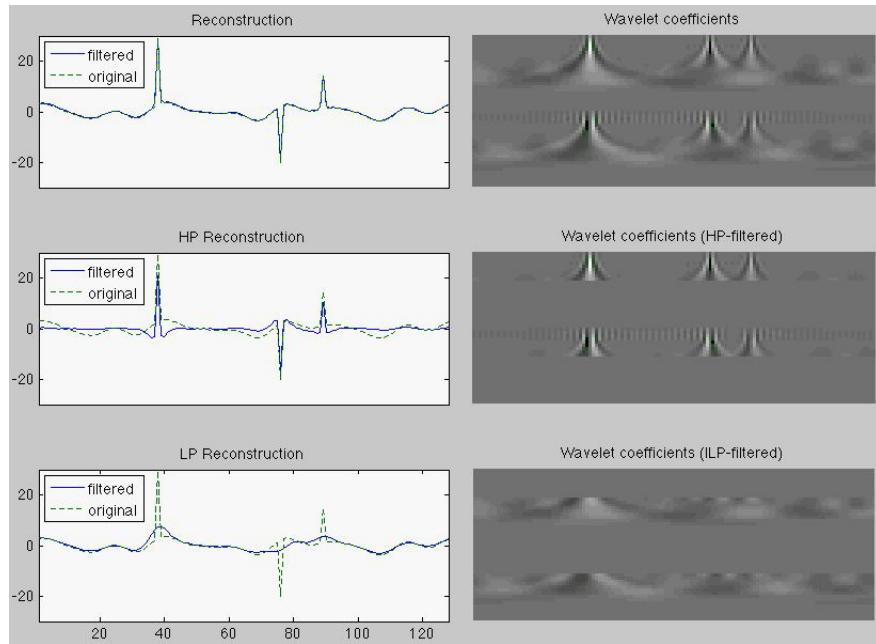


Figure 10.19 Separation of the signal and noise by wavelet filtering

The original signal with spiky noise and its reconstructions after HP and LP filtering are shown respectively in the top, middle and bottom panels on the left, while their wavelet coefficients are shown in the corresponding panels on the right.

Example 10.5: The annual average temperature in Los Angeles area from 1878 to 1997 (NOAA National Weather Service Center in the US) is shown in the top panel of Fig.10.20 (solid curve). The data clearly show a upward trend of the

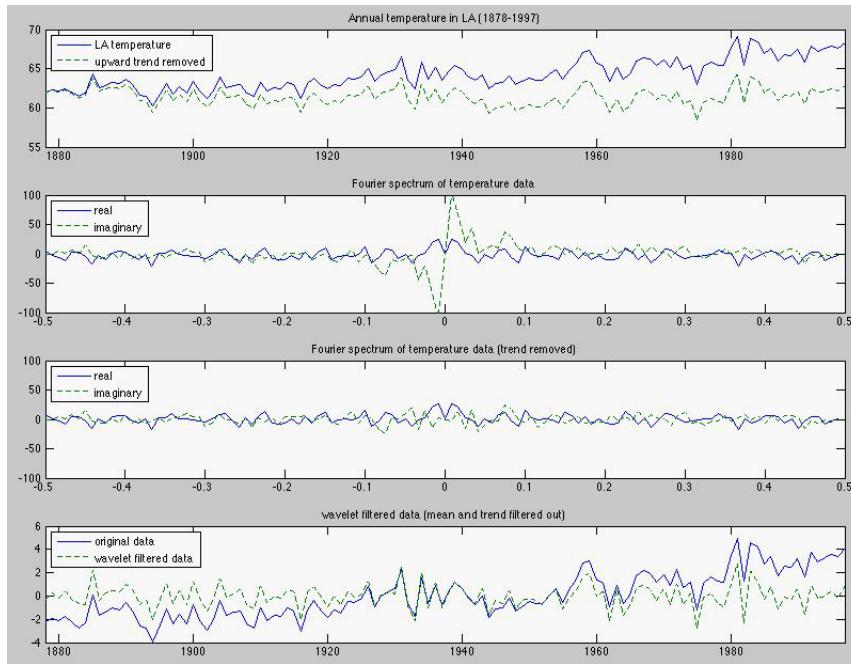


Figure 10.20 Annual temperature in LA area from 1878 to 1997

annual temperature, with a 5.57° F total rise over the 120 years, with an average annual increase of 0.0464° F.

The upward drift in the data can be removed in time domain. We first find the linear regression of the curve in terms of the slope and the intercept representing the trend, and then subtract it from the data. The result is shown as the dashed curve in the top panel of Fig.10.20. We next consider if and how this could also be done by filtering in either the Fourier or wavelet transform domain.

The Fourier spectra of the temperature data with and without the upward drift are shown in the 2nd and 3rd panel of Fig.10.20. We see that their real parts are the same, but their imaginary parts differ significantly at the low frequency region as the upward trend is an odd function, represented by both the positive and negative peaks in the imaginary part of the spectrum in the 2nd panel, which no longer exist in the spectrum in the 3rd panel, when this trend is removed. It is difficult to separate out the slow-changing trend from the rest of the signal by filtering in frequency domain, as their frequency components are mixed.

The filtering effect in the wavelet domain is shown in Fig.10.21. The wavelet coefficients of the signal before and after the removal of the upward trend (detected by linear regression) are shown respectively in the top and middle panels. Also LP-filtering is carried out by suppressing the wavelet coefficients of low scale levels corresponding to the slow-changing trend, as shown in the bottom panel. Then the temperature signal is reconstructed by the inverse wavelet

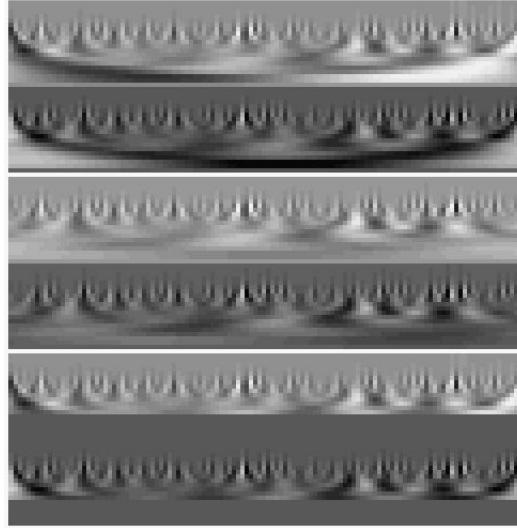


Figure 10.21 Wavelet transform of LA temperature data

In wavelet domain the original data (top) is compared with the same data with the upward trend removed by linear regression (middle), and by LP-filtering (bottom).

transform, as shown in the bottom panel of Fig.10.20. We see that indeed the upward trend is removed by wavelet filtering.

10.8 Homework Problems

1. Prove the time shift property of the CTWT as shown in Eq.10.28.
2. Prove the time scaling property of the CTWT as shown in Eq.10.29.
3. Show that if the center and width of a compactly supported mother wavelet function $\psi(t)$ are respectively t_0 and Δt , then those of a daughter wavelet $\psi_{s,\tau}(t) = \psi((t - \tau)/s)/\sqrt{s}$ are as shown in Eq.10.30.
4. Develop an m-file in Matlab to implement the DTWT algorithm for both the forward and inverse transform. Generate the DTWT of the eight signals in Fig.10.13 based on first the Morlet wavelets and then the Marr wavelet (Mexican hat).
5. Generate the following two signals in Matlab with $f_1 = 5$ and $f_2 = 25$:

—

$$x_1[n] = \cos(2\pi n f_1/N) + \cos(2\pi n f_2/N) \quad (10.68)$$

- composed of two halves of sinusoids of different frequencies

$$\begin{aligned}x_2[n] &= \cos(2\pi n f_1/N), & (n = 0, \dots, N/2 - 1) \\x_2[n] &= \cos(2\pi n f_2/N), & (n = N/2, \dots, N - 1)\end{aligned}\quad (10.69)$$

For the purpose of separating the two frequencies f_1 and f_2 contained in both signals $x_1[n]$ and $x_2[n]$, design a two-channel filter bank composed of two filters so that they each output one of the two frequencies. Carry out this approach based on both Fourier filtering and wavelet filtering.

6. As seen in the text, wavelet transform can achieve locality in both temporal and frequency domain, which is desirable in representing, detecting and possibly removing, if so desire, certain temporal signal features that are either local (such as irregular spikes) or non-stationary (such as long term effects of trend or non-periodic frequency change). Obtain datasets of your own choice that contain such characteristics and carry out filtering to separate such features with the rest of the signal in both Fourier frequency domain and wavelet transform domain. Compared the filtering effects of both methods.
7. Repeat Example 10.3 using Marr wavelets.
8. Repeat Example 10.4 using Marr wavelets.
9. Repeat Example 10.5 using Marr wavelets.

11 Multiresolution Analysis and Discrete Wavelet Transform

In the previous chapter we considered the continuous-time wavelet transform that converts a signal $x(t)$ in 1-D time domain into a 2-D function $X(s, \tau)$ in transform domain, based on the kernel functions $\psi_{s,\tau}(t)$ which are non-orthogonal and redundant. Now we will consider the concept of multiresolution analysis (MRA), also called multi-scale approximation (MSA), based on which various orthogonal and bi-orthogonal wavelets can be constructed as bases that span the function space $L^2(\mathbb{R})$, same as all the orthogonal transforms discussed before. The discrete implementation of this method is called the discrete wavelet transform (DWT), not to be confused with the discrete-time wavelet transform discussed (DTWT) in the previous chapter.

11.1 Multiresolution Analysis (MRA)

11.1.1 Scale Spaces

We can discretize both parameters s and τ in the wavelet function $\psi_{s,\tau}(t)$ defined in Eq.10.13 in a dyadic manner so that it becomes:

$$\psi_{j,k}(t) = \frac{1}{\sqrt{2^{-j}}}\psi\left(\frac{t - 2^{-j}k}{2^{-j}}\right) = 2^{j/2}\psi(2^jt - k), \quad j, k \in \mathbb{Z} = \{\dots, -1, 0, 1, \dots\} \quad (11.1)$$

The mother wavelet $\psi(t)$ is either expanded (dilated) if $j < 0$, or compressed if $j > 0$. In either case, it is also translated by an integer amount in time to the right if $k > 0$ or to the left if $k < 0$. While constructing the specific mother wavelet function $\psi(t)$, we can further impose the orthogonality requirement so that all wavelets $\psi_{j,k}(t)$ are orthogonal with respect to not only integer translation (in terms of k) but also dyadic scaling (in terms of j). In other words, at any given scale level j , these wavelets form an orthogonal basis that spans a space at the level, and all bases across different scale levels are also orthogonal to each other. In the following, we will develop the theory for the construction of such a set of orthogonal wavelet basis functions across different scale levels.

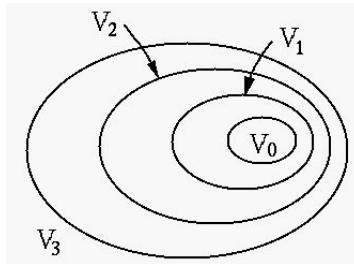


Figure 11.1 The nested V_j spaces for MRA

Definition: A multiresolution analysis (MRA) is a sequence of nested *scale spaces* $V_j \subset L^2(\mathbb{R})$:

$$\{0\} = V_{-\infty} \subset \cdots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots \subset V_\infty = L^2(\mathbb{R}) \quad (11.2)$$

that satisfies the following conditions:

- Completeness: the union of the nested spaces is the entire function space and their intersection is a set containing 0 as its only member:

$$\bigcup_{j \in \mathbb{Z}} V_j = V_\infty = L^2(\mathbb{R}), \quad \bigcap_{j \in \mathbb{Z}} V_j = V_{-\infty} = \{0\} \quad (11.3)$$

- Self-similarity in scale:

$$x(t) \in V_0 \quad \text{iff} \quad x(2^j t) \in V_j, \quad j \in \mathbb{Z} \quad (11.4)$$

- Self-similarity in translation:

$$x(t) \in V_0 \quad \text{iff} \quad x(t - k) \in V_0, \quad k \in \mathbb{Z} \quad (11.5)$$

- Existence of a function $\theta(t) \in V_0$ so that the family $\{\theta(t - k) \mid (k \in \mathbb{Z})\}$ is a Riesz basis (linearly independent frame, Definition 2.25) that spans V_0 :

$$V_0 = \text{span}(\theta(t - k), \quad k \in \mathbb{Z}) \quad (11.6)$$

The self-similarities in scale and translation can be combined if we relate Eq.11.4 to Eq. 11.5 with t replaced by $2^j t$:

$$x(t) \in V_0 \quad \text{iff} \quad x(2^j t) \in V_j \quad \text{iff} \quad x(2^j t - k) \in V_j \quad (11.7)$$

If we define another function $y(t) = x(2^j t)$, then the two self-similarities above can be expressed as:

$$y(t) \in V_j \quad \text{iff} \quad y(t - 2^{-j} k) \in V_j \quad (11.8)$$

i.e., any function in V_j translated by $2^{-j} k$ is still in V_j .

The significance of this set of nested scale spaces V_j ($j \in \mathbb{Z}$) is that any given function $x(t) \in L^2(\mathbb{R})$ can be approximated in any one of these spaces V_j with different resolutions or levels of details, and the greater j , the better the approximation. Due to the dyadic scaling, the resolution of V_{j+1} is twice that of V_j . We further consider the following two cases with $j > 0$:

- Space V_j is spanned by basis $\theta(2^j t)$ which is 2^j times narrower than $\theta(t)$ that spans V_0 , it is therefore capable of representing smaller scale or more detailed information in a signal $x(t)$, i.e., $V_0 \subset V_j$. In particular, when $j \rightarrow \infty$, a basis function of V_∞ is maximally compressed to become an impulse function and the space they span becomes the entire $L^2(\mathbb{R})$ in which any details in a signal can be represented as (Eq.1.6):

$$\int_{-\infty}^{\infty} x(\tau) \delta(t - \tau) d\tau = x(t) \in L^2(\mathbb{R}) \quad (11.9)$$

- Space V_{-j} is spanned by basis $\theta(2^{-j} t)$ which is 2^j times wider than $\theta(t)$ that spans V_0 , it can therefore only represent larger scale or less detailed information in a signal $x(t)$, i.e., $V_{-j} \subset V_0$. In particular, when $j \rightarrow \infty$, the basis function of $V_{-\infty}$ is expanded to have an infinite width but zero height, a constant 0 for all t , and the corresponding space becomes $\{0\}$, containing 0 as its only member.

Based on the Riesz basis $\theta(t) \in V_0$, a *father wavelet* $\phi(t)$ can be constructed in frequency domain as:

$$\Phi(f) = \mathcal{F}[\phi(t)] = \frac{\Theta(f)}{\left[\sum_k |\Theta(f - k)|^2\right]^{1/2}}, \quad (k \in \mathbb{Z}) \quad (11.10)$$

where $\Theta(f) = \mathcal{F}[\theta(t)]$ is the Fourier spectrum of $\theta(t)$.

Now we show that the father wavelet so defined is orthogonal to itself shifted by any integer amount k , i.e.,

$$\langle \phi(t - k), \phi(t) \rangle = \int_{-\infty}^{\infty} \phi(t - k) \overline{\phi}(t) dt = \delta[k], \quad (k \in \mathbb{Z}) \quad (11.11)$$

As the inner product is actually the autocorrelation of $\phi(t)$ evaluated at $t = k$, the equation above can be expressed as the product of the autocorrelation $r_\phi(\tau)$ and an impulse train with unity interval:

$$\int_{-\infty}^{\infty} \phi(t - \tau) \overline{\phi}(t) dt|_{\tau=k \in \mathbb{Z}} = r_\phi(\tau)|_{\tau=k \in \mathbb{Z}} = r_\phi(\tau) \sum_{k \in \mathbb{Z}} \delta(\tau - k) = \delta[k] \quad (11.12)$$

This product in time domain corresponds to a convolution in frequency domain:

$$|\Phi(f)|^2 * \sum_{k \in \mathbb{Z}} \delta(f - k) = \sum_{k \in \mathbb{Z}} |\Phi(f - k)|^2 = 1 \quad (11.13)$$

where $|\Phi(f)|^2 = \mathcal{F}[r_\phi(t)]$ (Eq.3.110) and $\sum_{k \in \mathbb{Z}} \delta(f - k) = \mathcal{F}[\sum_{k \in \mathbb{Z}} \delta(\tau - k)]$ (Eq.3.163). We see that this equation is indeed satisfied by $\Phi(f)$ constructed in Eq.11.10, and consequently the orthogonality of Eq.11.11 in time domain is also satisfied. Now the father wavelet $\phi(t) = \mathcal{F}^{-1}[\Phi(f)]$ can now be used to form an orthogonal basis to span V_0 :

$$V_0 = \text{span}(\phi(t - k), \quad k \in \mathbb{Z}) \quad (11.14)$$

The result in Eq.11.11 for V_0 can be generalized to any V_j by replacing t by $2^j t$ in the equation to get:

$$\begin{aligned} \int_{-\infty}^{\infty} \phi(2^j t - k) \bar{\phi}(2^j t) d(2^j t) &= \int_{-\infty}^{\infty} \sqrt{2^j} \phi(2^j t - k) \sqrt{2^j} \bar{\phi}(2^j t) dt \\ &= \langle \phi_{j,k}(t), \phi_{j,0}(t) \rangle = \delta[k] \end{aligned} \quad (11.15)$$

where $\phi_{j,k}(t)$ is a set of *scaling functions* defined as:

$$\phi_{j,k}(t) = \sqrt{2^j} \phi(2^j t - k) = 2^{j/2} \phi(2^j t - k) \in V_j, \quad k \in \mathbb{Z} \quad (11.16)$$

which can be used as an orthogonal basis to span V_j :

$$V_j = \text{span}(\phi_{j,k}(t), \quad k \in \mathbb{Z}) \quad (11.17)$$

In particular when $j = 0$, $\phi_{0,k}(t) = \phi(t - k)$ and the expression above becomes Eq.11.14. Now any $x(t) \in V_j$ can be represented in terms of the scaling functions as:

$$x(t) = \sum_{k \in \mathbb{Z}} \langle x(t), \phi_{j,k}(t) \rangle \phi_{j,k} \quad (11.18)$$

The scaling functions $\phi_{j,k}(t)$ in space V_j are also related to those in other levels. Specifically, $\phi(t) \in V_0 \subset V_1$ can be expressed in terms of the orthogonal basis $\phi_{1,k}(t) = \sqrt{2}\phi(2t - k) \in V_1$:

$$\phi(t) = \sum_{k \in \mathbb{Z}} h_0[k] \phi_{1,k}(t) = \sqrt{2} \sum_{k \in \mathbb{Z}} h_0[k] \phi(2t - k) \quad (11.19)$$

where the coefficients $h_0[k]$ can be found as the projection of $\phi(t)$ onto the k th basis function $\phi_{1,k}(t) = \sqrt{2}\phi(2t - k)$:

$$h_0[k] = \langle \phi(t), \sqrt{2}\phi(2t - k) \rangle = \sqrt{2} \int_{-\infty}^{\infty} \phi(t) \bar{\phi}(2t - k) dt \quad (11.20)$$

The relationship between V_0 and V_1 can be further generalized to V_j and V_{j+1} . Replacing t by $2^j t - l$ in Eq.11.19, we get:

$$\phi(2^j t - l) = \sqrt{2} \sum_{k \in \mathbb{Z}} h_0[k] \phi(2(2^j t - l) - k) = \sqrt{2} \sum_{k \in \mathbb{Z}} h_0[k] \phi(2^{j+1} t - (2l + k)) \quad (11.21)$$

But due to Eq.11.16, the above can be written as:

$$\phi_{j,l}(t) = \phi(2^j t - l) = \sum_{k \in \mathbb{Z}} h_0[k] \phi_{j+1,2l+k}(t) = \sum_{k' \in \mathbb{Z}} h_0[k' - 2l] \phi_{j+1,k'}(t) \quad (11.22)$$

where we have assumed $k' = 2l + k$. Comparing this equation with a discrete convolution $y[l] = h[l] * x[l] = \sum_k h[l - k]x[k]$, we see that it can be considered as a convolution under two conditions: (1) the coefficients are time reversed and (2) the output is downsampled. In other words, the equation actually describes a discrete FIR filter with $h_0[k]$ as its impulse response, called *scaling filter*, followed by a downsampler. As the resolution of the output $\phi_{j,l}(t) \in V_j$ is lower than that of the input $\phi_{j+1,k'}(t) \in V_{j+1}$, this scaling filter is a low-pass filter.

This filtering process can also be described in frequency domain. Taking the Fourier transform of Eq.11.19, we get

$$\begin{aligned}
 \Phi(f) &= \int_{-\infty}^{\infty} \phi(t)e^{-j2\pi ft} dt = \sqrt{2} \sum_{k \in \mathbb{Z}} h_0[k] \int_{-\infty}^{\infty} \phi(2t - k)e^{-j2\pi ft} dt \\
 &= \sqrt{2} \sum_{k \in \mathbb{Z}} h_0[k] \int_{-\infty}^{\infty} \phi(t')e^{-j2\pi f(t'+k)/2} d\left(\frac{t'}{2}\right) \\
 &= \frac{1}{\sqrt{2}} \sum_{k \in \mathbb{Z}} h_0[k] e^{-jk\pi f} \int_{-\infty}^{\infty} \phi(t')e^{-j2\pi ft'/2} dt' \\
 &= \frac{1}{\sqrt{2}} H_0\left(\frac{f}{2}\right) \Phi\left(\frac{f}{2}\right)
 \end{aligned} \tag{11.23}$$

where $t' = 2t - k$, and $H_0(f)$ is the DTFT spectrum of the discrete impulse response $h_0[k]$, i.e., the frequency response function of the scaling filter:

$$H_0(f) = \mathcal{F}[h_0[k]] = \sum_{k \in \mathbb{Z}} h_0[k] e^{-j2k\pi f} \tag{11.24}$$

Note that as the time gap between neighboring samples of $h_0[k]$ is $t_0 = 1$ (sampling frequency $F = 1/t_0 = 1$, $H_0(f \pm 1) = H_0(f)$ is periodic with period of $F = 1$, i.e., $H(f \pm 1) = H(f)$ and $H_0(f + 1/2) = H_0(f - 1/2)$.

Eq.11.23 can be further expanded recursively:

$$\begin{aligned}
 \Phi(f) &= \frac{1}{\sqrt{2}} H_0\left(\frac{f}{2}\right) \left[\frac{1}{\sqrt{2}} H_0\left(\frac{f}{4}\right) \Phi\left(\frac{f}{4}\right) \right] = \dots \\
 &= \prod_{j=1}^{\infty} \frac{1}{\sqrt{2}} H_0\left(\frac{f}{2^j}\right) \Phi(0) = \prod_{j=1}^{\infty} \frac{1}{\sqrt{2}} H_0\left(\frac{f}{2^j}\right)
 \end{aligned} \tag{11.25}$$

The last equal sign is based on the assumption that $\phi(t)$ is normalized, i.e., its DC component is 1:

$$\Phi(0) = \int_{-\infty}^{\infty} \phi(t)e^{-j2\pi ft} dt|_{f=0} = \int_{-\infty}^{\infty} \phi(t)dt = 1 \tag{11.26}$$

The summation index in the discussion above always takes values in the set of integers, e.g., $k \in \mathbb{Z}$. For simplicity, In the following we will only specify the summation index without explicitly showing the limits.

Example 11.1: Consider a father function defined as:

$$\phi(t) = \begin{cases} 1 & 0 < t < 1 \\ 0 & \text{else} \end{cases} \tag{11.27}$$

This is a square impulse which is indeed orthogonal to itself translated by any integer k :

$$\langle \phi(t), \phi(t - k) \rangle = \int_{-\infty}^{\infty} \phi(t)\phi(t - k)dt = \delta[k], \quad k \in \mathbb{Z} \tag{11.28}$$

Based on this father function, we can construct a set of scaling functions $\phi_{0,k}(t)$ that spans V_0 . Any function $x(t) \in L^2(\mathbb{R})$ can be approximated in V_0 :

$$x(t) \approx \sum_k c_k \phi_{0,k}(t) = \sum_k c_k \phi(t - k) \quad (11.29)$$

Replacing t in $\phi_{0,k}(t) = \phi(t - k)$ by $2^j t$ and including a normalization factor $2^{j/2}$, we get another set of orthonormal functions:

$$\phi_{j,k}(t) = 2^{j/2} \phi(2^j t - k), \quad k \in \mathbb{Z} \quad (11.30)$$

As $\phi(t) = 1$ when its argument satisfies $0 < t < 1$, we also have $\phi(2^j t - k) = 1$ when $0 < 2^j t - k < 1$, i.e.,

$$\frac{k}{2^j} < t < \frac{k}{2^j} + \frac{1}{2^j} \quad (11.31)$$

We see that $\phi_{j,k}(t)$ is a rectangular impulse of height $2^{j/2} = \sqrt{2^j}$ and width $1/2^j$, and it is shifted k times its width. Obviously these functions are also orthonormal and they span space V_j :

$$\langle \phi_{j,k}(t), \phi_{j,l}(t) \rangle = \delta[k - l], \quad k, l \in \mathbb{Z} \quad (11.32)$$

The basic ideas above are illustrated in Fig.11.2. The first two panels show two scaling functions $\phi(t) = \phi_{0,0}(t)$ and $\phi_{0,1}(t) = \phi(t - 1)$ both in V_0 , the next two panels show another two scaling functions $\phi_{1,0}(t) = \sqrt{2}\phi(2t)$ and $\phi_{1,1}(t) = \sqrt{2}\phi(2t - 1)$ in V_1 . Panel 5 shows a function $x(t) \in V_1$ represented as a linear combination of the scaling functions $\phi_{1,k}(t)$:

$$x(t) = 0.5\phi_{1,0}(t) + \phi_{1,1}(t) - 0.25\phi_{1,4}(t) \quad (11.33)$$

Finally panel 6 shows that a scaling function $\phi_{0,0}(t) \in V_0$ represented as a linear combination of the basis functions $\phi_{1,k}(t) \in V_1$ (Eq.11.22):

$$\phi_{0,l}(t) = h_0[0]\phi_{1,2l}(t) + h_0[1]\phi_{1,2l+1}(t) = \frac{1}{\sqrt{2}}\phi_{1,2k}(t) + \frac{1}{\sqrt{2}}\phi_{1,2k+1}(t) \quad (11.34)$$

where the coefficients $h_0[0] = h_0[1] = 1/\sqrt{2}$ are obtained according to Eq.11.20.

Generally the ideas illustrated in this example are valid if the square impulses are replaced by any family of functions with compact support , i.e., the functions are non-zero only over a finite duration.

11.1.2 Wavelet Spaces

Previously we constructed a sequence of nested scale spaces $V_j \subset V_{j+1}$ in which a given function $x(t) \in L^2(\mathbb{R})$ can be approximated at different scale levels, i.e., the approximation in V_{j+1} contains more detailed information in the signal than that in V_j . In other words, certain functions in V_{j+1} but not representable in V_j

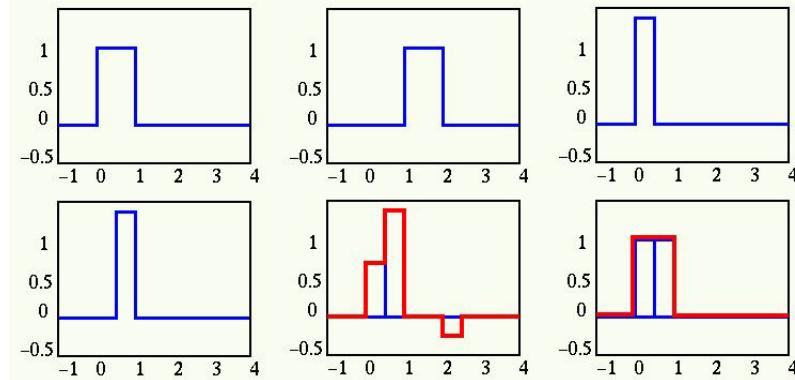


Figure 11.2 The basis functions that span scale spaces and some functions they represent

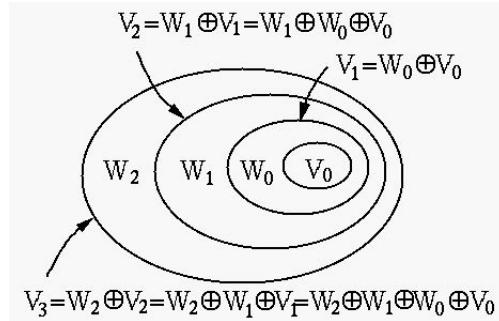


Figure 11.3 The nested V_j and W_j spaces for MRA

are contained in the difference space $W_j = V_{j+1} - V_j$, called the *wavelet space*. As $W_j \subset V_{j+1}$, $V_j \subset V_{j+1}$, and $W_j \cap V_j = \{0\}$, W_j is the complementary space of V_j , i.e., V_{j+1} is the direct sum of V_j and W_j , and this relationship can be carried out recursively:

$$V_{j+1} = W_j \oplus V_j = W_j \oplus W_{j-1} \oplus V_{j-1} = \dots \quad (11.35)$$

This result indicates that the approximation of $x(t)$ in V_j can be improved by including more detailed information in W_j , so that $x(t)$ is now approximated in $V_{j+1} = W_j \oplus V_j$ of higher resolution. As can be seen from Fig.11.3, this improvement can be continued if we start at an arbitrary initial level such as V_0 , and keep including more detailed information contained in W_j when $j \rightarrow \infty$, so that the signal can be ever more precisely approximated:

$$[\bigoplus_{j=0}^{\infty} W_j] \oplus V_0 = L^2(\mathbb{R}) \quad (11.36)$$

Same as the scale space V_0 that is spanned by a set of orthogonal scaling functions $\phi_{0,k} = \phi(t - k)$ derived from a father wavelet $\phi(t)$, here we assume

the wavelet space W_0 is also spanned by a set of orthogonal wavelet functions $\psi(t - k)$ that are derived from a *mother wavelet* $\psi(t)$. Similar to Eq.11.11, these wavelet functions are required to be orthogonal to themselves shifted by any integer amount:

$$\langle \psi(t - k), \psi(t) \rangle = \int_{-\infty}^{\infty} \psi(t - k) \overline{\psi}(t) dt = \delta[k], \quad k \in \mathbb{Z} \quad (11.37)$$

and they span the space W_0 :

$$W_0 = \text{span}(\psi(t - k), \quad k \in \mathbb{Z}) \quad (11.38)$$

Moreover, the mother and father wavelets are required to be orthogonal to each other with any integer shift:

$$\langle \phi(t - k), \psi(t) \rangle = \int_{-\infty}^{\infty} \phi(t - k) \overline{\psi}(t) dt = 0, \quad k \in \mathbb{Z} \quad (11.39)$$

Following the same process for the derivation of Eq.11.13 from Eq.11.11 for the scaling functions, we can also represent the orthogonalities of Eqs.11.37 and 11.39 in frequency domain as:

$$\sum_k |\Psi(f - k)|^2 = 1 \quad (11.40)$$

$$\sum_k \Phi(f - k) \overline{\Psi}(f - k) = 0 \quad (11.41)$$

This result in W_0 can be generalized to space W_j . Replacing t by $2^j t$ in Eq.11.37 we get:

$$\begin{aligned} \int_{-\infty}^{\infty} \psi(2^j t - k) \overline{\psi}(2^j t) d(2^j t) &= \int_{-\infty}^{\infty} \sqrt{2^j} \psi(2^j t - k) \sqrt{2^j} \overline{\psi}(2^j t) dt \\ &= \langle \psi_{j,k}(t), \psi_{j,0}(t) \rangle = \delta[k] \end{aligned} \quad (11.42)$$

where we have defined a set of orthogonal *wave functions* $\psi_{j,k}(t)$ as:

$$\psi_{j,k}(t) = \sqrt{2^j} \psi(2^j t - k) = 2^{j/2} \psi(2^j t - k) \in W_j, \quad k \in \mathbb{Z} \quad (11.43)$$

which can be used as an orthogonal basis to span W_j :

$$W_j = \text{span}(\psi_{j,k}(t), \quad k \in \mathbb{Z}) \quad (11.44)$$

Moreover, these wavelet functions $\psi_{j,k}(t)$ are further required to be orthogonal to the scaling functions $\phi_{j,k}(t)$ as well as themselves:

$$\langle \psi_{j,k}(t), \psi_{i,l}(t) \rangle = \delta[i - j] \delta[k - l] \quad (11.45)$$

$$\langle \phi_{j,k}(t), \psi_{j,l}(t) \rangle = 0 \quad (11.46)$$

Consequently, spaces W_j and V_j spanned respectively by $\psi_{j,k}(t)$ and $\phi_{j,l}(t)$ are orthogonal, i.e., $W_j \perp V_j$. Moreover, as $V_j = W_{j-1} \oplus V_{j-1}$, it follows that $W_j \perp V_{j-1}$ and $W_j \perp W_{j-1}$, i.e., the wavelet functions $\psi_{j,k}(t)$ are orthogonal with respect to j for different scale levels as well as to k for different integer translations in each

scale level. Further more, since all wavelet spaces W_j are spanned by $\psi_{j,k}(t)$, the entire function space $L^2(\mathbb{R}) = \bigoplus_j W_j$ is also spanned by these orthogonal wavelet functions:

$$L^2(\mathbb{R}) = \text{span}(\psi_{j,k}(t), (j, k \in \mathbb{Z})) \quad (11.47)$$

Similar to the representation of the father wavelet $\phi(t) \in V_0 \in V_1$ in Eq.11.19, the mother wavelet $\psi(t) \in V_0 \in V_1$ can also be expressed as a linear combination of the basis $\phi_{1,k}(t) = \sqrt{2}\phi(2t - k)$ in V_1 :

$$\psi(t) = \sum_k h_1[k] \phi_{1,k}(t) = \sqrt{2} \sum_k h_1[k] \phi(2t - k) \quad (11.48)$$

where the coefficients $h_1[k]$ can be found as the projection of $\psi(t)$ onto the k th basis function $\psi_{1,k}(t)$. These coefficients $h_1[k]$ must be related in some way to the coefficients $h_0[k]$ in order for the mother $\psi(t)$ and father wavelet $\phi(t)$ to be orthogonal as required, as to be discussed later.

We replace t by $2^j t - l$ in the equation above to get:

$$\begin{aligned} \psi(2^j t - l) &= \sqrt{2} \sum_k h_1[k] \phi(2(2^j t - l) - k) = \sqrt{2} \sum_k h_1[k] \phi(2^{j+1} t - (2l + k)) \\ &= \sqrt{2} \sum_{k'} h_1[k' - 2l] \phi(2^{j+1} t - k') \end{aligned} \quad (11.49)$$

where $k' = 2l + k$. Due to Eq.11.16, we have

$$\phi(2^{j+1} t - k) = 2^{-(j+1)/2} \phi_{j+1,k}(t) \quad (11.50)$$

Substituting this into the equation above we get:

$$\psi_{j,l}(t) = 2^{j/2} \psi(2^j t - l) = \sum_k h_1[k - 2l] \phi_{j+1,k}(t) \quad (11.51)$$

Similar to Eq.11.22 for the scaling functions $\phi_{j,l}(t)$, under the two conditions that the coefficients $h_1[k]$ are reversed in time and the output is downsampled, Eq.11.51 also describes an discrete FIR filter, called a *wavelet filter* with $h_1[k]$ as the impulse response, followed by a downampler. The input $\phi_{j+1,k}(t) \in V_{j+1}$ of the wavelet filter is the same as the scaling filter, but the output $\psi_{j,l}(t) \in W_j$ contains the high resolution contents of the input in V_{j+1} not represented by the output $\phi_{j,l}(t) \in V_j$ of the scaling filter, i.e., this wavelet filter is a high-pass filter.

This filtering process can also be described in frequency domain. Taking the Fourier transform on both sides of Eq.11.48 and following the steps in Eq.11.23 for the scaling functions, we get:

$$\begin{aligned} \Psi(f) &= \mathcal{F}[\psi(t)] = \sqrt{2} \sum_k h_1[k] \mathcal{F}[\phi(2t - k)] \\ &= \frac{1}{\sqrt{2}} \sum_k h_1[k] e^{-jk\pi} \Phi\left(\frac{f}{2}\right) = \frac{1}{\sqrt{2}} H_1\left(\frac{f}{2}\right) \Phi\left(\frac{f}{2}\right) \end{aligned} \quad (11.52)$$

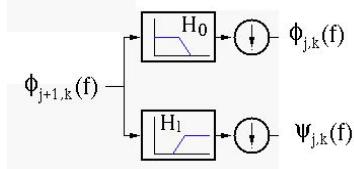


Figure 11.4 Scaling and wavelet filters in frequency domain

where $H_1(f) = \mathcal{F}[h_1[k]]$ is the frequency response function of the wavelet filter:

$$H_1(f) = \sum_k h_1[k] e^{-j2k\pi f} \quad (11.53)$$

Note again $H_1(f \pm 1) = H_1(f)$ is periodic with period 1 and $H_1(f + 1/2) = H_1(f - 1/2)$.

Same as in Eq.11.25, the wavelet filter can also be recursively expanded to become:

$$\Psi(f) = \frac{1}{\sqrt{2}} H_1\left(\frac{f}{2}\right) \prod_{j=2}^{\infty} \frac{1}{\sqrt{2}} H_0\left(\frac{f}{2^j}\right) \quad (11.54)$$

In order to satisfy the admissibility condition (Eq.10.21), the DC component of the wavelet $\psi(t)$ is required to be zero (Eq.10.10):

$$\Psi(0) = \int_{-\infty}^{\infty} \psi(t) e^{-j2\pi ft} dt|_{f=0} = \int_{-\infty}^{\infty} \psi(t) dt = 0 \quad (11.55)$$

The low-pass scaling filter and the high-pass wavelet filter followed by a down-sampler described respectively in Eqs.11.22 and 11.51 are illustrated in frequency domain in Fig.11.4, where the input $\Phi_{j+1,k}(f)$ is filtered by the scale and wavelet filters and then downsampled (denoted by the down-arrow) to produce $\Phi_{j,k}(f)$ and $\Psi_{j,k}(f)$, respectively. Moreover, this filtering-downsampling process can be further carried out recursively when the output $\phi_{j,k}(t)$ of the scaling filter is taken as the input of the scale and wavelet filters of the next level to produce $\Phi_{j-1,k}(t)$ and $\Psi_{j-1,k}(f)$, as shown on the left of Fig.11.12, to be considered later.

11.1.3 Properties of the Scaling and Wavelet Filters

Here we consider a set of properties required of the low-pass scaling filter and high-pass wavelet filter. Specifically the coefficients $h_0[k]$ and $h_1[k]$, or the frequency response functions $H_0(f)$ and $H_1(f)$, of these filters have to satisfy a set of conditions for their outputs, the scaling and wavelet functions $\phi(t)$ and $\psi(t)$ to be orthogonal as discussed previously. These required properties of the scaling and wavelet filters will be used in the design of these filters by which the wavelet transform is actually carried out.

1. Normalization in time domain:

$$\frac{1}{\sqrt{2}} \sum_k h_0[k] = 1 \quad (11.56)$$

We integrate both sides of Eq.11.19 with respect to t to get

$$\int_{-\infty}^{\infty} \phi(t) dt = \sqrt{2} \sum_k h_0[k] \int_{-\infty}^{\infty} \phi(2t - k) dt = \sum_k h_0[k] \frac{1}{\sqrt{2}} \int_{-\infty}^{\infty} \phi(t') dt' \quad (11.57)$$

where we have assumed $t' = 2t - k$, i.e., $t = (t' - k)/2$. Dividing both sides by $\int_{-\infty}^{\infty} \phi(t) dt \neq 0$, we get Eq.11.56.

2. Normalization in frequency domain:

$$H_0(0) = \sqrt{2}, \quad H_1(0) = 0 \quad (11.58)$$

These can easily obtained by letting $f = 0$ in Eqs.11.23 and 11.52, and noting $\Phi(0) = 1$ (Eq.11.26) and $\Psi(0) = 0$ (Eq.11.55). Equivalently, we have

$$\sum_k h_1[k] = 0, \quad \sum_k h_0[k] = \sqrt{2} \quad (11.59)$$

which can be also be easily shown by letting $f = 0$ in Eqs.11.24 and 11.53 and applying the results $H_0(0) = \sqrt{2}$ and $H_1(0) = 0$ above.

3. Orthogonalities of scaling and wavelet functions (time domain):

Previously we considered the required orthogonalities of the scaling functions (Eq.11.15), the wavelet functions (Eq.11.42), and between the scaling and wavelet functions (Eq.11.46). Now we show that these orthogonalities can also be represented in terms of the scaling and wavelet filters $h_0[k]$ and $h_1[k]$.

$$\begin{aligned} \sum_k h_0[k] \bar{h}_0[k - 2n] &= \delta[n] \\ \sum_k h_1[k] \bar{h}_1[k - 2n] &= \delta[n] \\ \sum_k h_0[k] \bar{h}_1[k - 2n] &= 0 \end{aligned} \quad (11.60)$$

In particular, when $n = 0$, we have

$$\sum_k |h_0[k]|^2 = 1, \quad \sum_k |h_1[k]|^2 = 1 \quad (11.61)$$

Proof: Substituting Eq.11.22 into Eq.11.15 (and replacing k by l), we get

$$\begin{aligned} \delta[l] &= \langle \phi_{j,l}(t), \phi_{j,0}(t) \rangle = \int_{-\infty}^{\infty} \phi_{j,l}(t) \bar{\phi}_{j,0}(t) dt \\ &= \sum_k \sum_{k'} h_0[k - 2l] \bar{h}_0[k'] \int_{-\infty}^{\infty} \phi_{j+1,k}(t) \bar{\phi}_{j+1,k'}(t) dt \\ &= \sum_k \sum_{k'} h_0[k - 2l] \bar{h}_0[k] \delta[k - k'] = \sum_k h_0[k - 2l] \bar{h}_0[k] \end{aligned} \quad (11.62)$$

In the same manner, we can also prove the second equation in Eq.11.60 for $h_1[k]$ by substituting Eq.11.51 into Eq.11.42, and the third equation for both $h_0[k]$ and $h_1[k]$ by substituting both Eqs.11.22 and 11.51 into Eq.11.46.

4. Orthogonalities of scaling and wavelet functions (frequency domain):

Previously we considered the orthogonalities of the scaling functions (Eqs.11.11 and 11.13), of the wavelet functions (Eq.s11.37 and 11.40), and between the scaling and wavelet functions (Eqs.11.39 and 11.41). Now we further show that these orthogonalities can also be represented in terms of the scaling and wavelet filters $H_0(f)$ and $H_1(f)$ in frequency domain.

$$\begin{aligned} |H_0(f)|^2 + |H_0(f + \frac{1}{2})|^2 &= 2 \\ |H_1(f)|^2 + |H_1(f + \frac{1}{2})|^2 &= 2 \\ H_0(f)\overline{H}_1(f) + H_0(f + \frac{1}{2})\overline{H}_1(f + \frac{1}{2}) &= 0 \end{aligned} \quad (11.63)$$

Proof:

Substituting Eq.11.23 into Eq.11.13, we get

$$\sum_k \left| H_0\left(\frac{f-k}{2}\right) \right|^2 \left| \Phi\left(\frac{f-k}{2}\right) \right|^2 = 2 \quad (11.64)$$

We then separate the even and odd terms in the summation to get:

$$\begin{aligned} &\sum_k \left| H_0\left(\frac{f-2k}{2}\right) \right|^2 \left| \Phi\left(\frac{f-2k}{2}\right) \right|^2 \\ &+ \sum_k \left| H_0\left(\frac{f-(2k+1)}{2}\right) \right|^2 \left| \Phi\left(\frac{f-(2k+1)}{2}\right) \right|^2 = 2 \end{aligned} \quad (11.65)$$

But as $H_0(f \pm k) = H_0(f)$ is periodic and due to Eq.11.13, the above can be written as

$$\begin{aligned} &\left| H_0\left(\frac{f}{2}\right) \right|^2 \sum_k \left| \Phi\left(\frac{f}{2} - k\right) \right|^2 + \left| H_0\left(\frac{f+1}{2}\right) \right|^2 \sum_k \left| \Phi\left(\frac{f+1}{2} - k\right) \right|^2 \\ &= \left| H_0\left(\frac{f}{2}\right) \right|^2 + \left| H_0\left(\frac{f}{2} + \frac{1}{2}\right) \right|^2 = 2 \end{aligned} \quad (11.66)$$

Replacing $f/2$ by f , we complete the proof.

The second equation in Eq.11.63 for $H_1(f)$ can be proven in the same way by substituting Eq.11.52 into Eq.11.40.

Now we prove the third equation in Eq.11.63 involving both $H_0(f)$ and $H_1(f)$. Substituting Eqs.11.23 and 11.52 into Eq.11.41, we get:

$$\begin{aligned} & \sum_k H_0\left(\frac{f-k}{2}\right) \Phi\left(\frac{f-k}{2}\right) \overline{H}_1\left(\frac{f-k}{2}\right) \overline{\Phi}\left(\frac{f-k}{2}\right) \\ &= \sum_k H_0\left(\frac{f-k}{2}\right) \overline{H}_1\left(\frac{f-k}{2}\right) \left| \Phi\left(\frac{f-k}{2}\right) \right|^2 = 0 \end{aligned} \quad (11.67)$$

We then separate the even and odd terms in the summation to get:

$$\begin{aligned} & \sum_k H_0\left(\frac{f-2k}{2}\right) \overline{H}_1\left(\frac{f-2k}{2}\right) \left| \Phi\left(\frac{f-2k}{2}\right) \right|^2 \\ &+ \sum_k H_0\left(\frac{f-(2k+1)}{2}\right) \overline{H}_1\left(\frac{f-(2k+1)}{2}\right) \left| \Phi\left(\frac{f-(2k+1)}{2}\right) \right|^2 = 0 \end{aligned} \quad (11.68)$$

Replacing $f/2$ by f' and noting that $H_i(f \pm k) = H_i(f)$ ($i = 1, 2$), we get:

$$\begin{aligned} & H_0(f') \overline{H}_1(f') \sum_k |\Phi(f' - k)|^2 \\ &+ H_0(f' - \frac{1}{2}) \overline{H}_1(f' - \frac{1}{2}) \sum_k \left| \Phi(f' - k - \frac{1}{2}) \right|^2 = 0 \end{aligned} \quad (11.69)$$

The proof is complete by realizing that both summations are equal to 1 (Eq.11.13).

In our discussion above the discrete scaling and wavelet filters are represented in frequency domain by their DTFT $H_0(f)$ and $H_1(f)$ (Eqs.11.24 and 11.53), respectively. Alternatively, these filters can also be represented in Z-domain as:

$$H_0(z) = \sum_k h_0[k] z^{-k}, \quad H_1(z) = \sum_k h_1[k] z^{-k} \quad (11.70)$$

which are also used in many wavelet literatures. When $H_0(z)$ and $H_1(z)$ are evaluated along the unit circle $|z| = 1$, i.e., $z = e^{j2\pi f}$, they become the same as $H_0(f)$ and $H_1(f)$. In particular, corresponding to $f = 0$ and $f + 1/2$, we have respectively $z = e^0 = 1$ and $e^{j2\pi(f+1/2)} = -e^{j2\pi f} = -z$. Now the normalization and orthogonality properties considered above can also be represented in Z-domain as:

$$H_0(1) = \sqrt{2}, \quad H_1(1) = 0 \quad (11.71)$$

$$|H_0(z)|^2 + |H_0(-z)|^2 = 2 \quad (11.72)$$

$$|H_1(z)|^2 + |H_1(-z)|^2 = 2 \quad (11.73)$$

$$H_0(z) \overline{H}_1(z) + H_0(-z) \overline{H}_1(-z) = 0 \quad (11.74)$$

11.1.4 Relationship Between Scaling and Wavelet Filters

We now show the scaling filter $H_0(f)$ and the wavelet filter $H_1(f)$ can be related by

$$H_1(f) = e^{-j2\pi f} \overline{H}_0(f - \frac{1}{2}), \quad \text{i.e.} \quad H_0(f) = e^{j2\pi f} \overline{H}_1(f - \frac{1}{2}) \quad (11.75)$$

We can easily verify that all required conditions in Eq.11.63 are satisfied by $H_0(f)$ and $H_1(f)$ related in Eq.11.75, i.e., the scaling and wavelet functions generated by the filters $H_0(f)$ and $H_1(f)$ related by Eq.11.75 are indeed orthogonal to themselves with integer translation, and they are also orthogonal to each other with integer translation and across different scale levels. First, given $H_0(f)$ (or $H_1(f)$) that satisfies the first (or second) equation in Eq.11.63, the corresponding $H_1(f)$ (or $H_0(f)$) given in Eq.11.75 will satisfy the second (or first) one. Second, substituting $H_1(f)$ in Eq.11.75 into the third equation in Eq.11.63 we see that it holds indeed:

$$\begin{aligned} & H_0(f)e^{j2\pi f}H_0(f - \frac{1}{2}) + H_0(f - \frac{1}{2})e^{j2\pi(f+1/2)}H_0(f) \\ &= H_0(f)e^{j2\pi f}H_0(f - \frac{1}{2}) - H_0(f - \frac{1}{2})e^{j2\pi f}H_0(f) = 0 \end{aligned} \quad (11.76)$$

This relationship in Eq.11.75 between $H_0(f)$ and $H_1(f)$ in frequency domain can be converted into time domain by taking the inverse Fourier transform on both sides of the equation and applying the time shift, modulation and complex conjugate properties of the DTFT (Eqs.4.32, 4.44 and 4.28):

$$h_1[k] = \mathcal{F}^{-1}[-e^{-j2\pi f} \overline{H}_0(f - \frac{1}{2})] = (-1)^k \overline{h}_0[1 - k] \quad (11.77)$$

The actual wavelet function $\psi(t) \in V_0$ can therefore be obtained by substituting these coefficients into Eq.11.48:

$$\psi(t) = \sqrt{2} \sum_k h_1[k] \phi(2t - k) = \sqrt{2} \sum_k (-1)^k \overline{h}_0[1 - k] \phi(2t - k) \quad (11.78)$$

We can verify that this wavelet function $\psi(t)$ is indeed orthogonal to its integer translations $\psi(t - l)$ for all $l \in \mathbb{Z}$, i.e., $\langle \psi(t - l), \psi(t) \rangle = \delta[l]$:

$$\begin{aligned}
& \langle \psi(t - l), \psi(t) \rangle = \int_{-\infty}^{\infty} \psi(t - l) \bar{\psi}(t) dt \\
&= 2 \sum_{k'} \sum_k (-1)^{k+k'} \bar{h}_0[1-k] h_0[1-k'] \int_{-\infty}^{\infty} \phi(2(t-l) - k') \bar{\phi}(2t - k) dt \\
&= 2 \sum_k \sum_m (-1)^{m+k} \bar{h}_0[1-k] h_0[1-m+2l] \int_{-\infty}^{\infty} \phi(2t-m) \bar{\phi}(2t-k) dt \\
&\quad (\text{where } m = 2l + k') \\
&= \sum_k \sum_m (-1)^{m+k} \bar{h}_0[1-k] h_0[1-m+2l] \delta[m-k] \\
&= \sum_k \bar{h}_0[1-k] h_0[1-k+2l] = \delta[l]
\end{aligned} \tag{11.79}$$

Here we have used the fact that $\phi_{1,k}(t)$ are orthonormal (Eq.11.15), and the last equal sign is due to Eq.11.60.

Replacing t by $2^j t - k$, we obtain the wavelet functions $\psi_{j,k}(t) = \psi(2^j t - k)$ that span W_j .

Example 11.2: The scaling function $\phi(t)$ considered in the previous example is a square impulse with unit height and width, and the coefficients are $h_0[0] = h_0[1] = 1/\sqrt{2}$. Now based on Eq. 11.77 the coefficients for the wavelet functions $\psi_{1,k}(t)$ can be obtained as:

$$\begin{aligned}
h_1[0] &= (-1)^0 h_0[1-0] = h_0[1] = 1/\sqrt{2} \\
h_1[1] &= (-1)^1 h_0[1-1] = -h_0[0] = -1/\sqrt{2}
\end{aligned} \tag{11.80}$$

and the wavelet function is:

$$\psi(t) = \sum_l h_1[l] \sqrt{2} \phi[2t-l] = \phi(2t) - \phi(2t-1) = \begin{cases} 1 & 0 \leq t < 1/2 \\ -1 & 1/2 \leq t < 1 \\ 0 & \text{else} \end{cases} \tag{11.81}$$

The first two panels of Fig.11.5 show two of the wavelet functions $\psi(t) = \psi_{0,0}(t)$ and $\psi_{0,2}(t) = \psi(t-2)$ in W_0 , and the 3rd panel shows a wavelet function $\psi_{1,0}(t) = \sqrt{2}\psi(2t)$ in W_1 . The 4th panel shows a function in V_0 spanned by $\phi_{0,k}(t)$, and the 5th panel show a function in W_0 spanned by $\psi_{0,k}(t)$, which cannot be represented in V_0 . The 6th panel shows the sum of these two functions in $V_1 = V_0 \oplus W_0$, which can be represented by $\phi_{1,k}(t)$ spanning V_0 , or, equivalently, by $\phi_{0,k}(t)$ and $\psi_{0,k}(t)$.

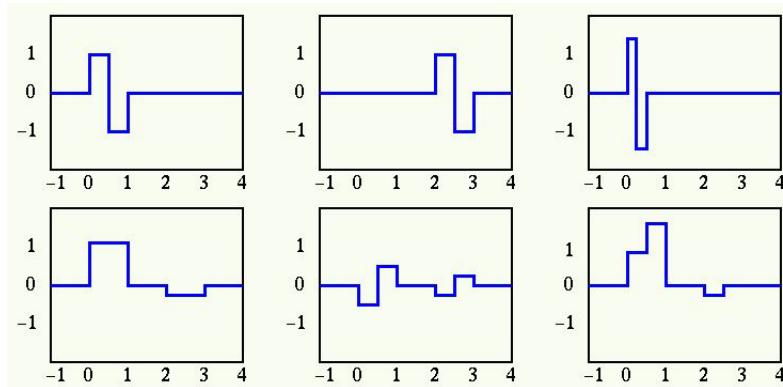


Figure 11.5 Basis functions that span wavelet spaces (top) and some functions they represent

11.1.5 Wavelet Series Expansion

A signal $x(t)$ can be approximated in any scale space V_j spanned by a set of scaling functions $\phi_{j,k}(t)$ as the orthogonal basis. For example, when $j = 0$ the approximation in V_0 is:

$$x(t) \approx \sum_k c_{0,k} \phi_{0,k}(t) = \sum_k c_{0,k} \phi_{0,k}(t) \quad (11.82)$$

where the *approximation coefficients* $c_{0,k}$ can be found as the projections of the signal onto the corresponding basis vector:

$$c_{0,k} = \langle x(t), \phi_{0,k}(t) \rangle = \int x(t) \overline{\phi}_{0,k}(t) dt, \quad (\text{for all } k) \quad (11.83)$$

Moreover, the signal can be ever more precisely approximated if progressively more detailed information contained in wavelet space W_j spanned by $\psi_{j,k}(t)$ is included when $j \rightarrow \infty$ (Eq.11.36):

$$\begin{aligned} x(t) &= \sum_k c_{0,k} \phi_{0,k}(t) + \sum_{j=0}^{\infty} \sum_k d_{j,k} \psi_{j,k}(t) \\ &= \sum_k \langle x(t), \phi_{0,k}(t) \rangle \phi_{0,k}(t) + \sum_{j=0}^{\infty} \sum_k \langle x(t), \psi_{j,k}(t) \rangle \psi_{j,k}(t) \end{aligned} \quad (11.84)$$

where $d_{j,k}$, called the *detail coefficients*, can be found as:

$$d_{j,k} = \langle x(t), \psi_{j,k}(t) \rangle = \int x(t) \overline{\psi}_{j,k}(t) dt, \quad (\text{for all } k \text{ and } j > 0) \quad (11.85)$$

Eq.11.84 is the *wavelet series expansion* of the signal $x(t)$, corresponding to Eq.3.5 of the Fourier series expansion considered in Chapter 3.

Example 11.3:

Here we use the Haar wavelets to approximate the following continuous function $x(t)$ defined over the period $0 \leq t < 1$:

$$x(t) = \begin{cases} t^2 & 0 \leq t < 1 \\ 0 & \text{else} \end{cases} \quad (11.86)$$

First note that each individual space (V_0, W_0, W_1, \dots) is spanned by different number of basis functions. For example, spaces V_0 and W_0 are spanned by only one basis function, but space W_1 is spanned by 2 basis functions, and space W_2 is spanned by 4 (Fig.8.9).

We can choose to start at scale level $j = 0$. According to Eqs.11.83 and 11.85, the approximation and wavelet coefficients can be obtained as:

$$\begin{aligned} c_0(0) &= \int_0^1 t^2 \varphi_{0,0}(t) dt = \int_0^1 t^2(t) dt = \frac{1}{3} \\ d_0(0) &= \int_0^1 t^2 \psi_{0,0}(t) dt = \int_0^{0.5} t^2(t) dt - \int_{0.5}^1 t^2(t) dt = -\frac{1}{4} \\ d_1(0) &= \int_0^1 t^2 \psi_{1,0}(t) dt = \int_0^{0.25} \sqrt{2}t^2(t) dt - \int_{0.25}^{0.5} t^2 \sqrt{2}(t) dt = -\frac{\sqrt{2}}{32} \\ d_1(1) &= \int_0^1 t^2 \psi_{1,1}(t) dt = \int_{0.5}^{0.75} \sqrt{2}t^2(t) dt - \int_{0.75}^1 t^2 \sqrt{2}(t) dt = -\frac{3\sqrt{2}}{32} \end{aligned} \quad (11.87)$$

Therefore the wavelet series expansion of the function $x(t)$ is

$$x(t) = \frac{1}{3}\phi_{0,0}(t) + [-\frac{1}{4}\psi_{0,0}(t)] + [-\frac{\sqrt{2}}{32}\psi_{1,0}(t) - \frac{3\sqrt{2}}{32}\psi_{1,1}(t)] + \dots \quad (11.88)$$

The first two coefficients are for spaces V_0 and W_0 , respectively, while both of the last two are for space W_1 . This process can be carried out further by including progressively more detailed information in wavelet spaces W_2, W_3, \dots, W_j as $j \rightarrow \infty$.

The definition of the multiresolution analysis requires the existence of a Riesz basis (not necessarily an orthogonal basis) that spans space V_0 , i.e., the MRA may be a biorthogonal MRA. In this case, there exists a dual function corresponding to each scaling or wavelet function. Specifically at the j th level of such a biorthogonal MRA, corresponding to the scaling function $\phi_{j,k}(t)$ and wavelet function $\psi_{j,k}(t)$, there exist respectively a dual scaling function $\tilde{\phi}_{j,k}(t)$ and a dual wavelet function $\tilde{\psi}_{j,k}(t)$ so that

$$\begin{aligned} <\phi_{j,k}(t), \tilde{\phi}_{j,l}(t)> &= \delta[k-l] \\ <\psi_{j,k}(t), \tilde{\psi}_{j,l}(t)> &= \delta[i-j]\delta[k-l] \\ <\phi_{j,k}(t), \tilde{\psi}_{j,l}(t)> &= <\psi_{j,k}(t), \tilde{\phi}_{j,l}(t)> = 0 \end{aligned} \quad (11.89)$$

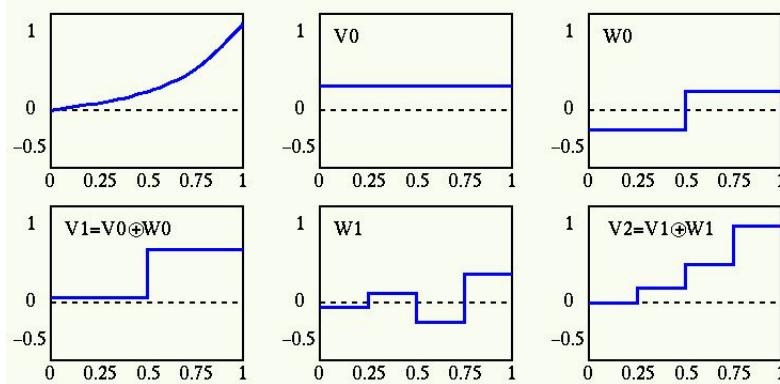


Figure 11.6 Wavelet approximation of a function

Same as $\phi_{j,k}(t)$ and $\psi_{j,k}(t)$ that span respectively V_j and W_j satisfying $V_j \oplus W_j = V_{j+1}$, the dual scaling and wavelet functions $\tilde{\phi}_{j,k}(t)$ and $\tilde{\psi}_{j,k}(t)$ respectively span \tilde{V}_j and \tilde{W}_j satisfying $\tilde{V}_j \oplus \tilde{W}_j = \tilde{V}_{j+1}$. Note, however, as these basis functions are in general not orthogonal, V_j and W_j are not orthogonal complement of each other in V_{j+1} , neither are \tilde{V}_j and \tilde{W}_j in \tilde{V}_{j+1} .

In this case, the wavelet series expansion in Eq.11.84 becomes

$$\begin{aligned} x(t) &= \sum_k \langle x(t), \phi_{0,k}(t) \rangle \tilde{\phi}_{0,k}(t) + \sum_{j=0}^{\infty} \sum_k \langle x(t), \psi_{j,k}(t) \rangle \tilde{\psi}_{j,k}(t) \\ &= \sum_k \langle x(t), \tilde{\phi}_{0,k}(t) \rangle \phi_{0,k}(t) + \sum_{j=0}^{\infty} \sum_k \langle x(t), \tilde{\psi}_{j,k}(t) \rangle \psi_{j,k}(t) \end{aligned} \quad (11.90)$$

11.1.6 Construction of Scaling and Wavelet Functions

To carry out the wavelet transform of a given signal, the scaling function $\phi(t)$ and the wavelet functions $\psi(t)$ need to be specifically determined. In general this is a design process which can be carried out in one of the following ways:

- Specify $\phi(t)$ and $\psi(t)$ in time domain;
- Specify their spectra $\Phi(f)$ and $\Psi(f)$ in frequency domain;
- Specify the corresponding filter coefficients $h_0[k]$ and $h_1[k]$ in time domain;
- Specify the corresponding filter frequency response functions $H_0(f)$ and $H_1(f)$ in frequency domain.

In the following we will consider these different methods. Keep in mind that it is desirable for the scaling and wavelet functions to have good localities in both time and frequency domains. Ideally they should be *compactly supported*, i.e., they are non-zero only within a finite domain.

- **Haar wavelets**

The scaling and wavelet functions can be constructed by the following steps:

1. Choose the scaling function $\phi(t)$ satisfying Eq.11.11:

$$\langle \phi(t - k), \phi(t) \rangle = \delta[k]$$

or $\Phi(f)$ satisfying Eq.11.13:

$$\sum_{k \in \mathbb{Z}} |\Phi(f - k)|^2 = 1$$

For the Haar transform, we simply choose the scaling function as:

$$\phi(t) = \begin{cases} 1 & 0 \leq t < 1 \\ 0 & \text{else} \end{cases} \quad (11.91)$$

2. Find scaling coefficients $h_0[k]$ based on Eq.11.20:

$$h_0[k] = \langle \phi(t), \sqrt{2}\phi(2t - k) \rangle$$

or $H_0(f)$ according to Eq.11.23:

$$H_0(f) = \sqrt{2} \frac{\Phi(2f)}{\Phi(f)}$$

For the Haar transform, we have

$$\begin{aligned} h_0[k] &= \sqrt{2} \int_{-\infty}^{\infty} \phi(t)\phi(2t - k)dt = \sqrt{2} \int_0^1 \phi(2t - k)dt \\ &= \frac{1}{\sqrt{2}} \int_0^2 \phi(t' - k)dt' = \frac{1}{\sqrt{2}} \begin{cases} 1 & k = 0, 1 \\ 0 & \text{else} \end{cases} \end{aligned} \quad (11.92)$$

3. Find wavelet coefficients $h_1[k]$ according to Eq.11.77

$$h_1[k] = (-1)^k \bar{h}_0[1 - k]$$

or $H_1(f)$ according to Eq.11.75

$$H_1(f) = e^{-j2\pi f} \overline{H}_0(f - \frac{1}{2})$$

For the Haar transform, we have:

$$h_1[k] = (-1)^k h_0[1 - k] = \frac{1}{\sqrt{2}} \begin{cases} 1 & k = 0 \\ -1 & k = 1 \\ 0 & \text{otherwise} \end{cases} \quad (11.93)$$

4. Find wavelet function $\psi(t)$ according to Eq.11.78

$$\psi(t) = \sqrt{2} \sum_k (-1)^k \bar{h}_0[1 - k] \phi(2t - k)$$

or $\Psi(f)$ according to Eq.11.52

$$\Psi(f) = \frac{1}{\sqrt{2}} H_1(\frac{f}{2}) \Phi(\frac{f}{2})$$

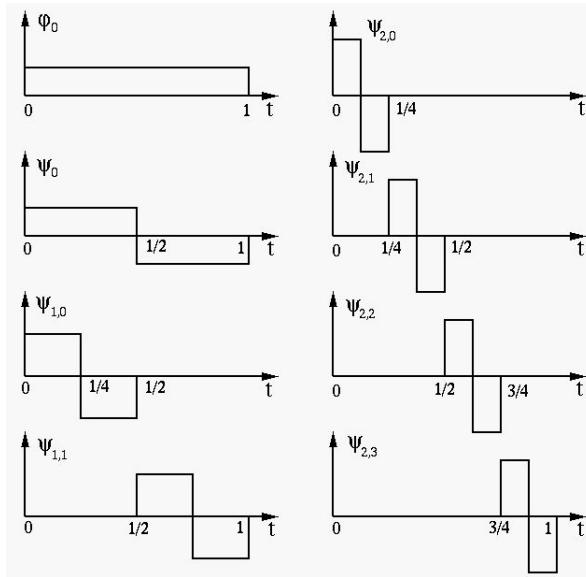


Figure 11.7 Haar scaling and wavelet functions

For Haar transform, we have:

$$\psi(t) = h_1[0]\phi_{1,0}(t) + h_1[1]\phi_{1,1}(t) = \begin{cases} 1 & 0 \leq t < 1/2 \\ -1 & 1/2 \leq t < 1 \\ 0 & \text{else} \end{cases} \quad (11.94)$$

Based on $\phi(0) = \phi_{0,0}(t)$ and $\psi(0) = \psi_{0,0}(t)$, all other $\psi_{j,k}(t)$ can be obtained, as the rows in the Haar matrix in Eq.8.76.

Obviously the Haar scaling and wavelet functions $\phi(t)$ and $\psi(t)$ have perfect temporal locality. However, similar to the ideal filter discussed before, the drawback of the Haar wavelets is their poor frequency locality, due obviously to their sinc-like $\Phi(f)$ and $\Psi(f)$ caused by the sharp corners of the rectangular time window in both $\phi(t)$ and $\psi(t)$.

- **Meyer wavelets**

Here we construct a wavelet with good locality in both time and frequency domains by avoiding sharp discontinuities in both domains. We start in frequency domain by considering the spectrum $\Phi(f)$ of the scaling function $\phi(t)$. First define a function for the smooth transition from 0 to 1 and then use it to define a smooth frequency window. Specifically consider a third order polynomial shown in Fig.11.9(a):

$$\nu(f) = \begin{cases} 0 & f < 0 \\ 3f^2 - 2f^3 & 0 \leq f \leq 1 \\ 1 & f > 1 \end{cases} \quad (11.95)$$

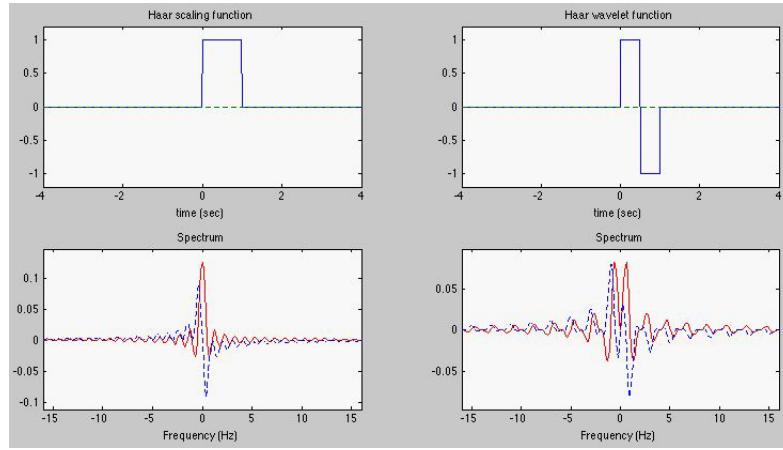


Figure 11.8 Haar scaling and wavelet functions (top) and their spectra (bottom)
(The real and imaginary parts of the spectra are shown respectively by solid and dashed curves.)

and define $\Phi(f)$ as:

$$\Phi(f) = \begin{cases} \sqrt{\nu(2+3f)} & f \leq 0 \\ \sqrt{\nu(2-3f)} & f \geq 0 \end{cases} \quad (11.96)$$

Here the function $3f^2 - 2f^3$ is chosen so that $\nu(1/2) = 1/2$ and $\nu(f) + \nu(1-f) = 1$, in order to satisfy the orthogonality in Eq.11.13. (Other functions such as $10f^3 - 15f^4 + 6f^5$ satisfying the same conditions could also be used.) As shown in Fig.11.9(b), $\Phi^2(f) = 1$ when $|f| \leq 1/3$, $\Phi^2(f) = 0$ when $2/3 \leq |f| < 1$, and $\phi^2(f) + \phi^2(f \pm 1) = 1$ during the transition interval $1/3 < |f| < 2/3$ where the two neighboring copies of $\Phi(f)$ overlap, i.e., Eq.11.13 is indeed satisfied.

Given $\Phi(f)$, we next find the scaling filter $H_0(f)$ based on $H_0(f) = \sqrt{2}\Phi(2f)/\Phi(f)$ (Eq.11.23), where $\Phi(2f)$, a compressed version of $\Phi(f)$, is zero for all $|f| > 1/3$. When $|f| < 1/3$, $\Phi(f) = 1$ and $\Phi(2f)/\Phi(f) = \Phi(2f)$. Also, as $H_0(f \pm 1) = H_0(f)$ is periodic, it can be obtained as:

$$H_0(f) = \sum_k \Phi(2(f-k)) = \sum_k \Phi(2f-2k) \quad (11.97)$$

These functions $\Phi(f)$, $\Phi(2f)$ and $H_0(f)$ are shown in Fig.11.9(b), (c) and (d), respectively.

Given $H_0(f)$, we can find $H_1(f)$ based on Eq.11.75:

$$H_1(f) = e^{-j2\pi f} \overline{H_0}(f - \frac{1}{2}) = e^{-j2\pi f} \sum_k \Phi(2f-2k-1) \quad (11.98)$$

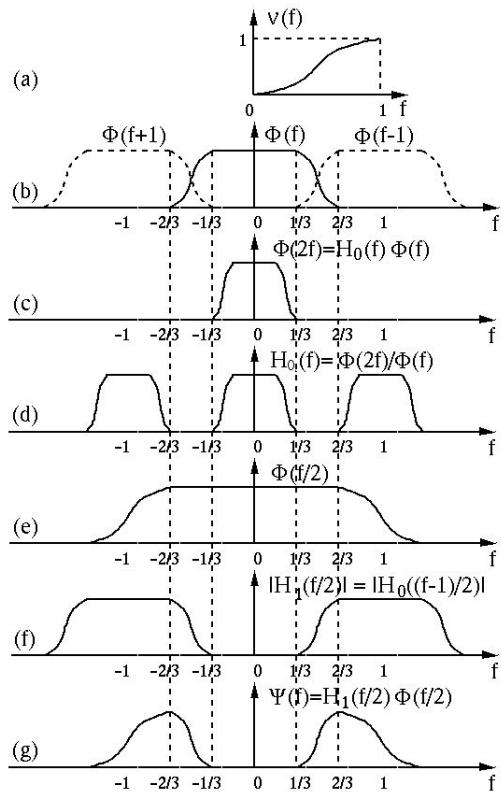


Figure 11.9 Construction of Meyer scaling and wavelet functions

and then $\Psi(f)$ based on Eq.11.52:

$$\begin{aligned}
 \Psi(f) &= \frac{1}{\sqrt{2}} H_1\left(\frac{f}{2}\right) \Phi\left(\frac{f}{2}\right) = \frac{1}{\sqrt{2}} e^{-j\pi f} H_0\left(\frac{f-1}{2}\right) \Phi\left(\frac{f}{2}\right) \\
 &= \frac{1}{\sqrt{2}} e^{-j\pi f} \sum_k \Phi(f-2k-1) \Phi\left(\frac{f}{2}\right) \\
 &= \begin{cases} 0 & |f| < 1/3 \\ -\frac{1}{\sqrt{2}} e^{-j2\pi f} \Phi(f-1) & 1/3 < |f| < 2/3 \\ -\frac{1}{\sqrt{2}} e^{-j2\pi f} \Phi(f/2) & 2/3 < |f| < 4/3 \\ 0 & |f| > 4/3 \end{cases} \quad (11.99)
 \end{aligned}$$

These functions $\Phi(f/2)$, $H_1(f/2)$ and $\Psi_0(f)$ are shown in Fig.11.9(e), (f) and (g), respectively.

Finally, the scaling function $\phi(t)$ and wavelet function $\psi(t)$ can be obtained by inverse Fourier transform of $\Phi(f)$ and $\Psi(f)$, respectively, as shown in Fig.11.10, and the coefficients for the scaling and wavelet filters can be found

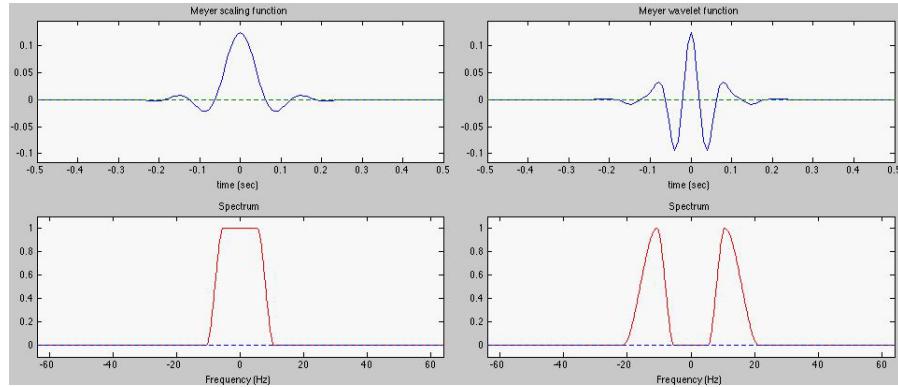


Figure 11.10 Meyer scaling and wavelet functions (top) and their spectra (bottom)
(The real and imaginary parts of the spectra are shown respectively by solid and dashed curves.)

by the inverse DTFT:

$$h_i[k] = \mathcal{F}^{-1}[H_i(f)] = \int_0^1 H_i(f) e^{-j2\pi kf} df, \quad i = 0, 1, \quad k \in \mathbb{Z} \quad (11.100)$$

The Matlab code segment for generating the Mayer wavelets is given below:

```
N=1024; % total number of samples
M=N/8; % size of period
Phi=zeros(1,N);
Psi=zeros(1,N);
for i=1:N
    f=abs(i-N/2-1);
    if f<M/3
        Phi(i)=1;
    elseif f<2*M/3
        Phi(i)=nu(2-f/(M/3)); % Meyer scaling function
    end
    if (f>M/3 & f<2*M/3)
        Psi(i)=nu(f/(M/3)-1);
    elseif (f>2*M/3 & f<4*M/3)
        Psi(i)=nu(2-f/(2*M/3));
    end
end
phi=fftshift(ifft(fftshift(Phi)));
psi=fftshift(ifft(fftshift(Psi)));
where
function y = nu(f)
y=3*f^2-2*f^3;
```

```
end
```

- **Daubechies' wavelets**

In addition to the temporal locality (ideally with compact support), it is also desirable for a wavelet function $\psi(t)$ to have a high number of *vanishing moments*, so that a signal can be effectively represented by the wavelet transform.

To understand this point, we first need to understand the concepts of vanishing moments and *regular functions*. The number of vanishing moments of a wavelet function $\psi(t)$ is N if all of its moments lower than N are zero:

$$\langle t^n, \psi(t) \rangle = \int_{-\infty}^{\infty} t^n \psi(t) dt = 0, \quad 0 \leq n < N \quad (11.101)$$

Also, a function $x(t)$ is regular if it can be approximated by a polynomial $p(t) = \sum_{n=0}^M c_n t^n$ around any t . When this signal is represented in space W_0 spanned by the wavelet basis $\psi_{0,k} = \psi(t - k)$:

$$x(t) = \sum_k d_{0,k} \psi_{0,k}(t) = \sum_k d_{0,k} \psi(t - k) \quad (11.102)$$

then the coefficient

$$d_{0,k} = \langle x(t), \psi(t - k) \rangle \approx \sum_{n=0}^M c_n \langle t^n, \psi(t - k) \rangle \quad (11.103)$$

is zero if $N > M$. We see that the greater number of vanishing moments N , the more coefficients $d_{0,k}$ in the wavelet expansion may become zero (or small enough to be ignored). The same argument can also be made for the higher scale levels of $j > 0$. Due to this result of much reduced transform coefficients, the signal can be more effectively represented. This is obviously desirable in various applications such as data compression.

Daubechies' wavelets are widely used due to its two favorite features: (1) they are compactly supported, and (2) they have the maximal number of vanishing moments for a given support. The derivation of Daubechies' wavelets is based on the normalization and orthogonality properties of $H_0(f)$ given in Eqs.11.58 and 11.63. But for convenience and conciseness we will use the alternative expressions of these properties in Eqs.11.71 through 11.74 in terms of $H_0(z)$ in Z-domain (with $z = e^{j2\pi f}$).

We let $z = 1$ in Eq.11.72 and note $H_0(1) = \sqrt{2}$ to get:

$$|H_0(1)|^2 + |H_0(-1)|^2 = 2 + |H_0(-1)|^2 = 2 \quad (11.104)$$

from which we get $|H_0(-1)| = 0$. We further see that $z = -1$ is a root of the polynomial $H_0(z) = \sum_k h_0[k]z^{-k}$, i.e., it must have a factor $(1 + z^{-1})^N$ for some N , and therefore can be written in the following form:

$$H_0(z) = (1 + z^{-1})^N Q(z) \quad (11.105)$$

where $Q(z)$ is a polynomial of z^{-1} . Daubechies proved¹ that the minimum degree of $Q(z)$ is $N - 1$, i.e., $H_0(z)$ is a polynomial of order $N + N - 1 = 2N - 1$ containing $2N$ terms $h_0[k]z^{-k}$ ($k = 0, \dots, 2N - 1$), assuming the filter is causal with $h_0[k] = 0$ for all $k < 0$), and the scaling and wavelet functions $\phi(t)$ and $\psi(t)$ corresponding such a $H_0(z)$ are compactly supported. Specifically, $\phi(t) \neq 0$ for $0 \leq t \leq 2N - 1$ and $\psi(t) \neq 0$ for $-(N - 1) \leq t \leq N$, and the wavelet function has the maximum number of vanishing moment N given the compact support of length $2N$.

Here we consider the three cases when $N = 1$, $N = 2$ and $N = 3$.

- $N = 1$ (Daubechies 2 or D2, same as the Haar transform):

The order of $Q(z)$ is $N - 1 = 0$, i.e., $Q(z) = c$ is a constant and $H_0(z) = c(1 + z^{-1})$. But as $H_0(1) = c2 = \sqrt{2}$, we get $c = 1/\sqrt{2}$ and $h_0[0] = h_0[1] = 1/\sqrt{2}$, i.e., this is the Haar scaling filter already considered above.

- $N = 2$ (Daubechies 4 or D4):

The order of $Q(z)$ is $N - 1 = 1$ and

$$H_0(z) = (1 + z^{-1})^2 Q(z) = (1 + z^{-1})^2(c_0 + c_1 z^{-1}) \quad (11.106)$$

The two coefficients c_0 and c_1 can be obtained by using Eqs.11.71 through 11.74 as constraining equations. We first evaluate $H_0(z)$ above at $z = 1$ to get (Eq.11.71):

$$H_0(1) = 4(c_0 + c_1) = \sqrt{2}, \quad \text{i.e.} \quad c_1 + c_2 = \frac{\sqrt{2}}{4} \quad (11.107)$$

We next evaluate $H_0(z)$ and $H_0(-z)$ at $z = j$ to get:

$$\begin{aligned} H_0(j) &= (1 - j)^2(c_0 - jc_1) = -2(jc_0 + c_1) \\ H_0(-j) &= (1 + j)^2(c_0 + jc_1) = 2(jc_0 - c_1) \end{aligned} \quad (11.108)$$

Substituting these into Eq.11.72 we get

$$|H_0(j)|^2 + |H_0(-j)|^2 = 8(c_0^2 + c_1^2) = 2, \quad \text{i.e.} \quad c_0^2 + c_1^2 = \frac{1}{4} \quad (11.109)$$

Solving Eqs.11.107 and 11.109, we get $c_{0,1} = (1 \pm \sqrt{3})/4\sqrt{2}$ and

$$\begin{aligned} H_0(z) &= \frac{1}{4\sqrt{2}}(1 + z^{-1})^2[(1 + \sqrt{3}) + (1 - \sqrt{3})z^{-1}] \\ &= \frac{1}{4\sqrt{2}}[(1 + \sqrt{3}) + (3 + \sqrt{3})z^{-1} + (3 - \sqrt{3})z^{-2} + (1 - \sqrt{3})z^{-3}] \end{aligned} \quad (11.110)$$

¹ Daubechies, I., *Ten Lectures on Wavelets* (CBMS-NSF Regional Conference Series in Applied Mathematics), Society for Industrial and Applied Mathematics, 1992

and the four Daubechies scaling filter coefficients are:

$$\begin{aligned} h_0[0] &= \frac{1 + \sqrt{3}}{4\sqrt{2}} = 0.4829621 \\ h_0[1] &= \frac{3 + \sqrt{3}}{4\sqrt{2}} = 0.8365163 \\ h_0[2] &= \frac{3 - \sqrt{3}}{4\sqrt{2}} = 0.2241439 \\ h_0[3] &= \frac{1 - \sqrt{3}}{4\sqrt{2}} = -0.1294095 \end{aligned} \quad (11.111)$$

The corresponding wavelet coefficients can be obtained according to Eq.11.77 $h_1[k] = (-1)^k h_0[1-k]$ as:

$$\begin{aligned} h_1[1] &= -h_0[0] = -0.4829621 \\ h_1[0] &= h_0[1] = 0.8365163 \\ h_1[-1] &= -h_0[2] = -0.2241439 \\ h_1[-2] &= h_0[3] = -0.1294095 \end{aligned} \quad (11.112)$$

– $N = 3$ (Daubechies 6 or D6):

The order of $Q(z)$ is $N - 1 = 2$ and

$$H_0(z) = (1 + z^{-1})^3 Q(z) = (1 + z^{-1})^3 (c_0 + c_1 z^{-1} + c_2 z^{-2}) \quad (11.113)$$

Here again the three coefficients c_0 , c_1 and c_2 can be obtained by using the normalization and orthogonality conditions given in Eqs. 11.71 through 11.74 as the constraining equations. Similar to the case of $N = 2$, we can find the $2N = 6$ coefficients of the scaling filter as:

$$\begin{aligned} h_0[0] &= \left[1 + \sqrt{10} + \sqrt{5 + 2\sqrt{10}} \right] / 16\sqrt{2} = 0.3326706 \\ h_0[1] &= \left[5 + \sqrt{10} + 3\sqrt{5 + 2\sqrt{10}} \right] / 16\sqrt{2} = 0.8068915 \\ h_0[2] &= \left[10 - 2\sqrt{10} + 2\sqrt{5 + 2\sqrt{10}} \right] / 16\sqrt{2} = 0.4598775 \\ h_0[3] &= \left[10 - 2\sqrt{10} - 2\sqrt{5 + 2\sqrt{10}} \right] / 16\sqrt{2} = -0.1350110 \\ h_0[4] &= \left[5 + \sqrt{10} - 3\sqrt{5 + 2\sqrt{10}} \right] / 16\sqrt{2} = -0.0854413 \\ h_0[5] &= \left[1 + \sqrt{10} - \sqrt{5 + 2\sqrt{10}} \right] / 16\sqrt{2} = 0.0352263 \end{aligned} \quad (11.114)$$

The corresponding wavelet coefficients $h_1[k]$ can be obtained from Eq.11.77. No analytical expression exists for either the scaling function $\phi(t)$ or the wavelet function $\psi(t)$. However, once the coefficients $h_0[k]$ and $h_1[k]$ for the

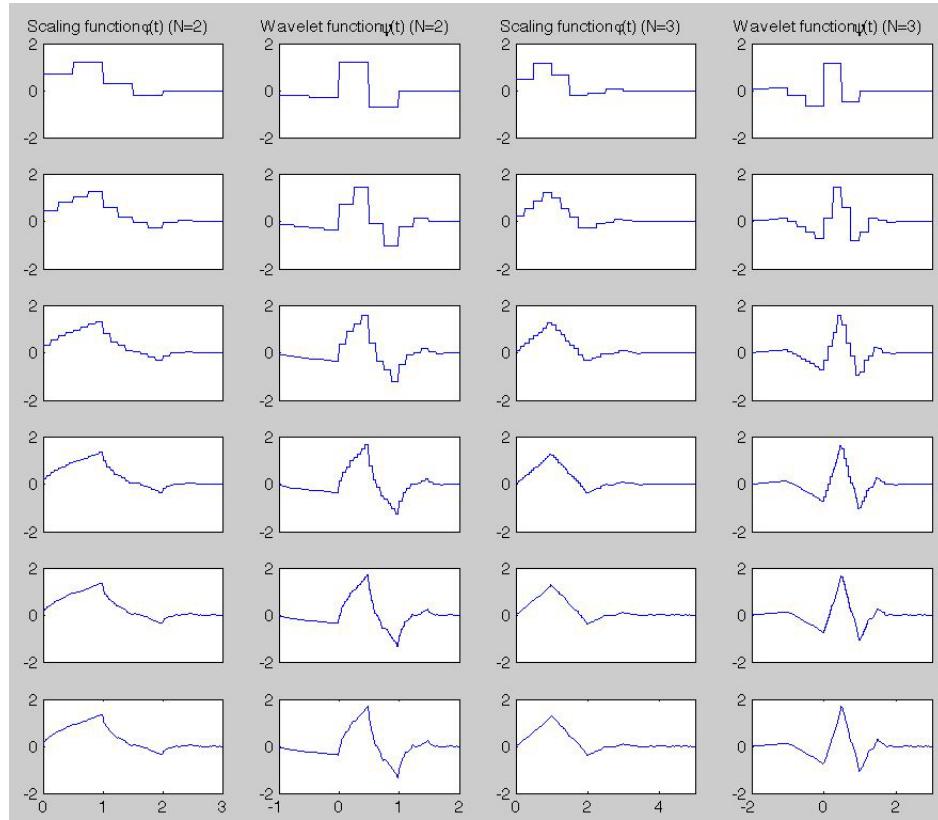


Figure 11.11 Iterative approximations of Daubechies' scaling and wavelet functions
The scaling and wavelet functions $\phi(t)$ and $\psi(t)$ for $N = 2$ are shown in the two columns on the left while those for $N = 3$ are shown in the two columns on the right. The six rows show the first six intermediate results of the iteration based on Eqs.11.19 and 11.48.

scaling and wavelet filters are available, $\phi(t)$ and $\psi(t)$ can be iteratively constructed by Eqs.11.19 and 11.48 (or Eqs.11.25 and 11.54 in frequency domain) based on the initial D2 or Haar scaling function (Eq.11.91).

Such a construction is implemented in the Matlab function given below, by which the Daubechies scaling function $\phi(t)$ and wavelet function $\psi(t)$ are iteratively constructed. The resulting scaling and wavelet functions of the first six iterations are shown in Fig.11.11. The waveforms of the scaling and wavelet functions of order $N > 3$ can be similarly obtained and they indeed become smoother as order N increases.

```
function daubechies
T=3; % time period in second
s=64; % sampling rate: s samples/second
t0=1/s; % sampling period
```

```

N=T*s;                      % total number of samples
K=4;                         % length of coefficient vector
r3=sqrt(3);
h0=[1+r3 3+r3 3-r3 1-r3]/4; % Daubechies coefficients
h1=fliplr(h0);              % time reversal of h0
h1(2:2:K)=-h1(2:2:K);      % negate odd terms

phi=zeros(1,N);             % scaling function
psi=zeros(1,N);             % wavelet function
phi0=zeros(1,N);
for i=1:s
    phi0(i)=1;              % initialize scaling function
end
for j=1:log2(s);
    for n=1:N
        phi(n)=0; psi(n)=0;
        for k=0:3
            l=2*n-k*s;
            if (l>0 & l<=N)
                phi(n)=phi(n)+h0(k+1)*phi0(l);
            end
            l=2*n-k*s;
            if (l>0 & l<=N)
                psi(n)=psi(n)+h1(k+1)*phi0(l);
            end
        end
    end
    phi0=phi;                 % update scaling function
end
subplot(2,1,1)
plot(0:t0:T-t0,phi)
title('Scaling function');
subplot(2,1,2)
plot(-1:t0:T-1-t0,psi);
title('Wavelet function')

```

11.2 Discrete Wavelet Transform (DWT)

11.2.1 Discrete Wavelet Transform (DWT)

To numerically carry out the wavelet series expansion of a signal $x(t)$ as shown in Eq.11.84, both the scaling function $\phi_{0,k}(t)$ and wavelet function $\psi_{j,k}(t)$ as well

as the signal $x(t)$ need to be discretized so that they are all represented as N-D vectors $\phi_{0,k}$, $\psi_{j,k}$ and \mathbf{x} , composed respectively of $\phi_{j,k}[n]$, $\psi_{j,k}[n]$ and $x[n]$ as the nth components. Due to the dyadic scaling, there are in total $J = \log_2 N$ scale levels (N assumed to be a power of 2 for convenience). Also, as the data size is 2^j at each level $j = 0, \dots, J-1$, there are 2^j possible translations $k = 0, \dots, 2^j - 1$. In particular, at the highest scale level $j = J-1$, there is only $2^0 = 1$ sample at the lowest scale level $j = 0$ and $2^{J-1} = N/2$ samples.

Now the wavelet expansion becomes the *discrete wavelet transform (DFT)* by which the discrete signal $\mathbf{x} = [x[0], \dots, x[N-1]]^T$ is represented as a weighted sum in the scaling and wavelet spaces spanned by the orthogonal basis vectors $\phi_{0,k}$ and $\psi_{j,k}$:

$$\mathbf{x} = \langle \mathbf{x}, \phi_{0,0} \rangle \phi_{0,0} + \sum_{j=0}^{J-1} \sum_{k=0}^{2^j-1} \langle \mathbf{x}, \psi_{j,k} \rangle \psi_{j,k} \quad (11.115)$$

which can also be represented in component form:

$$x[n] = X_\phi[0,0]\phi_{0,0}[n] + \sum_{j=0}^{J-1} \sum_{k=0}^{2^j-1} X_\psi[j,k]\psi_{j,k}[n], \quad (n = 0, \dots, N-1) \quad (11.116)$$

This is the inverse DWT by which the signal \mathbf{x} is reconstructed from its DWT *approximation coefficient* $X_\phi[0,0]$ and *detail coefficients* $X_\psi[j,k]$, which can be found as the projections of the signal vector onto the corresponding basis vectors, similar to the case of wavelet series expansion in Eqs.11.83 and 11.85:

$$X_\phi[0,0] = \langle \mathbf{x}, \phi_{0,0} \rangle = \sum_{n=0}^{N-1} x[n] \bar{\phi}_{0,0}[n] \quad (11.117)$$

$$X_\psi[j,k] = \langle \mathbf{x}, \psi_{j,k} \rangle = \sum_{n=0}^{N-1} x[n] \bar{\psi}_{j,k}[n], \quad (j = 0, \dots, J-1, k = 0, \dots, 2^j-1) \quad (11.118)$$

These equations are the forward DWT by which the DWT coefficients are obtained, including $X_\phi[0,0]$ and $X_\psi[j,k]$ for all $J = \log_2$ scale levels ($j = 0, \dots, J-1 = \log_2 N - 1$) each with 2^j integer translations ($k = 0, \dots, 2^j - 1$). As there are in total $1 + \sum_{j=0}^{J-1} 2^j = 2^J = N$ coefficients, we can arrange them also as an N-D vector in the DWT transform domain, as shown in Fig.11.13 for $N = 2^3 = 8$.

At the lowest level when $j = 0$, the signal is simply approximated by its average represented by $\phi_{0,0}[n] = 1$. However it is not always necessary to start the approximation process from this lowest scale level. On the other hand, at the highest possible level when $j = J$ (not part of the DFT in Eqs.11.116 or 11.118), the full resolution is achieved in V_J where the signal is simply represented by all of its N original samples $x[n]$ ($n = 0, \dots, N-1$).

Same as all other discrete orthogonal transforms considered in previous chapters, the DWT also represents a discrete signal in terms of its DWT coefficients. (Note, however, different from all previous transforms, the DWT coefficients represent different translations as well as different scales, while the coefficients of other transforms, such as the DFT and DCT, only represent different frequencies.) In the DWT domain, various signal processing operations, such as filtering, noise reduction, feature extraction and data compression, can be carried out. The inverse DWT transform can then be carried out to reconstruct the signal back in time domain.

Example 11.4: When $N = 4$, the discrete Haar scaling and wavelet functions are given as the rows of the following matrix (Eq.8.74):

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix} \begin{array}{l} \phi_{0,0}[n] \\ \psi_{0,0}[n] \\ \psi_{1,0}[n] \\ \psi_{1,1}[n] \end{array} \quad (11.119)$$

Given a discrete signal $\mathbf{x} = [x[0], \dots, x[N-1]]^T = [1, 4, -3, 0]^T$, the DWT coefficients can be found by Eqs.11.117 and 11.118. The coefficient in V_0 is:

$$X_\phi[0, 0] = \frac{1}{2} \sum_{m=0}^3 x[n]\phi_{0,0}[n] = \frac{1}{2}[1 \cdot 1 + 4 \cdot 1 - 3 \cdot 1 + 0 \cdot 1] = 1 \quad (11.120)$$

The coefficient in W_0 is:

$$X_\psi[0, 0] = \frac{1}{2} \sum_{m=0}^3 x[n]\psi_{0,0}[n] = \frac{1}{2}[1 \cdot 1 + 4 \cdot 1 - 3 \cdot (-1) + 0 \cdot (-1)] = 4 \quad (11.121)$$

The two coefficients in W_1 are:

$$X_\psi[1, 0] = \frac{1}{2} \sum_{m=0}^3 x[n]\psi_{1,0}[n] = \frac{1}{2}[1 \cdot \sqrt{2} + 4 \cdot (-\sqrt{2}) - 3 \cdot 0 + 0 \cdot 0] = -1.5\sqrt{2} \quad (11.122)$$

$$X_\psi[1, 1] = \frac{1}{2} \sum_{m=0}^3 x[n]\psi_{1,1}[n] = \frac{1}{2}[1 \cdot 0 + 4 \cdot 0 - 3 \cdot \sqrt{2} + 0 \cdot (-\sqrt{2})] = -1.5\sqrt{2} \quad (11.123)$$

Or in matrix form, we have

$$\begin{bmatrix} 1 \\ 4 \\ -1.5\sqrt{2} \\ -1.5\sqrt{2} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix} \begin{bmatrix} 1 \\ 4 \\ -3 \\ 0 \end{bmatrix} \quad (11.124)$$

Now the 4-point discrete signal can be expressed as a linear combination of these basis functions:

$$\begin{aligned} x[n] &= \frac{1}{2}[X_\phi[0,0]\phi_{0,0}[n] + C_\psi[0,0]\psi_{0,0}[n] + X_\phi[1,0]\psi_{1,0}[n] + X_\phi[1,1]\psi_{1,1}[n]] \\ n &= 0, \dots, 3 \end{aligned} \quad (11.125)$$

or in matrix form:

$$\begin{bmatrix} 1 \\ 4 \\ -3 \\ 0 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 & \sqrt{2} & 0 \\ 1 & 1 & -\sqrt{2} & 0 \\ 1 & -1 & 0 & \sqrt{2} \\ 1 & -1 & 0 & -\sqrt{2} \end{bmatrix} \begin{bmatrix} 1 \\ 4 \\ -1.5\sqrt{2} \\ -1.5\sqrt{2} \end{bmatrix} \quad (11.126)$$

This is the inverse DWT.

11.2.2 Fast Wavelet Transform (FWT)

The total number of operations in Eq. 11.118 for the forward DWT (or Eq. 11.116 for the inverse DWT) is proportional to the product of the vector length N and number of integer shifts $\sum_{j=0}^{J-1} 2^j = N$, i.e., the computational complexity is $O(N^2)$. For example, when the Haar transform as a DWT is implemented as a matrix multiplication in Eq. 8.80, its complexity is obviously $O(N^2)$. Now we will consider Mallat's fast wavelet transform (FWT) algorithm for the DWT with a linear complexity of $O(N)$ (as we have already seen for the case of discrete Haar transform in subsection 8.3.3).

A given N-D signal vector $\mathbf{x} = [x[0], \dots, x[N-1]]^T$ can be represented in any scale space V_j spanned by orthogonal basis $\phi_{j,k}$ or wavelet space W_j spanned by orthogonal basis $\psi_{j,k}$ ($j = 0, \dots, J-1$), in terms of the following coefficients:

$$X_\phi[j, k] = \sum_{n=0}^{N-1} x[n]\bar{\phi}_{j,k}[n], \quad X_\psi[j, k] = \sum_{n=0}^{N-1} x[n]\bar{\psi}_{j,k}[n] \quad (11.127)$$

Note that these equations are the same as the forward DWT in Eqs. 11.117 (when $j = k = 0$) and 11.118. Moreover, due to the recursive relationships of $\phi_{j,l}(t)$ and $\psi_{j,l}(t)$ (Eqs. 11.22 and 11.51), these equations can both be expressed in terms of

the coefficients $X_\phi[j+1, k]$ at the next higher scale level:

$$\begin{aligned} X_\phi[j, k] &= \sum_{n=0}^{N-1} x[n] \sum_l \bar{h}_0[l - 2k] \bar{\phi}_{j+1,l}[n] = \sum_l \bar{h}_0[l - 2k] \sum_{n=0}^{N-1} x[n] \bar{\phi}_{j+1,l}[n] \\ &= \sum_l \bar{h}_0[l - 2k] X_\phi[j+1, l] \end{aligned} \quad (11.128)$$

$$\begin{aligned} X_\psi[j, k] &= \sum_{n=0}^{N-1} x[n] \sum_l \bar{h}_1[l - 2k] \bar{\phi}_{j+1,l}[n] = \sum_l \bar{h}_1[l - 2k] \sum_{n=0}^{N-1} x[n] \phi_{j+1,l}[n] \\ &= \sum_l \bar{h}_1[l - 2k] X_\phi[j+1, l] \end{aligned} \quad (11.129)$$

These operations can be carried out recursively until the highest scale level at $j+1 = J$ is reached. The corresponding space V_J is spanned by $\phi_{J,k}[n] = \delta[k - n]$ as the standard basis, and the signal x is simply represented by all of its N samples:

$$X_\phi[J, k] = \sum_{n=0}^{N-1} x[n] \phi_{J,k}[n] = \sum_{n=0}^{N-1} x[n] \delta[k - n] = x[k] \quad (k = 0, \dots, N-1) \quad (11.130)$$

Comparing Eqs.11.128 and 11.129 with the discrete convolution (Eq.4.147):

$$y[k] = h[k] * x[k] = \sum_{n=0}^{N-1} x[n] h[k - n] \quad (11.131)$$

we see that the DWT coefficients $X_\phi[j, k]$ and $X_\psi[j, k]$ at the j th scale level can both be obtained from the coefficients $X_\phi[j+1, k]$ at the $(j+1)$ th scale level by:

- Convolution with time-reversed $\bar{h}_0[k]$ and $\bar{h}_1[k]$;
- Sub-sampling to keep every other samples in the convolution.

Eqs.11.128 and 11.129 can therefore be considered as a filtering process:

$$\begin{aligned} X_\phi[j, k] &= \bar{h}_0[-l] * X_\phi[j+1, l] \Big|_{l=2k} \\ X_\psi[j, k] &= \bar{h}_1[-l] * X_\phi[j+1, l] \Big|_{l=2k} \end{aligned} \quad (11.132)$$

(Note that this operation of convolution followed by sub-sampling for the DWT coefficients is the same as in that in Eqs.11.22 and 11.51 for the scaling and wavelet functions.) This filtering process can be implemented as either a convolution in time domain or, equivalently, as a multiplication in frequency domain.

Now we see that both $X_\phi[j, k]$ and $X_\psi[j, k]$ for all $j < J$ can be obtained by filtering $X_\phi[j+1, k]$ of the next higher scale level $j+1$, which in turn can be obtained from still higher level $j+2$, and this recursion can be carried out until the highest level $j = J$ is reached where $X_\phi[J, k] = x[k]$ are simply the N signal samples originally given. Based on this recursion, the forward DWT in Eqs.11.117 and 11.118 can be implemented by the *analysis filter bank* shown on the left-hand side of Fig.11.12, by which all N DWT coefficients in Eqs.11.117

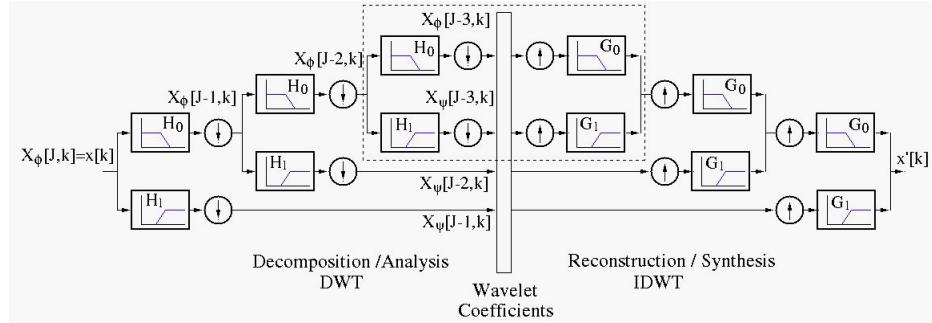


Figure 11.12 Filter banks for both forward and inverse DWT

Inside the dashed box is a two-channel filter bank as the building block of the decomposition-reconstruction filter bank system.

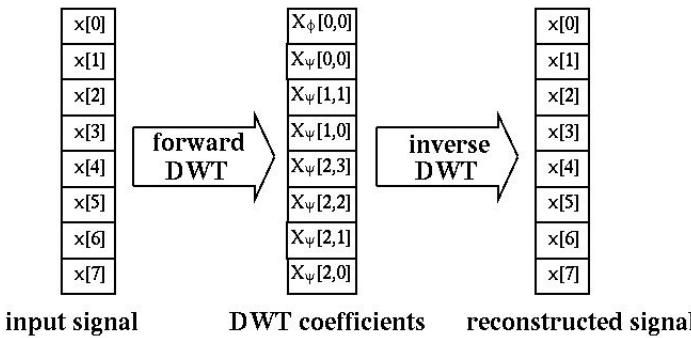


Figure 11.13 Vector representations of the forward and inverse DWT ($N = 8$)

and 11.118 can be generated, as represented by the vertical bar in the middle of the figure. This is the FWT algorithm. As the data size is halved by the sub-sampling of each iteration, the total computational complexity of the FWT is linear:

$$O(N + \frac{N}{2} + \frac{N}{4} + \frac{N}{8} + \dots + 1) = O(N) \quad (11.133)$$

The right-hand side of Fig.11.12 is for the inverse DWT by which the signal is to be reconstructed from its DWT coefficients, to be discussed next. Same as all orthogonal transforms considered before, for an N-D signal vector $\mathbf{x} = [x[0], \dots, x[N - 1]]^T$, there are also N DWT coefficients in the transform domain which can be arranged as an N-D vector, same as the N-D spectrum vector of the DFT or DCT, as shown in Fig.11.13.

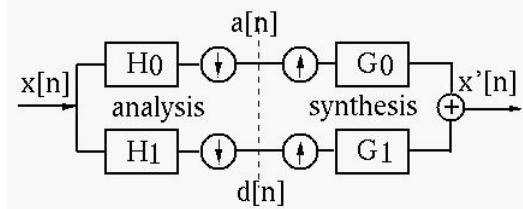


Figure 11.14 Two-channel filter bank

11.3 Filter Bank Implementation of DWT and Inverse DWT

11.3.1 Two-Channel Filter Bank and Inverse DWT

The complexity of the inverse DWT in Eq.11.116 for signal reconstruction is $O(N^2)$, as mentioned above. Here we consider the fast inverse DWT that can be carried as a sequence of filtering operations with the same linear complexity $O(N)$ for the forward DWT in the *synthesis filter bank*, as illustrated on the right-hand side of Fig.11.12. In the following we will derive the theory needed for the design of the filters G_0 and G_1 in the synthesis filter bank.

The DWT filter bank shown in Fig.11.12 can be considered as a hierarchical structure composed of a set of basic *two-channel filter banks*, which in turn is composed of two pairs of filters, the analysis and synthesis filter banks, as shown inside the dashed box in Fig.11.12 and also Fig.11.14. The analysis bank contains a low-pass filter, represented by $h_0[n]$ or $H_0(f) = \mathcal{F}[h_0[n]]$, that takes input $x[n]$ and generates output $a[n]$ (approximation), and a high-pass filter, represented by $h_1[n]$ or $H_1(f) = \mathcal{F}[h_1[n]]$, that takes the same input $x[n]$ and generates output $d[n]$ (detail). Each of these filters is followed by a down sampler. The synthesis bank also contains a pair of filters represented respectively by $g_0[n]$ or $G_0(f) = \mathcal{F}[g_0[n]]$ and $g_1[n]$ or $G_1(f) = \mathcal{F}[g_1[n]]$, each proceeded by an up sampler. We have already considered filters $h_0[n]$ and $h_1[n]$ of the analysis filter bank, and will now concentrate on the design of $g_0[n]$ and $g_1[n]$, so that the sum of their outputs, the output \hat{x} of the synthesis filter bank, can be identical to the input x (with possibly some delay). Once this perfect reconstruction is achieved by the basic two-channel filter bank at this lowest level, it can also be achieved recursively at each of the next higher levels in the entire filter bank in Fig.11.12.

As in Eqs.11.128 and 11.129, the outputs $a[k]$ and $d[k]$ of the two analysis filters of the two-channel filter bank can be written as:

$$a[k] = \sum_n h_0[n - 2k]x[n], \quad d[k] = \sum_n h_1[n - 2k]x[n] \quad (11.134)$$

which can be considered as the inner products of vectors $\mathbf{x} = [\dots, x[n], \dots]^T$ and $\mathbf{h}_i(k) = [\dots, h_i[n - 2k], \dots]^T$ ($i = 0, 1$):

$$a[k] = \langle \mathbf{x}, \mathbf{h}_0(k) \rangle, \quad d[k] = \langle \mathbf{x}, \mathbf{h}_1(k) \rangle \quad (11.135)$$

The output $\hat{x}[n]$ of the two-channel filter bank can be written as:

$$\hat{x}[n] = \sum_k a[k]g_0[n - 2k] + \sum_k d[k]g_1[n - 2k], \quad \text{for all } n \quad (11.136)$$

or in vector form:

$$\begin{aligned} \hat{\mathbf{x}} &= \sum_k a[k]\mathbf{g}_0(k) + \sum_k d[k]\mathbf{g}_1(k) \\ &= \sum_k \langle \mathbf{x}, \mathbf{h}_0(k) \rangle \mathbf{g}_0(k) + \sum_k \langle \mathbf{x}, \mathbf{h}_1(k) \rangle \mathbf{g}_1(k) \end{aligned} \quad (11.137)$$

where $\mathbf{g}_0(k)$ and $\mathbf{g}_1(k)$ are two vectors composed of the time-reversed version of the synthesis filter coefficients $g_i[n - 2k]$. Our goal here is to design the two filters $g_0[n]$ and $g_1[n]$ in the synthesis filter bank so that its output $\hat{x}[n] = x[n]$ is a perfect reconstruction of the input for the original signal. The derivation can be carried out in either time or frequency domain based on the DTFT or Z-transform (with $z = e^{j2\pi f}$ evaluated along the unit circle). Here we choose to use the DTFT approach, although the Z-transform is also used in some literatures. Note again that all DTFT spectra are periodic with period 1, e.g., $H_0(f \pm 1) = H_0(f)$ and $H_0(f + 1/2) = H_0(f - 1/2)$.

According to the down-sampling property of the DTFT (Eq.4.45 for $k = 2$), the sub-sampled outputs $a[n]$ of $H_0(f)$ and $d[n]$ of $H_1(f)$, when given the same input $x[n]$, can be expressed in frequency domain as:

$$A(f) = \frac{1}{2}[H_0(\frac{f}{2})X(\frac{f}{2}) + H_0(\frac{f+1}{2})X(\frac{f+1}{2})] \quad (11.138)$$

$$D(f) = \frac{1}{2}[H_1(\frac{f}{2})X(\frac{f}{2}) + H_1(\frac{f+1}{2})X(\frac{f+1}{2})] \quad (11.139)$$

Next, according to the upsampling property of the DTFT (Eq.4.51), the overall output of the two-channel filter bank can be expressed as:

$$\begin{aligned} \hat{X}(f) &= G_0(f)A(2f) + G_1(f)D(2f) \\ &= \frac{1}{2}[G_0(f)H_0(f) + G_1(f)H_1(f)] X(f) \\ &\quad + \frac{1}{2}[G_0(f)H_0(f + \frac{1}{2}) + G_1(f)H_1(f + \frac{1}{2})] X(f + \frac{1}{2}) \end{aligned} \quad (11.140)$$

For perfect reconstruction we need to have $X(f) = \hat{X}(f)$, i.e., the coefficient of the first term of $X(f)$ should be 1 (or a pure delay) and that of the second term of $X(f + 1/2)$ is zero:

$$\begin{cases} G_0(f)H_0(f) + G_1(f)H_1(f) = 2 \\ G_0(f)H_0(f + \frac{1}{2}) + G_1(f)H_1(f + \frac{1}{2}) = 0 \end{cases} \quad (11.141)$$

These two equations can be written in matrix form as:

$$\begin{bmatrix} H_0(f) & H_1(f) \\ H_0(f + \frac{1}{2}) & H_1(f + \frac{1}{2}) \end{bmatrix} \begin{bmatrix} G_0(f) \\ G_1(f) \end{bmatrix} = \mathbf{H}(f) \begin{bmatrix} G_0(f) \\ G_1(f) \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \end{bmatrix} \quad (11.142)$$

where \mathbf{H} is a 2 by 2 matrix defined as:

$$\mathbf{H}(f) = \begin{bmatrix} H_0(f) & H_1(f) \\ H_0(f + \frac{1}{2}) & H_1(f + \frac{1}{2}) \end{bmatrix}, \quad \mathbf{H}^{-1}(f) = \frac{1}{\Delta(f)} \begin{bmatrix} H_1(f + \frac{1}{2}) & -H_1(f) \\ -H_0(f + \frac{1}{2}) & H_0(f) \end{bmatrix} \quad (11.143)$$

where $\Delta(f)$ is the determinant of $\mathbf{H}(f)$:

$$\Delta(f) = H_0(f)H_1(f + \frac{1}{2}) - H_0(f + \frac{1}{2})H_1(f) \quad (11.144)$$

Note that

$$\Delta(f + \frac{1}{2}) = -\Delta(f) \quad (11.145)$$

Solving Eq.11.142 we get:

$$\begin{bmatrix} G_0(f) \\ G_1(f) \end{bmatrix} = \mathbf{H}^{-1}(f) \begin{bmatrix} 2 \\ 0 \end{bmatrix} = \frac{1}{\Delta(f)} \begin{bmatrix} H_1(f + \frac{1}{2}) & -H_1(f) \\ -H_0(f + \frac{1}{2}) & H_0(f) \end{bmatrix} \begin{bmatrix} 2 \\ 0 \end{bmatrix} \quad (11.146)$$

i.e.,

$$G_0(f) = \frac{2}{\Delta(f)} H_1(f + \frac{1}{2}), \quad G_1(f) = \frac{-2}{\Delta(f)} H_0(f + \frac{1}{2}) \quad (11.147)$$

Replacing f by $f + \frac{1}{2}$ in the second equation for $G_1(f)$ we get:

$$G_1(f + \frac{1}{2}) = \frac{2}{\Delta(f)} H_0(f) \quad (11.148)$$

Dividing the two sides of this equation by those of the first equation in Eq.11.147 we get:

$$G_1(f + \frac{1}{2})H_1(f + \frac{1}{2}) = G_0(f)H_0(f), \quad \text{or} \quad G_1(f)H_1(f) = G_0(f + \frac{1}{2})H_0(f + \frac{1}{2}) \quad (11.149)$$

This equation can be substituted back into the two equations in Eq.11.141 in different ways to get the following four conditions for perfect reconstruction:

$$\begin{aligned} G_0(f)H_0(f) + G_0(f + \frac{1}{2})H_0(f + \frac{1}{2}) &= 2 \\ G_1(f)H_1(f) + G_1(f + \frac{1}{2})H_1(f + \frac{1}{2}) &= 2 \\ G_1(f)H_0(f) + G_1(f + \frac{1}{2})H_0(f + \frac{1}{2}) &= 0 \\ G_0(f)H_1(f) + G_0(f + \frac{1}{2})H_1(f + \frac{1}{2}) &= 0 \end{aligned} \quad (11.150)$$

Comparing these four equations with Eq.11.63 required of $H_0(f)$ and $H_1(f)$ (for the orthogonalities of the scaling and wavelet functions), we see that if we let

$$G_0(f) = \overline{H}_0(f), \quad G_1(f) = \overline{H}_1(f) \quad (11.151)$$

then all four equations in Eq.11.150 hold, i.e., the condition for a perfect reconstruction is satisfied. Moreover, applying the DTFT property in Eq.4.30 to these

two relations in frequency domain we get the following in time domain:

$$g_0[n] = \bar{h}_0[-n], \quad g_1[n] = \bar{h}_1[-n] \quad (11.152)$$

In other words, the perfect reconstruction can be achieved by the synthesis filters if their coefficients are the complex conjugate and time reversed (conjugate mirror) version of coefficients of the analysis filters.

We also note that the four equations in Eq.11.150 are actually the down and upsampled versions of $G_0(f)H_0(f)$, $G_1(f)H_1(f)$, $G_1(f)H_0(f)$, and $G_0(f)H_1(f)$ (recall Eq.4.54), and they correspond to the following four down-sampled convolutions in time domain:

$$\begin{aligned} g_0[2n] * h_0[2n] &= \sum_k h_0[k]g_0[2n - k] = \delta[n] \\ g_1[2n] * h_1[2n] &= \sum_k h_1[k]g_1[2n - k] = \delta[n] \\ g_1[2n] * h_0[2n] &= \sum_k h_1[k]g_0[2n - k] = 0 \\ g_0[2n] * h_1[2n] &= \sum_k h_0[k]g_1[2n - k] = 0 \end{aligned} \quad (11.153)$$

Comparing these four time convolutions with Eq.11.60, we reach the same conclusion above: the condition for perfect reconstruction is satisfied if the coefficients of the synthesis filters satisfy $g_0[n] = \bar{h}_0[-n]$ and $g_1[n] = \bar{h}_1[-n]$.

To see how the two-channel filter bank can actually be implemented we list below the Matlab code, which carries out first the analysis filtering for signal decomposition with $H_0(f) = \mathcal{F}[h_0[k]]$ (filtering coefficients $h_0[k]$ provided as input) and $H_1(f)$ (Eq. 11.75), and then the synthesis filtering for signal reconstruction with $G_0(f)$ and $G_1(f)$ (Eq.11.151), both as multiplications in frequency domain (although they can also be equivalently carried out in time domain as circular convolutions).

```
function y=TwoChannelFilterBank(x,h)
    h=h/norm(h); % normalize h
    K=length(h); % length of filter (K<N)
    N=length(x); % length of signal vector
    h0=zeros(1,N); h0(1:K)=h; % analysis filter H0
    H0=fft(h0);
    for k=0:N-1
        l=mod(k-N/2,N); % rotation by 1/2
        H1(k+1)=exp(-j*2*pi*k/N)*conj(H0(l+1)); % analysis filter H1
    end
    G0=conj(H0); G1=conj(H1); % synthesis filters G0 and G1:
    % Decomposition by analysis filters:
    A=fft(x); % input
    d=ifft(A.*H1); % filtering to get d (detail)
```

```

a=ifft(A.*H0);           % filtering to get a (approximation)
d=d(1:2:length(d));     % downsampling d
a=a(1:2:length(a));     % downsampling a
% Reconstruction by synthesis filters:
a=upsample(a,2);         % upsampling a
d=upsample(d,2);         % upsampling d
a=ifft(fft(a).*G0);      % filtering of a
d=ifft(fft(d).*G1);      % filtering of d
y=a+d;                   % perfect reconstruction of x
end

```

Here the input x is the signal vector, and input h is a vector containing the filtering coefficients $h_0[k]$. For example, $h = [1 1]$ for D2 (Haar), or $h = [0.4830 \ 0.8365 \ 0.2241 \ -0.1294]$ for D4. The output y is a perfect reconstruction of the input x .

Having obtained the two-channel filter bank in Fig.11.14 capable of perfect reconstruction, we can use it as the building block to construct the filter bank in Fig.11.12, by which the input signal is perfectly reconstructed at the output. Note that the iteration of the DWT on the left of the figure can be terminated at any scale level before reaching the lowest possible scale level (top level in the figure), depending on the actual signal processing need, as the data can always be perfectly reconstructed from any level by the inverse DWT on the right.

The Matlab code for both the forward DWT for signal decomposition and the inverse DWT for signal reconstruction is listed below. The algorithm is basically a recursion of the operations in the two-channel filter bank shown above. The input of the forward DWT function includes a vector x for the signal to be DWT transformed and another vector h for the father wavelet coefficients $h_0[k]$, and the output is a vector w for the DWT coefficients. Note that the size $N = 2^n$ of the data vector x is assumed to be a power of two for convenience. Note that, unlike the fast algorithms of all previously considered orthogonal transforms (except DHT) of complexity of $O(N \log_2 N)$, all containing an inner loop that is carried out $\log_2 N$ times, here the number of iterations is smaller than $\log_2 N$ when the length of the filter h is greater than two (except D2 or Haar when h has only two non-zero components). In general, the iteration in the DWT does not have to be always carried out to the lowest possible scale level.

```

function w=mydwt(x,h)
K=length(h);
if K>N
    error('K should be less than N');          % assume N > K
end
N=length(x);
n=log2(N);
if n~=int16(n)

```

```

        error('Length of data x should be power of 2');
end
h=h/norm(h);                                     % normalize h
h0=zeros(1,N); h0(1:K)=h; H0=fft(h0);          % scaling function
for k=0:N-1
    l=mod(k-N/2,N);                           % rotation by 1/2
    H1(k+1)=exp(-j*2*pi*k/N)*conj(H0(l+1)); % wavelet function
end
a=x;
n=length(a);
w=[];
while n>=K
    A=fft(a);
    d=real(ifft(A.*H1));           % convolution d=a*h1
    a=real(ifft(A.*H0));           % convolution a=a*h0
    d=d(2:2:n);                  % downsampling d
    a=a(2:2:n);                  % downsampling a
    H0=H0(1:2:length(H0));        % subsampling H0
    H1=H1(1:2:length(H1));        % subsampling H1
    w=[d,w];                      % concatenate DWT coefficients
    n=n/2;
end
w=[a w];                                         % residual in scale space V_0
end

```

The input of the inverse DWT function include a vector w for the DWT coefficients and a vector h for the father wavelet coefficients $h_0[k]$, and the output is a vector y for the reconstructed signal x .

```

function y=myidwt(w,h)
K=length(h);
N=length(w);
n=log2(N);
h=h/norm(h);                                     % normalize h
h0=zeros(1,N); h0(1:K)=h; H0=fft(h0);
for k=0:N-1
    l=mod(k-N/2,N);
    H1(k+1)=exp(-j*2*pi*k/N)*conj(H0(l+1));
end
G0=conj(H0); G1=conj(H1); % synthesis filters
i=0;
while 2^i<K
    i=i+1;           % starting scale based on filter length
end

```

```

n=2^(i-1);
a=w(1:n);
while n<N
    d=w(n+1:2*n);           % get detail
    a=upsample(a,2,1);       % upsampling a
    d=upsample(d,2,1);       % upsampling d
    if n==1
        a=a'; d=d';         % upsampling 1x1 is column vector
    end
    n=2*n;                  % signal size is doubled
    A=fft(a).*G0(1:N/n:N); % convolve a with subsampled G0
    D=fft(d).*G1(1:N/n:N); % convolve d with subsampled G1
    a=real(ifft(A));
    d=real(ifft(D));
    a=a+d;
end
y=a;
end

```

Example 11.5: The DWT of an 8-point signal vector $\mathbf{x} = [0, 0, 2, 3, 4, 0, 0, 0]^T$ can be obtained by the code above. Depending on the wavelet functions used different DWT coefficients will be generated. When the Haar wavelets are used, the output is exactly the same as Eq.8.81 obtained by the discrete Haar transform:

$$\mathbf{X} = \mathbf{H}^T \mathbf{x} = [3.18, 0.35, -2.50, 2.0, 0.0, -0.71, 2.83, 0.0]^T \quad (11.154)$$

But when Daubechies' wavelets are used, we get a different set of DWT coefficients:

$$\mathbf{X} = [0.91, 3.60, -1.84, 2.65, 0.84, -0.65, 1.93, 0.000]^T \quad (11.155)$$

In either case, the signal is perfectly reconstructed by the inverse DWT.

11.3.2 Two-Dimensional DWT

Similar to all orthogonal transforms previously discussed, the discrete wavelet transform can also be extended to a 2-D transform that can be applied to 2-D signals such as an image. To do so, we first extend the 1-D two-channel filter bank shown in Fig.11.14 to basic 2-D filter bank, as shown in Fig.11.15, where the left half is the analysis filter bank for signal decomposition while the right half is the synthesis filter bank for signal reconstruction. The input of the analysis filter bank is a 2-D array treated as the coefficients $X_\phi[j]$ at the previous scale level j . We first carry out both low-pass (LP) and high-pass (HP) filtering corresponding to H_0 and H_1 , respectively, on each of the N columns of this array (vertical

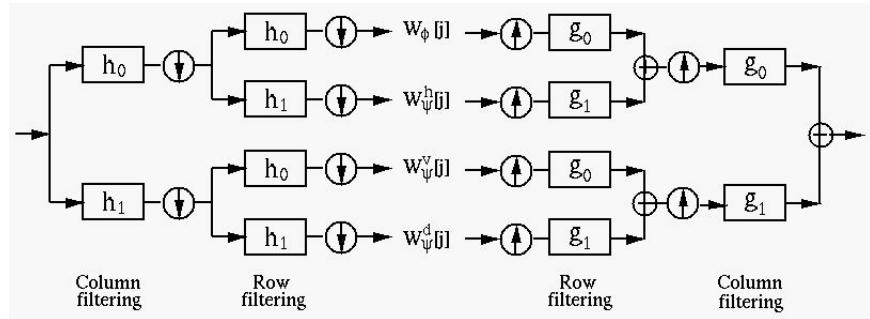


Figure 11.15 2-D two-channel filter bank

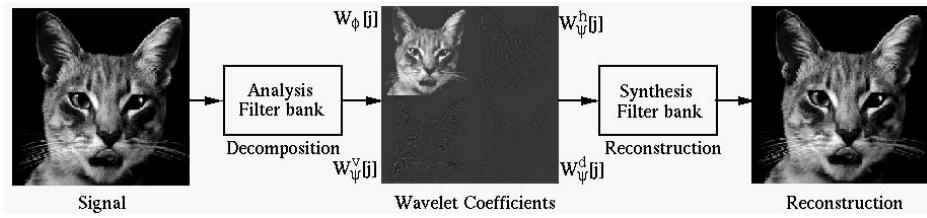


Figure 11.16 Signal decomposition and reconstruction by 2-D two-channel filter bank

filtering), and then, after downsampling, we carry out the same filtering on the rows of the resulting array (horizontal filtering). The outcomes of this two-stage filtering process are four sets of coefficients at the next lower scale level $j - 1$, including $W_\phi[j - 1]$, LP-filtered in both vertical and horizontal directions, $W_\psi^h[j - 1]$, LP-filtered vertically but HP-filtered horizontally, $W_\psi^v[j - 1]$, HP-filtered vertically but LP-filtered horizontally, and $W_\psi^d[j - 1]$, HP-filtered in both directions. These four sets of coefficients, each one quarter of the original size of the input 2-D array, are stored as the upper-left, upper-right, lower-left and lower-right quarters of a 2-D array, respectively. Same as in the 1-D case, the synthesis filter bank on the right of Fig.11.15 reverses the process to generate a perfect reconstruction of the input signal as the output. An example of the decomposition and reconstruction carried out by this 2-D two-channel filter bank is shown in Fig.11.16.

This two-stage filtering-downsampling operation can be applied to $W_\phi[j - 1]$, one of the four sets of coefficients that is LP-filtered in both directions and stored in the top-left quarter of the array, to generate the four sets of coefficients at the next lower scale level $j - 2$, as illustrated in Fig.11.17. Moreover, similar to the hierarchical process shown in Fig.11.12, this process can be carried out recursively, until, if needed, the lowest possible scale level is reached. If the input data is an $N \times N$ 2-D array, then the 2-D DWT coefficients at any scale level, including the final and lowest level, is also an $N \times N$ matrix. For example, the 2-D DWT coefficients obtained at each of four consecutive iterations of the 2-D DWT recursion are shown in Fig.11.18. Same as in the case of 1-D DWT, the

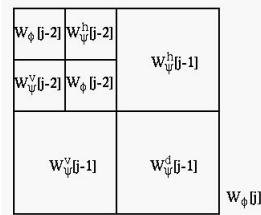


Figure 11.17 Recursion of 2-D discrete wavelet transform

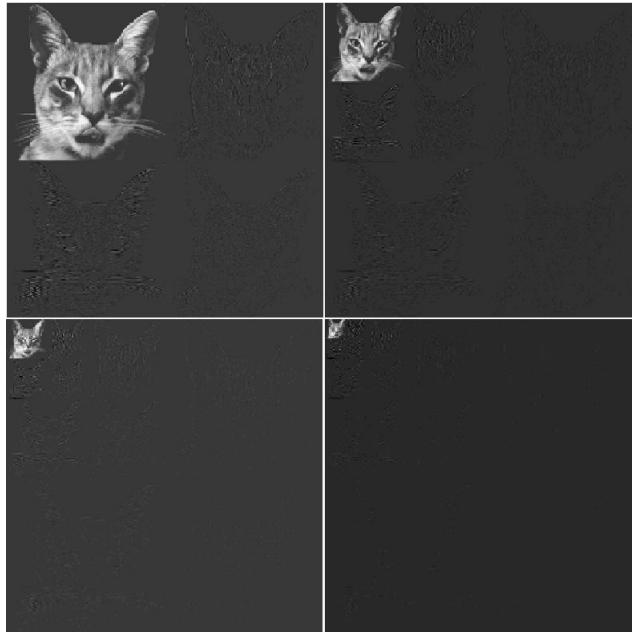


Figure 11.18 2-D DWT coefficients obtained at four consecutive stages

2-D DWT iteration can terminate at any of these scale levels, at which the data can always be perfectly reconstructed by the inverse transform.

Note that the 2-D array composed of the DWT coefficients is similar to the spectrum of most 2-D orthogonal transforms (except DFT), such as the DCT, in the sense that the coefficients around the top-left and lower-right corners represent respectively the signal components of the lowest and highest scale levels, corresponding to the lowest and highest frequency components in the DCT. The 2-D DWT coefficients can therefore be filtered (HP, LP, BP, etc.), similar to the filtering of the 2-D DCT spectrum.

The Matlab code for both the forward and inverse 2-D DWT transform is listed below. The input of the forward DWT function includes a 2-D array x for the signal, such as an image, and a vector h for the father wavelet filter

coefficients $h_0[k]$, and the output is a 2D array w of the same size as the input array for the DWT coefficients.

```

function w=dwt2d(x,h)
K=length(h);
[M,N]=size(x);
if M~=N
    error('Input should be a square array');
end
if K>N
    error('Data size should be larger than size of filter');
end
n=log2(N);
if n~=int16(n)
    error('Length of data x should be power of 2');
end
h0=zeros(1,N);
h0(1:K)=h;
H0=fft(h0);
for k=0:N-1
    l=mod(k-N/2,N);
    H1(k+1)=-exp(-j*2*pi*k/N)*conj(H0(l+1));
end
a=x;
w=zeros(N);
n=length(a);
while n>=K
    t=zeros(n,n);
    for k=1:n % for all n columns
        A=fft(double(a(:,k)));
        D=real(ifft(A.*H1'));
        A=real(ifft(A.*H0'));
        t(:,k)=[A(2:2:n); D(2:2:n)]; % save filtered column
    end
    for k=1:n % for all n rows
        A=fft(t(k,:));
        D=real(ifft(A.*H1));
        A=real(ifft(A.*H0));
        t(k,:)=[A(2:2:n) D(2:2:n)]; % save filtered row
    end
    w(1:n,1:n)=t; % concatenate coefficients
    H0=H0(1:2:length(H0)); % subsampling H0
    H1=H1(1:2:length(H1)); % subsampling H1
    n=n/2; % size of the next level
end

```

```

a=t(1:n,1:n);                                % up-left quarter as input
end

```

The inputs of the inverse DWT function include a 2-D array w for the 2D DWT coefficients and a vector h for the father wavelet coefficients $h_0[k]$, and the output is a 2D array y for the reconstruction of the input data array.

```

function y=idwt2d(w,h)
K=length(h);
N=length(w);
n=log2(N);
h=h/norm(h);                               % normalize h
h0=zeros(1,N); h0(1:K)=h; H0=fft(h0);
for k=0:N-1
    l=mod(k-N/2,N);
    H1(k+1)=-exp(-j*2*pi*k/N)*conj(H0(l+1));
end
G0=conj(H0); G1=conj(H1); % synthesis filters
i=0;
while 2^i<N
    i=i+1;                      % starting scale based on filter length
end
n=2^(i-1);                         % signal size of initial scale
y=w;
t=y(1:n,1:n);
while n<N
    g0=G0(1:N/(2*n):N);
    g1=G1(1:N/(2*n):N);
    for k=1:n                      % filtering n rows
        % rows in top half:
        a=upsample(y(k,1:n),2,1);    % approximate
        d=upsample(y(k,n+1:2*n),2,1); % detail
        A=fft(a).*g0;                % convolve a with G0
        D=fft(d).*g1;                % convolve d with G1
        y(k,1:2*n)=real(ifft(A)+ifft(D));
        % rows in bottom half:
        a=upsample(y(n+k,1:n),2,1);    % approximate
        d=upsample(y(n+k,n+1:2*n),2,1); % detail
        A=fft(a).*g0;                % convolve a with G0
        D=fft(d).*g1;                % convolve d with G1
        y(n+k,1:2*n)=real(ifft(A)+ifft(D));
    end
    for k=1:2*n                      % filtering 2n columns
        a=upsample(y(1:n,k),2,1);    % top half
    end
end

```

```

d=upsample(y(n+1:2*n,k),2,1); % bottom half
A=fft(a).*g0'; % convolve a with G0
D=fft(d).*g1'; % convolve d with G1
y(1:2*n,k)=real(ifft(A)+ifft(D))/2;
end
n=n*2;
end

```

11.4 Applications in Filtering and Compression

Example 11.6: Consider a set of signals, denoted by \mathbf{x} , as shown in the 1st and 3rd columns (dashed curves) of Fig.11.19, and their DCT and DWT (Daubechies D6) coefficients, generically denoted by $\mathbf{X} = \mathcal{T}[\mathbf{x}]$, as shown in the 2nd and 4th columns of the figure. Compression is then carried out in both the DCT and DWT domains by suppressing to zero certain percentage (80% in this case) of the transform coefficients with lowest magnitudes. The compressed coefficients, denoted by \mathbf{X}' , are shown as the solid curves in the 2nd and 4th columns, in comparison with the original ones (dashed curves). Finally, the signals are reconstructed by inverse transforms of the modified coefficients to get $\mathbf{x}' = \mathcal{T}^{-1}[\mathbf{X}']$, shown as the solid curves in the 1st and 3rd columns, in comparison with the original signals.

The performance of the DCT and DWT when used for compression can be evaluated in terms of both energy loss and signal error. As $\|\mathbf{x}\|^2 = \|\mathbf{X}\|^2$ and $\|\mathbf{x}'\|^2 = \|\mathbf{X}'\|^2$ (Parseval's identity), the percentage energy loss due to the compression can be found in either time or transform domain as:

$$\frac{\|\mathbf{X}\|^2 - \|\mathbf{X}'\|^2}{\|\mathbf{X}\|^2} = \frac{\|\mathbf{x}\|^2 - \|\mathbf{x}'\|^2}{\|\mathbf{x}\|^2} \quad (11.156)$$

On the other hand, the percentage signal error caused by the compression can also be defined as:

$$\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{\|\mathbf{x}\|^2} \quad (11.157)$$

It can be shown (see homework) that the signal error happens to be the same as energy loss, $\|\mathbf{x} - \mathbf{x}'\|^2 = \|\mathbf{x}\|^2 - \|\mathbf{x}'\|^2$, in this case.

The signal error depend on the transform method used, as well as the specific signal, as listed in Table 11.6. We see that the DCT and DWT are each good at representing certain types of signals. For example, the DCT is effective for the sinusoidal signals such as in cases 1, 2 and 3, while the DWT is effective for non-periodic and spiky signals such as in cases 4, 5 and 8. Note in particular that the DWT is especially effective at representing irregular spiky signals. Also,

neither transform method can represent the random noise as it is close to a white noise with energy relatively evenly distributed over all components in either DCT or DWT domain. It is also interesting to compare the errors with D4 and D6 wavelets in cases 1 and 7. When compared with D4, D6 performs better in case 1 of a smooth sinusoid, but worse in case 7 of a square wave. In general, as D6 is smoother than D4, it is more effective than D4 to represent smooth signals but less so for signals with discontinuities.

	Signal type	Percentage Error		
		DCT	D4	D6
1	Sinusoid	0.00	0.56	0.11
2	Two-tone sinusoids	2.23	9.57	10.17
3	Decaying sinusoid	0.08	4.00	2.01
4	Chirp	24.39	16.64	14.99
5	Sawtooth	2.12	0.00	0.16
6	Triangle	0.00	0.00	0.00
7	Square wave	1.05	0.31	1.82
8	Impulses	42.31	1.90	3.86
9	Random noise	35.82	41.01	40.83

Example 11.7: A piecewise linear signal (1st row in Fig.11.20) is contaminated by some random noise (2nd row). Two different types of filtering are then applied to remove the noise as much as possible, based on the DWT, (first four rows in the figure) as well as the DCT (last four rows) for comparison.

- Low-pass filtering is first carried out to remove the lower 7/8 of the coefficients after either the DCT or DWT, and then the filtered signal is reconstructed by inverse transform, as shown in the 3rd and 7th rows, respectively. While the high frequency noise is significantly reduced, the original signal is also distorted due to removal of the high frequency or high scale level components in the signal.
- Thresholding filtering is then carried out to remove all transform coefficients with values lower than a threshold (0.2 in this example), as shown in the 4th and 8th rows for the DCT and DWT filtering, respectively. We see that filtering based on the DWT removes more noise than that based on the DCT, due to the fact that in the transform domain, the signal is better separated from the noise by the DWT, while they are completely mixed together in the DCT spectrum. Comparing the 1st and 2nd rows on the right for the DCT coefficients, we see that the high frequency components of the signal are mixed those of the noise, while the same comparison of the 5th and 6th rows for the DWT coefficients shows that the signal components have more concentrated energy compared to those of the noise, allowing them to be better separated. Further comparison of the DWT and DCT representation of this specific piece-

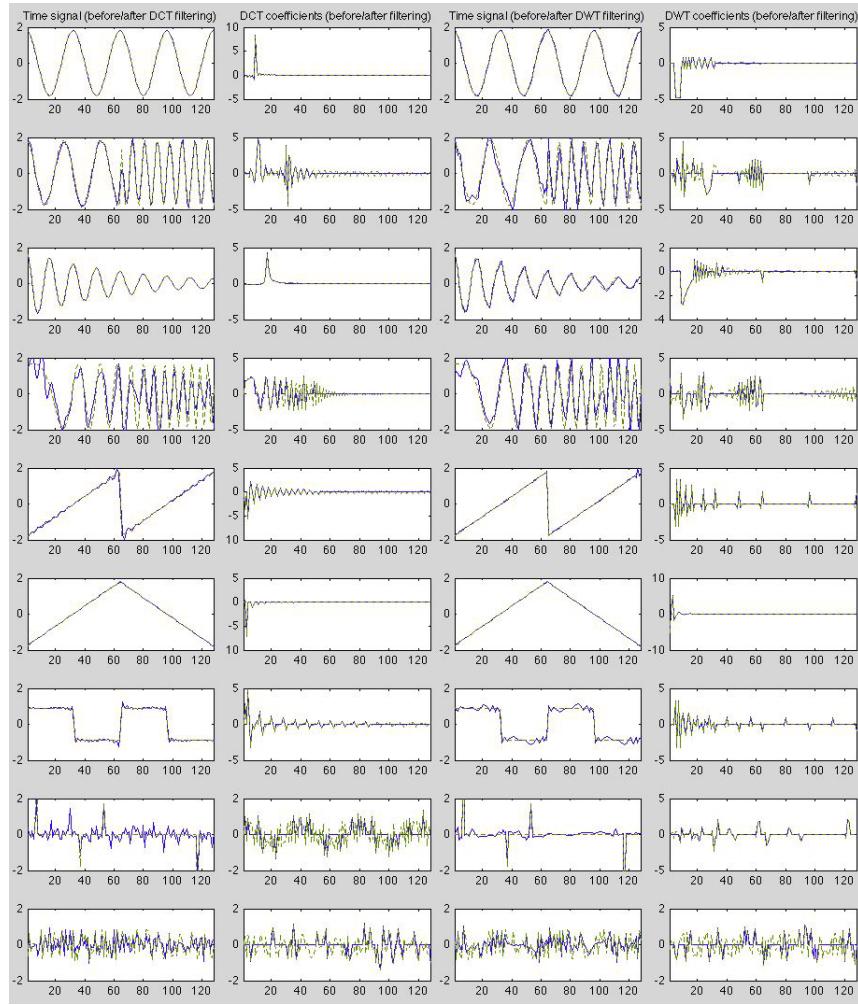


Figure 11.19 Compression of some typical signals by DCT (left) and DWT (right)

The 1st and 3rd columns show the time signals compared with their reconstructions based on only 20% of the transform coefficients, as shown in the 2nd and 4th columns for the DCT and DWT, respectively. In both time and transform domains, the signals before (dashed curves) and after (solid curves) the compression are shown for comparison.

wise linear signal in the 1st and 5th rows reveals that the signal can be much more efficiently represented by the DWT rather than the DCT, as many fewer coefficients are needed to in the representation by the DWT, also indicating better compression rate can be achieved by the DWT.

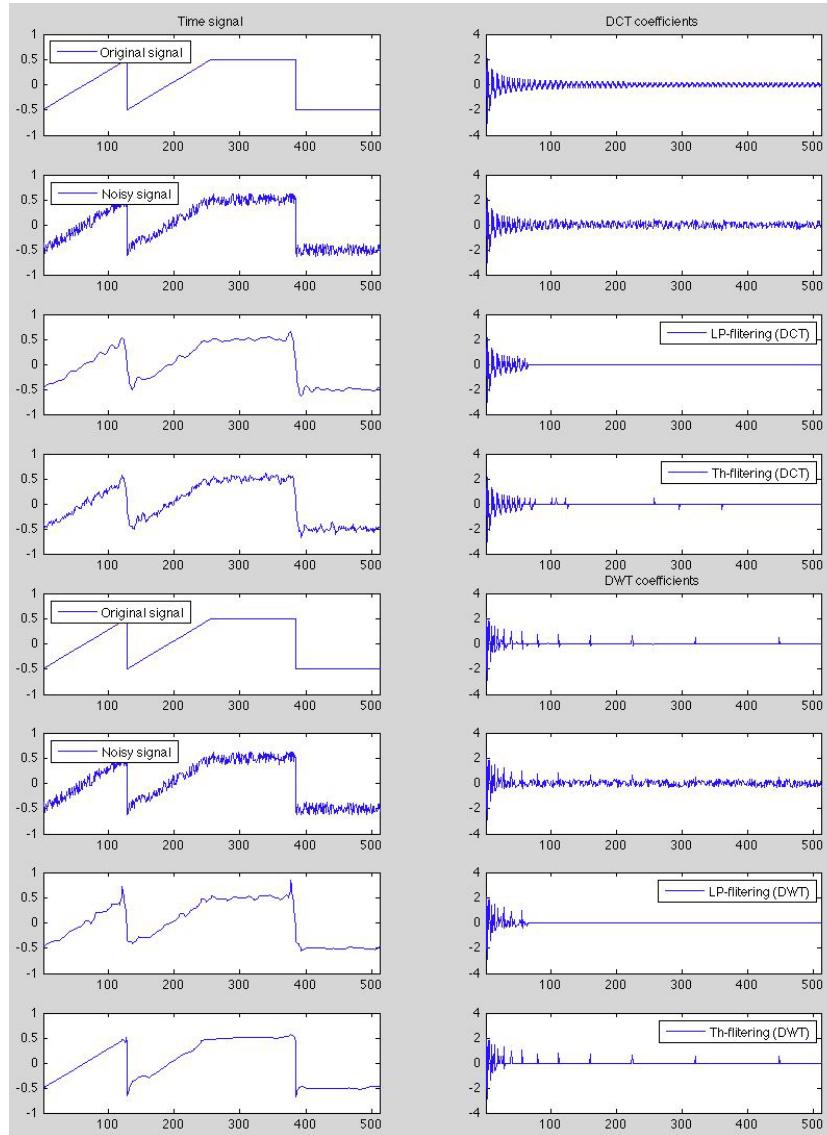


Figure 11.20 Signal filtering based on DCT and DWT

The time signals are shown on the left while the corresponding transform (DWT and DCT) coefficients are shown on the right. The top four rows are for the DCT while the bottom four are for the DWT. A nonlinear mapping $y = x^{0.6}$ is used to plot the coefficients in transform domain for the low values to be better seen.

Example 11.8: In Fig.11.21, the first two rows show the images of Lenna, both the original (1st row) and contaminated by white noise (2nd row), together with their 2-D DCT (middle) and DWT (right) spectra. The third row shows the

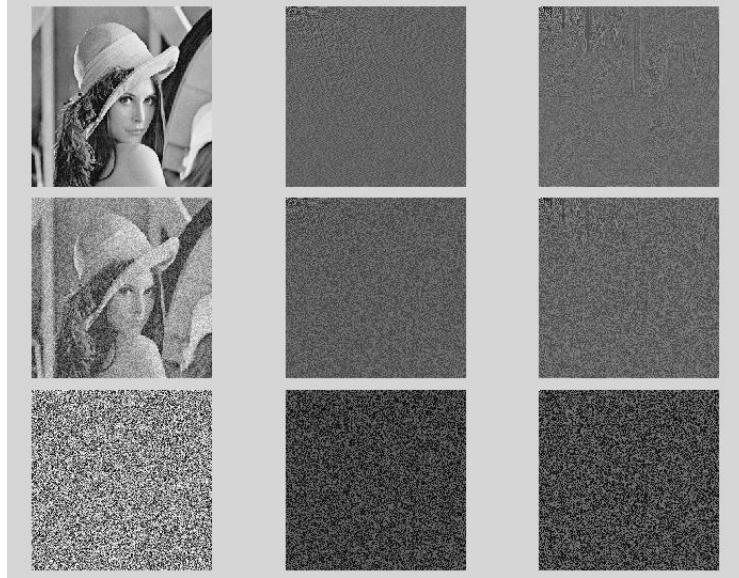


Figure 11.21 The image of Lenna and its DCT and DWT spectra

The first row shows the image and its DCT and DWT spectra, while the second row shows the same but contaminated by white noise and its DCT and DWT spectra.

noise and its DCT and DWT spectra. We see that the noise is indeed white as its energy is relatively evenly distributed over the entire frequency domain. To remove the noise, two different types of filtering are carried out in the transform domains, as shown in Fig.11.22. First, we use an ideal low-pass filter to remove all frequencies higher than a given cut-off frequency, i.e., farther than a specified distance away from the DC component (top-left corner) in the spectrum. Then the image is reconstructed based on the filtered spectrum. The results are shown in columns 1 for DCT and 3 for DWT. Next, we remove 98% of the frequency components to keep only the remaining 2% components carrying maximum possible energy, as shown in columns 2 for DCT and 4 for DWT. By visual inspection we see that the DWT filtered image is obviously better in terms of both image details and remaining amount of noise.

Example 11.9:

The same image of Lenna is transformed by each of seven different 2-D transform methods considered through out the book, the discrete Fourier (DFT), cosine (DCT), sine (DST), Walsh-Hadamard (WHT), slant (SLT) and discrete Haar (DHT) transforms, as well as the discrete wavelet transform (DWT), generically denoted by \mathcal{T} , resulting seven spectra $\mathbf{X} = \mathcal{T}[\mathbf{x}]$, as shown on the left in



Figure 11.22 Filtering of image Lenna in DCT and DWT domains

The first row shows the filtered spectra by DCT (1st and 2nd) and DWT (3rd and 4th), while the second row shows the corresponding images reconstructed by inverse DCT and DWT.

Fig.11.23. Moreover, for the purpose of compression, 99.5% of the coefficients in each of the spectra \mathbf{X} is suppressed with only 0.5% of the coefficients of the greatest magnitudes kept (1 to 200 compression rate). The modified spectra, denoted by \mathbf{Y} , are shown in the middle of the figure. Then the corresponding inverse transform is carried out to reconstruct the image as $\mathbf{y} = \mathcal{T}^{-1}[\mathbf{X}]$, as shown on the right of the figure. The compression results based on these different transform methods can be evaluated both subjectively by visual inspection and numerically by the relative error. Same as in the 1-D compression considered in Example 11.6, the energy loss due to the compression is the same as the signal error:

$$\frac{\|\mathbf{X}\|^2 - \|\mathbf{Y}\|^2}{\|\mathbf{X}\|^2} = \frac{\|\mathbf{x}\|^2 - \|\mathbf{y}\|^2}{\|\mathbf{x}\|^2} = \frac{\|\mathbf{x} - \mathbf{y}\|^2}{\|\mathbf{x}\|^2} \quad (11.158)$$

The same compression is also carried out for two other images of the cat and panda. The compression results in terms of the signal error are summarized in Table 11.9, from which we see that the error depends on the specific transform method used as well as the data being processed. For all three images, the DCT and DWT always have the lowest error among all methods. Moreover, based on visual inspection of the compressed images, we see that the compressed image reconstructed by the DWT method always looks the best, even when its error is slightly higher than that of the DCT.

Transform method	Percentage Error		
	Panda	Cat	Lenna
DFT	1.84	9.24	2.52
DCT	1.47	7.69	2.19
DST	2.60	8.06	3.07
WHT	2.19	10.98	2.99
SLT	1.78	9.46	2.57
DHT	2.13	9.76	2.63
DWT	1.73	7.59	2.38

11.5 Homework Problems

1. Here we consider the Haar wavelet transform as illustrated in Examples 11.1 and 11.2.
 - a. Verify that all properties of the scaling and wavelet filters (Eqs.11.56, 11.60, 11.58, and 11.63) are satisfied by the scaling and wavelet filters of Haar transform.
 - b. Based on Eqs.11.27 and 11.81, find the spectra $\Phi(f) = \mathcal{F}[\phi(t)]$ and $\Psi(f) = \mathcal{F}[\psi(t)]$.
 - c. Verify that Eqs.11.23 and 11.52 hold.
2. Reconsider the two-channel filter bank shown in Fig.11.14 but now using the z-transform as the analysis tool. Design the filter for perfect reconstruction (PR) by following the steps below.
 - a. Show that the output of the 2-channel filter is:

$$\begin{aligned}
 \hat{X}(z) &= G_0(z)A(z^2) + G_1(z)D(z^2) \\
 &= \frac{1}{2}[G_0(z)H_0(z) + G_1(z)H_1(z)] X(z) \\
 &\quad + \frac{1}{2}[G_0(z)H_0(-z) + G_1(z)H_1(-z)] X(-z)
 \end{aligned} \tag{11.159}$$

- b. For the 2-channel filter bank to achieve perfect reconstruction, its output $\hat{x}[n]$ has to be identical to the input $x[n]$, with a delay of m samples, i.e., $\hat{x}[n] = x[n - m]$, or $\hat{X}(z) = X(z)z^{-m}$ in z-domain. Given filters $H_0(z)$ and $H_1(z)$, find $G_0(z)$ and $G_1(z)$ for perfect reconstruction.
 Hint: for PR we let: $G_0(z)H_0(z) + G_1(z)H_1(z) = 2z^{-m}$ and $G_0(z)H_0(-z) + G_1(z)H_1(-z) = 0$.
 - c. For convenience, we set

$$\Delta(z) = H_0(z)H_1(-z) - H_0(-z)H_1(z) = 2z^{-m} \tag{11.160}$$

Show that $G_0(z)$ and $G_1(z)$ obtained above can be expressed as:

$$G_0(z) = H_1(-z), \quad G_1(z) = -H_0(-z) \tag{11.161}$$

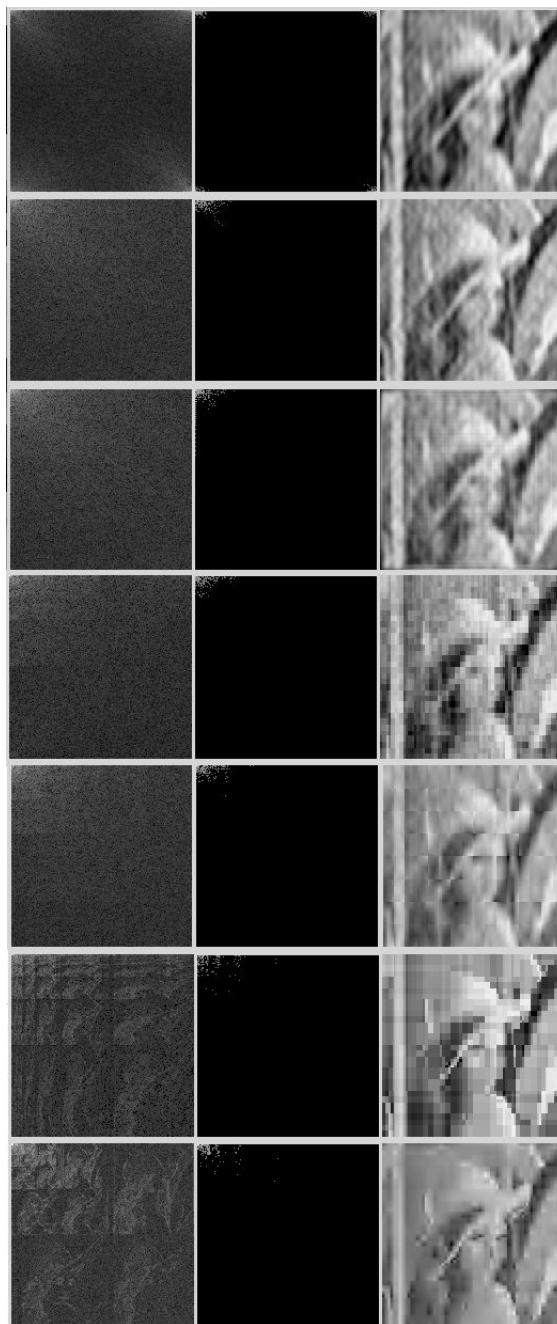


Figure 11.23 Image compression based on various orthogonal transforms
Seven different transform methods are used including, from top down, DFT, DCT, DST, WHT, SLT, DHT and DWT. The spectra and their compressed versions are shown on the left and middle columns (A nonlinear mapping $y = x^{0.3}$ is applied for the coefficients of low magnitude to be visible. Also, as the DFT is a complex transform, the spectra shown here are for the magnitudes of the transform coefficients). The reconstructed signal based on the compressed spectra are shown in the right column.

What do these relationships mean in time domain in terms of the filter coefficients $g_0[n]$ and $g_1[n]$ given $h_0[n]$ and $h_1[n]$? (Hint: Consider Eq.6.212) Is the function $\Delta(z)$ given in Eq.11.160 even, odd, or neither? Is m an even or odd number?

3. Obtain the coefficients of the 4-tap Daubechies filters by following the steps below.

- a. Derive the following identity:

$$\begin{aligned} 1 &= [\cos^2(\pi f) + \sin^2(\pi f)]^3 \\ &= \cos^6(\pi f) + 3\cos^4(\pi f)\sin^2(\pi f) + 3\sin^2(\pi f + \pi/2)\cos^4(\pi f + \pi/2) \\ &\quad + \cos^6(\pi f + \pi/2) \end{aligned} \quad (11.162)$$

- b. Define:

$$|H_0(f)|^2 = 2[\cos^6(\pi f) + 3\cos^4(\pi f)\sin^2(\pi f)] \quad (11.163)$$

Show both the normalization and orthogonality properties of a scaling filter given in Eq.11.58 and 11.63 are satisfied by this $H_0(f)$, i.e., it can indeed be used as a scaling filter, as the notation suggested.

- c. Find $H_0(f)$ by taking square root of $|H_0(f)|^2$, which can be written as:

$$\begin{aligned} |H_0(f)|^2 &= 2\cos^4(\pi f)[(\cos(\pi f))^2 + (\sqrt{3}\sin(\pi f))^2] \\ &= 2\cos^4(\pi f) |\cos^2(\pi f) + j\sqrt{3}\sin^2(\pi f)|^2 \end{aligned} \quad (11.164)$$

Express the result in the form of a 3rd order polynomial of $e^{-j2\pi kf}$. Verify the four coefficients are indeed the coefficients for the Daubechies scaling filter of $N = 2$.

4. Obtain the coefficients of the 4-tap Daubechies filters for the 2-channel filter bank with perfect reconstruction by following the steps below.

- a. Define $Q(z) = a_0 + a_1z^{-1} + a_2z^{-2}$ (coefficients a_0 , a_1 and a_2 to be determined) and choose:

$$H_0(z)G_0(z) = (1 + z^{-1})^{2N}Q(z) = (1 + z^{-1})^4Q(z) \quad (11.165)$$

where we have chosen $N = 2$. Write $H_0(z)G_0(z)$ and $H_0(-z)G_0(-z)$ as a polynomial of z^{-1} , and show $\Delta(z)$ in Eq.11.160 can be written as

$$\Delta(z) = 2(4a_0 + a_1)z^{-1} + 2(4a_0 + 6a_1 + 4a_2)z^{-3} + 2(a_1 + 4a_2)z^{-5} = 2z^{-m} \quad (11.166)$$

- b. Determine the coefficients a_0 , a_1 and a_2 by choosing to keep only the term of z^{-3} in $\Delta(z)$ above, i.e.,

$$4a_0 + a_1 = 0; \quad 4a_0 + 6a_1 + 4a_2 = 1, \quad a_1 + 4a_2 = 0 \quad (11.167)$$

Solve these equations to find a_0 , a_1 , a_2 , and show

$$Q(z) = a_0 + a_1z^{-1} + a_2z^{-2} = \frac{z^{-1}}{16}(-z + 4 - z^{-1}) \quad (11.168)$$

- c. Show that the term $-z + 4 - z^{-1}$ in $Q(z)$ obtained above can be factored to become:

$$-z + 4 - z^{-1} = (a + bz)(a + bz^{-1}) \quad (11.169)$$

Find the two coefficients a and b .

- d. Given the coefficients a and b , show that $Q(z)$ can be written as:

$$Q(z) = \frac{z^{-1}}{32} [(1 + \sqrt{3}) + (1 - \sqrt{3})z][(1 + \sqrt{3}) + (1 - \sqrt{3})z^{-1}] \quad (11.170)$$

and $H_0(z)G_0(z)$ can be written as a product:

$$\left[\frac{(1 + z^{-1})^2(1 + \sqrt{3}) + (1 - \sqrt{3})z^{-1}}{4\sqrt{2}} \right] \left[\frac{z^{-3}(1 + z)^2(1 + \sqrt{3}) + (1 - \sqrt{3})z}{4\sqrt{2}} \right] \quad (11.171)$$

which is actually a product of the following two 3rd order polynomials of z^{-1} :

$$H_0(z) = \frac{1}{4\sqrt{2}} [(1 + \sqrt{3}) + (3 + \sqrt{3})z^{-1} + (3 - \sqrt{3})z^{-2} + (1 - \sqrt{3})z^{-3}]$$

$$G_0(z) = \frac{1}{4\sqrt{2}} [(1 - \sqrt{3}) + (3 - \sqrt{3})z^{-1} + (3 + \sqrt{3})z^{-2} + (1 + \sqrt{3})z^{-3}]$$

As $H_0(z) = \sum_n h_0[n]z^{-n}$, we see that the coefficients $h_0[n]$ are

$$h_0[0] = \frac{1 + \sqrt{3}}{4\sqrt{2}}, h_0[1] = \frac{3 + \sqrt{3}}{4\sqrt{2}}, h_0[2] = \frac{3 - \sqrt{3}}{4\sqrt{2}}, h_0[3] = \frac{1 - \sqrt{3}}{4\sqrt{2}}$$

same as those given in Eq.11.111.

- e. Find $H_1(z)$ and $G_1(z)$ according to Eqs.11.161. These four filters $H_0(z)$, $G_0(z)$, $H_1(z)$ and $G_1(z)$ form an orthonormal filter bank with perfect reconstruction and leads to the Daubechies D_4 wavelets.
5. Obtain the six coefficients $h_0[k]$ ($k = 0, \dots, 5$) of the Daubechies scaling filter of order $N = 3$. Verify that they are the same as those given in Eq. 11.114. Revise the Matlab code provided to construct the scaling and wavelet functions $\phi(t)$ and $\psi(t)$ of order $N = 3$.
6. Prove that the energy loss in Eq.11.156 and signal error in Eq.11.157 in Example 11.6 are the same, i.e.,

$$\|\mathbf{X}\|^2 - \|\mathbf{Y}\|^2 = \|\mathbf{x}\|^2 - \|\mathbf{y}\|^2 = \|\mathbf{x} - \mathbf{y}\|^2 \quad (11.172)$$

Hint: As $\mathbf{X} = \mathcal{T}[\mathbf{x}]$ is an orthogonal transform, we have $\langle \mathbf{x}, \mathbf{y} \rangle = \langle \mathbf{X}, \mathbf{Y} \rangle$. Also the compression in the transform domain can be expressed as $Y[n] = c_n X[n]$ ($n = 0, \dots, N - 1$) where $c_n = 1$ if the n th coefficient $Y[n] = X[n]$ is kept during the compression, or $c_n = 0$ if it is suppressed to zero.

7. Compress each of the signals in Example 11.6 by suppressing 90% of the coefficients after each one of the orthogonal transform methods discussed throughout the book, including DFT, DCT, DST, WHT, DHT as well as DWT. Evaluate these methods quantitatively and qualitatively in terms of

- Percentage of signal energy contained in the remaining 10% of the transform coefficients.
 - Percentage error between reconstructed signal and the original.
 - Subjective comparison of the reconstructed signal and the original.
8. Repeat the previous problem on a set of different images of your choice, evaluate all of the orthogonal transform methods with the same quantitative and qualitative criteria.

12 Appendix 1: Review of Linear Algebra

12.1 Basic Definitions

- **Matrix**

An $m \times n$ matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ or $\mathbb{C}^{m \times n}$ is an array of m rows and n columns

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}_{m \times n} \quad (12.1)$$

where $a_{ij} \in \mathbb{R}$ or \mathbb{C} is the element in the i th (first index) row and j th (second index) column. In particular,

- if $m = n$, \mathbf{A} becomes a square matrix;
- if $m = 1$, \mathbf{A} becomes an n -dimensional (1 by n) row vector;
- if $n = 1$, \mathbf{A} becomes an m -dimensional (m by 1) column vector.

Through out the book, a vector \mathbf{a} is always assumed to be a column vector, unless otherwise specified. Sometimes it is convenient to express a matrix in terms of its column vectors

$$\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_n] \quad (12.2)$$

where \mathbf{a}_j ($j = 1, \dots, n$) is an m -dimensional column vector:

$$\mathbf{a}_j = \begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{bmatrix} \quad (12.3)$$

The i th row is an n -dimensional row vector $[a_{i1} \ a_{i2} \ \cdots \ a_{in}]$.

- **Transpose and Conjugate Transpose**

The *transpose* of an $m \times n$ matrix \mathbf{A} , denoted by \mathbf{A}^T , is an $n \times m$ matrix obtained by swapping elements a_{ij} and a_{ji} for all $i, j \in \{1, \dots, n\}$. In other words, the j th column of \mathbf{A} becomes the j th row of \mathbf{A}^T , and at the same time,

the i th row of \mathbf{A} becomes the i th column of \mathbf{A}^T :

$$\mathbf{A}^T = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n]^T = \begin{bmatrix} \mathbf{a}_1^T \\ \mathbf{a}_2^T \\ \vdots \\ \mathbf{a}_n^T \end{bmatrix} = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{m1} \\ a_{12} & a_{22} & \cdots & a_{m2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1n} & a_{2n} & \cdots & a_{mn} \end{bmatrix}_{n \times m} \quad (12.4)$$

where \mathbf{a}_j is the j th column of \mathbf{A} and its transpose \mathbf{a}_j^T is the j th row of \mathbf{A}^T :

$$\mathbf{a}_j^T = \begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{nj} \end{bmatrix}^T = [a_{1j}, a_{2j}, \dots, a_{nj}] \quad (12.5)$$

Here are some important properties related to transpose:

$$(\mathbf{A}^T)^T = \mathbf{A}, \quad (\mathbf{AB})^T = \mathbf{B}^T \mathbf{A}^T \quad (12.6)$$

The *conjugate transpose* of an $m \times n$ complex matrix \mathbf{A} , denoted by \mathbf{A}^* , is the complex conjugate of its transpose, i.e.,

$$\mathbf{A}^* = \overline{\mathbf{A}^T} = \overline{\mathbf{A}}^T \quad (12.7)$$

i.e., the element in the i th row and j th column of \mathbf{A}^* is the complex conjugate of the element in the j th row and i th column of \mathbf{A} . We obviously have:

$$(\mathbf{A}^*)^* = \mathbf{A}, \quad (\mathbf{AB})^* = \mathbf{B}^* \mathbf{A}^* \quad (12.8)$$

- **Identity Matrix**

The *identity matrix* \mathbf{I} is a special $n \times n$ square matrix with all elements being zero except those along the main diagonal which are 1:

$$\mathbf{I} = \text{diag}[1, \dots, 1] = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}_{n \times n} \quad (12.9)$$

The identity matrix can also be expressed in terms of its column vectors:

$$\mathbf{I} = [\mathbf{e}_1, \dots, \mathbf{e}_i, \dots, \mathbf{e}_n] \quad (12.10)$$

where \mathbf{e}_i ($i = 1, \dots, n$) is an n -dimensional column vector with all elements equal to zero except the i th one which is 1:

$$\mathbf{e}_i = [e_{1i}, \dots, e_{ni}]^T = [0, \dots, 0, 1, 0, \dots, 0]^T \quad (12.11)$$

i.e., the $e_{ij} = 0$ for all $i \neq j$ and $e_{ii} = 1$ for all $i = 1, \dots, n$.

- **Scalar Multiplication**

A matrix \mathbf{A} can be multiplied by a scalar c to get

$$c\mathbf{A} = c \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} = \begin{bmatrix} ca_{11} & ca_{12} & \cdots & ca_{1n} \\ ca_{21} & ca_{22} & \cdots & ca_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ ca_{m1} & ca_{m2} & \cdots & ca_{mn} \end{bmatrix} \quad (12.12)$$

- **Dot Product**

The *dot product*, also called *inner product*, of two real column vectors $\mathbf{x} = [x_1, \dots, x_n]^T$ and $\mathbf{y} = [y_1, \dots, y_n]^T$ is defined as

$$\mathbf{x} \cdot \mathbf{y} = \langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^T \bar{\mathbf{y}} = \mathbf{y}^* \mathbf{x} = [x_1, \dots, x_n] \begin{bmatrix} \bar{y}_1 \\ \vdots \\ \bar{y}_n \end{bmatrix} = \sum_{i=1}^n x_i \bar{y}_i \quad (12.13)$$

where $\overline{u+jv} = u - jv$ is complex conjugate of $u+jv$. If the inner product of \mathbf{x} and \mathbf{y} is zero, then the two vectors are said to be *orthogonal*, denoted by $\mathbf{x} \perp \mathbf{y}$. In particular when $\mathbf{x} = \mathbf{y}$, we have:

$$\langle \mathbf{x}, \mathbf{x} \rangle = \|\mathbf{x}\|^2 = \sum_{i=1}^n x_i \bar{x}_i = \sum_{i=1}^n |x_i|^2 > 0 \quad (12.14)$$

where

$$\|\mathbf{x}\| = \sqrt{\sum_{i=1}^n |x_i|^2} \quad (12.15)$$

is called the *norm* of \mathbf{x} . If $\|\mathbf{x}\| = 1$, \mathbf{x} is *normalized*.

- **Matrix Multiplication**

The product of an $m \times k$ matrix \mathbf{A} and a $k \times n$ matrix \mathbf{B} is

$$\mathbf{A}_{m \times k} \mathbf{B}_{k \times n} = \mathbf{C}_{m \times n} \quad (12.16)$$

where the element in the i th row and j th column of \mathbf{C} is the dot product of the i th row vector of \mathbf{A} and the j th column of \mathbf{B} :

$$c_{ij} = [a_{i1}, \dots, a_{ik}] \begin{bmatrix} b_{k1} \\ \vdots \\ b_{kn} \end{bmatrix} = \sum_{l=1}^k a_{il} b_{lj} \quad (12.17)$$

For this multiplication to be possible, the number of columns of \mathbf{A} must be equal to the number of rows of \mathbf{B} , so that the dot product can be carried out. Otherwise, the two matrices can not be multiplied.

- **Trace**

The *trace* of \mathbf{A} is defined as the sum of the element along the main diagonal:

$$\text{tr}(\mathbf{A}) = \sum_{i=1}^n a_{ii} \quad (12.18)$$

Here are some properties of the trace:

$$\operatorname{tr}(\mathbf{A} + \mathbf{B}) = \operatorname{tr}\mathbf{A} + \operatorname{tr}\mathbf{B}, \quad \operatorname{tr}(c\mathbf{A}) = c \operatorname{tr}\mathbf{A}, \quad \operatorname{tr}(\mathbf{AB}) = \operatorname{tr}(\mathbf{BA}) \quad (12.19)$$

- **Rank**

If none of a set of vectors can be expressed as a linear combination of the rest of the vectors, then these vectors are *linearly independent*. The *rank* of a matrix \mathbf{A} , denoted by $\operatorname{rank}\mathbf{A}$, is the maximum number of linearly independent columns of \mathbf{A} , which is the same as the maximum number of linearly independent rows. Obviously the rank of an m by n matrix is no larger than the smaller of m and n :

$$\operatorname{rank}\mathbf{A} \leq \min(m, n) \quad (12.20)$$

If the equation holds, matrix \mathbf{A} has a *full rank*.

- **Determinant**

The *determinant* of an $n \times n$ matrix \mathbf{A} , denoted by $\det\mathbf{A}$ or $|\mathbf{A}|$, is a scalar that can be recursively defined as

$$|\mathbf{A}| = \det\mathbf{A} = \sum_{j=1}^n (-1)^{j+1} a_{1j} \det\mathbf{A}_{1j} \quad (12.21)$$

where \mathbf{A}_{1j} is an $(n - 1) \times (n - 1)$ matrix obtained by deleting the first row and j th column of \mathbf{A} , and the determinant of a 1 by 1 matrix is $\det(a) = a$. If \mathbf{A} is not a full rank matrix, its determinant is 0.

In particular, when $n = 2$,

$$\det \begin{bmatrix} a & b \\ c & d \end{bmatrix} = ad - bc \quad (12.22)$$

and when $n = 3$,

$$\begin{aligned} \det \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} &= a \det \begin{bmatrix} e & f \\ h & i \end{bmatrix} - b \det \begin{bmatrix} d & f \\ g & i \end{bmatrix} + c \det \begin{bmatrix} d & e \\ g & h \end{bmatrix} \\ &= aei - afh - bdi + bfg + cdh - ceg = (aei + bfg + cdh) - (gec + hfa + idb) \end{aligned} \quad (12.23)$$

Here are some important properties of the determinant (here \mathbf{A} and \mathbf{B} are square matrices):

$$\det(\mathbf{AB}) = \det(\mathbf{BA}) = \det\mathbf{A} \det\mathbf{B}, \quad \det(\mathbf{A}^T) = \det\mathbf{A}, \quad \det(c\mathbf{A}) = c^n \det\mathbf{A} \quad (12.24)$$

- **Inverse Matrix**

If \mathbf{A} is an $n \times n$ square matrix and there exists another $n \times n$ matrix \mathbf{B} so that $\mathbf{AB} = \mathbf{BA} = \mathbf{I}$, then $\mathbf{B} = \mathbf{A}^{-1}$ is the *inverse* of \mathbf{A} , which can be obtained

by:

$$\mathbf{A}^{-1} = \frac{1}{\det \mathbf{A}} \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mn} \end{bmatrix}^T \quad (12.25)$$

where c_{ij} is the ij-th *cofactor* defined as

$$c_{ij} = (-1)^{i+j} \det \mathbf{\mu}_{ij} \quad (12.26)$$

with $\mathbf{\mu}_{ij}$ being an $(n-1) \times (n-1)$ *minor matrix* obtained by removing the ith row and jth column \mathbf{A} . Obviously if \mathbf{A} is not full rank matrix, $\det \mathbf{A} = 0$, then \mathbf{A}^{-1} does not exist.

The following statements are equivalent:

- \mathbf{A} is invertible, i.e., inverse matrix \mathbf{A}^{-1} exists.
- \mathbf{A} is full rank.
- $\det \mathbf{A} \neq 0$.
- All column and row vectors are linearly independent.
- All eigenvalues of \mathbf{A} are nonzero (to be discussed later).

These are some basic properties related to inverse of a matrix \mathbf{A} :

$$(\mathbf{A}^{-1})^{-1} = \mathbf{A}, \quad (c\mathbf{A})^{-1} = \frac{1}{c} \mathbf{A}^{-1}, \quad (\mathbf{AB})^{-1} = \mathbf{B}^{-1} \mathbf{A}^{-1}, \quad (\mathbf{A}^{-1})^T = (\mathbf{A}^T)^{-1} \quad (12.27)$$

• Pseudo-Inverse Matrix

Let \mathbf{A} be an $m \times n$ matrix. If $m \neq n$, then \mathbf{A} is not a square matrix and its inverse does not exist. However, we can find its *pseudo-inverse* \mathbf{A}^- , an $n \times m$ matrix, as shown below.

- If \mathbf{A} has more rows than columns, i.e., $m > n$, then

$$\mathbf{A}^- = (\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}^* \quad (12.28)$$

We can verify that $\mathbf{A}^- \mathbf{A} = \mathbf{I}$:

$$\mathbf{A}^- \mathbf{A} = (\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}^* \mathbf{A} = \mathbf{I}_{n \times n} \quad (12.29)$$

But $\mathbf{A} \mathbf{A}^- \neq \mathbf{I}$:

- If \mathbf{A} has more columns than rows, i.e., $m < n$, then

$$\mathbf{A}^- = \mathbf{A}^* (\mathbf{A} \mathbf{A}^*)^{-1} \quad (12.30)$$

We can verify that $\mathbf{A} \mathbf{A}^- = \mathbf{I}$:

$$\mathbf{A} \mathbf{A}^- = \mathbf{A} \mathbf{A}^* (\mathbf{A} \mathbf{A}^*)^{-1} = \mathbf{I}_{m \times m} \quad (12.31)$$

But $\mathbf{A}^- \mathbf{A} \neq \mathbf{I}$:

Note that the pseudo-inverses in Eq.12.28 ($m > n$) and Eq.12.30 ($m < n$) are essentially the same. Assume \mathbf{A} has more rows than columns ($m > n$), then another matrix defined as $\mathbf{B} = \mathbf{A}^*$ has more columns than rows. Taking

conjugate transpose on both sides of Eq.12.28, we get:

$$(\mathbf{A}^-)^* = [(\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}]^* = \mathbf{A} (\mathbf{A}^* \mathbf{A})^{-1} = (\mathbf{A}^*)^- \quad (12.32)$$

i.e.,

$$\mathbf{B}^- = \mathbf{B}^* (\mathbf{B} \mathbf{B}^*)^{-1} \quad (12.33)$$

which is the same as Eq.12.30.

We can also show that $(\mathbf{A}^-)^- = \mathbf{A}$. If $m > n$, then we have:

$$\begin{aligned} (\mathbf{A}^-)^- &= [(\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}]^- = [(\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}]^* [(\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}]^{*-1} \\ &= \mathbf{A} (\mathbf{A}^* \mathbf{A})^{-1} [(\mathbf{A}^* \mathbf{A})^{-1} \mathbf{A}^* \mathbf{A} (\mathbf{A}^* \mathbf{A})^{-1}]^{-1} \\ &= \mathbf{A} (\mathbf{A}^* \mathbf{A})^{-1} (\mathbf{A}^* \mathbf{A}) = \mathbf{A} \end{aligned} \quad (12.34)$$

Similarly we can show the same is true if $m < n$.

Specially when $m = n$, \mathbf{A} is invertible and the pseudo-inverse in either Eq.12.28 or Eq.12.30 becomes the regular inverse $\mathbf{A}^- = \mathbf{A}^{-1}$.

12.2 Eigenvalues and Eigenvectors

For any $n \times n$ matrix \mathbf{A} , if there exists an n by 1 vector ϕ and a scalar λ satisfying

$$\mathbf{A}_{n \times n} \phi_{n \times 1} = \lambda \phi_{n \times 1} \quad (12.35)$$

then λ and ϕ are called the *eigenvalue* and *eigenvector* of \mathbf{A} , respectively. To obtain λ , we rewrite the above equation as

$$(\lambda \mathbf{I} - \mathbf{A}) \phi = 0 \quad (12.36)$$

This is a homogeneous algebraic equation system of n equations for n unknowns, the elements in vector ϕ . This equation system has non-zero solutions if and only if

$$\det(\lambda \mathbf{I} - \mathbf{A}) = 0 \quad (12.37)$$

This nth order equation of λ is the *characteristic equation* of the matrix \mathbf{A} , which can be solved to get n solutions, the n eigenvalues $\{\lambda_1, \dots, \lambda_n\}$ of \mathbf{A} . Substituting each λ_i back into the equation system, we can obtain the non-zero solution, the eigenvector ϕ_i corresponding to eigenvalue λ_i :

$$\mathbf{A} \phi_i = \lambda_i \phi_i, \quad (i = 1, \dots, n) \quad (12.38)$$

Putting all n such equations together, we get

$$\mathbf{A} [\phi_1, \dots, \phi_n] = [\lambda_1 \phi_1, \dots, \lambda_n \phi_n] = [\phi_1, \dots, \phi_n] \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} \quad (12.39)$$

Defining

$$\Phi = [\phi_1, \dots, \phi_n], \quad \text{and} \quad \Lambda = \text{diag}[\lambda_1, \dots, \lambda_n] \quad (12.40)$$

we can write the equation above in a more compact form:

$$A\Phi = \Phi\Lambda, \quad \text{or} \quad \Phi^{-1}A\Phi = \Lambda \quad (12.41)$$

The trace and determinant of A can be obtained in terms of its eigenvalues

$$\text{tr } A = \sum_{k=1}^n \lambda_k \quad (12.42)$$

$$\det A = \prod_{k=1}^n \lambda_k \quad (12.43)$$

A^T has the same eigenvalues and eigenvectors as A :

$$A^T \phi_i = \lambda_i \phi_i, \quad (i = 1, \dots, n) \quad (12.44)$$

A^m has the same eigenvectors as A , but its eigenvalues are $\{\lambda_1^m, \dots, \lambda_n^m\}$:

$$A^m \phi_i = \lambda_i^m \phi_i, \quad (i = 1, \dots, n) \quad (12.45)$$

In particular, when $m = -1$, the eigenvalues become A^{-1} are $\{1/\lambda_1, \dots, 1/\lambda_n\}$:

$$A^{-1} \phi_i = \frac{1}{\lambda_i} \phi_i, \quad (i = 1, \dots, n) \quad (12.46)$$

A Hermitian matrix A is *positive definite*, denoted by $A > 0$, if and only if for any nonzero $x = [x_1, \dots, x_n]^T$, the quadratic form x^*Ax is greater than zero:

$$x^*Ax > 0 \quad (12.47)$$

In particular, if we let $x = \phi_i$ be the eigenvector corresponding to the i th eigenvalue λ_i , then the above becomes:

$$\phi_i^* A \phi_i = \lambda_i \phi_i^* \phi_i > 0 \quad (12.48)$$

as $\phi_i^* \phi_i = \|\phi_i\|^2 > 0$, we know $\lambda_i > 0$ for all $i = 1, \dots, n$, i.e., $A > 0$ if and only if all of its eigenvalues are greater than zero. Also, as the eigenvalues of A^{-1} are $1/\lambda_i$, $i = (1, \dots, n)$, we have $A > 0$ if and only if $A^{-1} > 0$.

12.3 Hermitian Matrix and Unitary Matrix

A matrix A is *Hermitian* if it is equal to its *conjugate transpose*:

$$A = \overline{A}^T = \overline{A^T} = A^* \quad (12.49)$$

In particular if a Hermitian matrix $\overline{A} = A$ is real, then it is *symmetric* $A = A^T$. All eigenvalues λ_i of a Hermitian matrix are real, and all eigenvectors ϕ_i corresponding to distinct eigenvalues are orthogonal. If the eigenvectors are normal-

ized with unit norm, then they are *orthonormal* (both orthogonal and normalized):

$$\langle \phi_i, \phi_j \rangle = \delta[i - j], \quad (i, j = 1, \dots, n) \quad (12.50)$$

A matrix \mathbf{A} is *unitary* if its conjugate transpose is equal to its inverse:

$$\mathbf{A}^* = \mathbf{A}^{-1}, \quad \text{i.e.} \quad \mathbf{A}^* \mathbf{A} = \mathbf{A} \mathbf{A}^* = \mathbf{I} \quad (12.51)$$

When a unitary matrix is real $\mathbf{A} = \overline{\mathbf{A}}$, then it is *orthogonal* $\mathbf{A}^T = \mathbf{A}^{-1}$. The absolute values of all eigenvalues (may be complex) of a unitary matrix are $|\lambda_i| = 1$, i.e. they lie on the unit circle centered at 0 in the complex plane. The determinant of a unitary matrix \mathbf{A} is $\det \mathbf{A} = \prod_{k=1}^n \lambda_k = pm1$.

Let $\Lambda = \text{diag}[\lambda_1, \dots, \lambda_n]$ and $\Phi = [\phi_1, \dots, \phi_n]$ be the eigenvalue and eigenvector matrices of a Hermitian matrix $\mathbf{A}^* = \mathbf{A}$. If all columns ϕ_i of Φ are orthonormal, then Φ is unitary satisfying:

$$\Phi^{-1} = \Phi^*, \quad \text{i.e.} \quad \Phi \Phi^* = \Phi^* \Phi = \mathbf{I} \quad (12.52)$$

and the eigenequation of the Hermitian matrix \mathbf{A} can be written as:

$$\mathbf{A}\Phi = \Phi\Lambda \quad (12.53)$$

i.e.,

$$\Phi^{-1} \mathbf{A} \Phi = \Phi^* \mathbf{A} \Phi = \Lambda, \quad \text{or} \quad \mathbf{A} = \Phi \Lambda \Phi^{-1} = \Phi \Lambda \Phi^* \quad (12.54)$$

From the first equation above we see that the Hermitian matrix \mathbf{A} can be diagonalized by its unitary eigenvector matrix Φ . From the second equation we see that the matrix \mathbf{A} can be decomposed to be expressed as:

$$\mathbf{A} = \Phi \Lambda \Phi^* = [\phi_1, \dots, \phi_n] \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n \end{bmatrix} \begin{bmatrix} \phi_1^* \\ \vdots \\ \phi_n^* \end{bmatrix} = \sum_{i=1}^n \lambda_i \phi_i \phi_i^* \quad (12.55)$$

Based on any unitary matrix $\mathbf{A} = [\mathbf{a}_1 \cdots, \mathbf{a}_n]$ (where the i th column vector is $\mathbf{a}_k = [a_{1k}, \dots, a_{nk}]^T$), a *unitary transform* of a vector $\mathbf{x} = [x_1, \dots, x_n]^T$ can be defined as:

$$\left\{ \begin{array}{l} \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \mathbf{A}^{-1} \mathbf{x} = \mathbf{A}^* \mathbf{x} = \begin{bmatrix} \mathbf{a}_1^* \\ \mathbf{a}_2^* \\ \vdots \\ \mathbf{a}_n^* \end{bmatrix} \mathbf{x} \\ \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \mathbf{A} \mathbf{y} = \begin{bmatrix} \mathbf{a}_1 & \mathbf{a}_2 & \cdots & \mathbf{a}_n \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \sum_{j=1}^n y_j \mathbf{a}_j \end{array} \right. \quad (12.56)$$

The first and second equations are respectively the forward and inverse transforms. In particular, when $\mathbf{A} = \overline{\mathbf{A}}$ is real, $\mathbf{A}^{-1} = \mathbf{A}^T$ is an orthogonal matrix and the corresponding transform is an *orthogonal transform*.

The forward transform can also be written in component form:

$$y_j = \langle \mathbf{x}, \mathbf{a}_j \rangle = \mathbf{a}_j^* \mathbf{x} = \sum_{i=1}^n x_i \bar{a}_{ij}, \quad (j = 1, \dots, n) \quad (12.57)$$

where the transform coefficient $y_i = \mathbf{a}_i^* \mathbf{x}$ represents the projection of \mathbf{x} onto the i th column vector \mathbf{a}_i of the transform matrix \mathbf{A} . The *inverse transform* can also be written as:

$$\mathbf{x} = \sum_{j=1}^n y_j \mathbf{a}_j \quad \text{or in component form: } x_i = \sum_{j=1}^n a_{ij} y_j \quad (i = 1, \dots, n) \quad (12.58)$$

By this transform, vector \mathbf{x} is represented as a linear combination (weighted sum) of the n column vectors $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n$ of matrix \mathbf{A} . Geometrically, \mathbf{x} is a point in the n -dimensional space spanned by these n orthonormal basis vectors. Each coefficient y_i is the coordinate in the i th dimension, which can be obtained as the projection of \mathbf{x} onto the corresponding basis vector \mathbf{a}_i .

A unitary (orthogonal) transform $\mathbf{y} = \mathbf{Ax}$ can be interpreted geometrically as the rotation of the vector X about the origin, or equivalently, the representation of the same vector in a rotated coordinate system. A unitary (orthogonal) transform $\mathbf{y} = \mathbf{Ax}$ does not change the vector's length:

$$\|\mathbf{y}\|^2 = \mathbf{y}^* \mathbf{y} = (\mathbf{A}^* \mathbf{x})^* (\mathbf{A}^* \mathbf{x}) = \mathbf{x}^* \mathbf{A} \mathbf{A}^* \mathbf{x} = \mathbf{x}^* \mathbf{x} = \|\mathbf{x}\|^2 \quad (12.59)$$

as $\mathbf{AA}^* = \mathbf{AA}^{-1} = \mathbf{I}$. This is the Parseval's relation. If \mathbf{x} is interpreted as a signal, then its length $\|\mathbf{x}\|^2 = \|\mathbf{y}\|^2$ represents the total energy or information contained in the signal, which is preserved during any unitary transform.

12.4 Toeplitz and Circulant Matrices

A square matrix is called a *Toeplitz matrix* if any element a_{mn} is equal to its lower-right neighbor a_{m+1n+1} , i.e., every diagonal of the matrix is composed of the same value. For example, the following matrix is a Toeplitz matrix:

$$A_T = \begin{bmatrix} a & b & c & d & e & f \\ g & a & b & c & d & e \\ h & g & a & b & c & d \\ i & h & g & a & b & c \\ j & i & h & g & a & b \\ k & j & i & h & g & a \end{bmatrix} \quad (12.60)$$

An $N \times N$ Toeplitz matrix can be formed by a sequence $\cdots a_{-2}, a_{-1}, a_0, a_1, a_2, \dots$:

$$A_T = \begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_{N-3} & a_{N-2} & a_{N-1} \\ a_{-1} & a_0 & a_1 & \cdots & a_{N-4} & a_{N-3} & a_{N-2} \\ a_{-2} & a_{-1} & a_0 & \cdots & a_{N-5} & a_{N-4} & a_{N-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ a_{3-N} & a_{4-N} & a_{5-N} & \cdots & a_0 & a_1 & a_2 \\ a_{2-N} & a_{3-N} & a_{4-N} & \cdots & a_{-1} & a_0 & a_1 \\ a_{1-N} & a_{2-N} & a_{3-N} & \cdots & a_{-2} & a_{-1} & a_0 \end{bmatrix} \quad (12.61)$$

In particular, if the sequence is periodic: $x_n = x_{n+N}$ with period N , then the Toeplitz matrix above becomes a *circulant matrix*, composed of N rows each rotated one element to the right relative to the previous row:

$$A_T = \begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_{N-3} & a_{N-2} & a_{N-1} \\ a_{N-1} & a_0 & a_1 & \cdots & a_{N-4} & a_{N-3} & a_{N-2} \\ a_{N-2} & a_{N-1} & a_0 & \cdots & a_{N-5} & a_{N-4} & a_{N-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ a_3 & a_4 & a_5 & \cdots & a_0 & a_1 & a_2 \\ a_2 & a_3 & a_4 & \cdots & a_{N-1} & a_0 & a_1 \\ a_1 & a_2 & a_3 & \cdots & a_{N-2} & a_{N-1} & a_0 \end{bmatrix} \quad (12.62)$$

When the period N of the sequence is increased to approach infinity $N \rightarrow \infty$, the periodic sequence approaches aperiodic, correspondingly, the circulant matrix asymptotically becomes a Toeplitz matrix.

12.5 Vector and Matrix Differentiation

Let $\mathbf{x} = [x_1, \dots, x_n]^T$ be an n-D vector composed of n variables x_k ($k = 1, \dots, n$). A vector differentiation operator is defined as

$$\frac{d}{d\mathbf{x}} = \left[\frac{\partial}{\partial x_1}, \dots, \frac{\partial}{\partial x_n} \right]^T \quad (12.63)$$

which can be applied to any scalar function $f(\mathbf{x})$ to find its derivative with respect to its variable argument \mathbf{x} :

$$\frac{d}{d\mathbf{x}} f(\mathbf{x}) = \left[\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n} \right]^T \quad (12.64)$$

Vector differentiation has the following properties:

-

$$\frac{d}{d\mathbf{x}} (\mathbf{b}^T \mathbf{x}) = \frac{d}{d\mathbf{x}} (\mathbf{x}^T \mathbf{b}) = \mathbf{b} \quad (12.65)$$

•

$$\frac{d}{dx}(\mathbf{x}^T \mathbf{x}) = 2\mathbf{x} \quad (12.66)$$

•

$$\frac{d}{dx}(\mathbf{x}^T \mathbf{A} \mathbf{x}) = 2\mathbf{A} \mathbf{x}, \quad (12.67)$$

where $\mathbf{A} = [a_{ij}]_{n \times n} = \mathbf{A}^T$ is an n by n symmetric matrix.

To show the third one, we first consider the k th element of the vector ($k = 1, \dots, n$):

$$\frac{\partial}{\partial x_k}(\mathbf{x}^T \mathbf{A} \mathbf{x}) = \frac{\partial}{\partial x_k} \sum_{i=1}^n \sum_{j=1}^n a_{ij} x_i x_j = \sum_{i=1}^n a_{ik} x_i + \sum_{j=1}^n a_{kj} x_j = 2 \sum_{i=1}^n a_{ik} x_i \quad (12.68)$$

Note that here we have used the assumption $\mathbf{A}^T = \mathbf{A}$. Putting all n elements in vector form, we get Eq.12.67. In particular, when $\mathbf{A} = \mathbf{I}$, we obtain Eq.12.66. In particular, when $n = 1$, we get this familiar derivative in the scalar case:

$$\frac{d}{dx}(ax^2) = \frac{d}{dx}(xax) = 2ax \quad (12.69)$$

Let $\mathbf{A} = [a_{ij}]_{m \times n}$ ($i = 1, \dots, m$, $j = 1, \dots, n$) be an m by n matrix. A matrix differentiation operator is defined as

$$\frac{d}{d\mathbf{A}} = \begin{bmatrix} \frac{\partial}{\partial a_{11}} & \cdots & \frac{\partial}{\partial a_{1n}} \\ \vdots & \ddots & \vdots \\ \frac{\partial}{\partial a_{m1}} & \cdots & \frac{\partial}{\partial a_{mn}} \end{bmatrix} \quad (12.70)$$

which can be applied to any scalar function $f(\mathbf{A})$ to find its derivative with respect to its matrix argument \mathbf{A} :

$$\frac{d}{d\mathbf{A}} f(\mathbf{A}) = \begin{bmatrix} \frac{\partial}{\partial a_{11}} f(\mathbf{A}) & \cdots & \frac{\partial}{\partial a_{1n}} f(\mathbf{A}) \\ \vdots & \ddots & \vdots \\ \frac{\partial}{\partial a_{m1}} f(\mathbf{A}) & \cdots & \frac{\partial}{\partial a_{mn}} f(\mathbf{A}) \end{bmatrix} \quad (12.71)$$

In particular when $f(\mathbf{A}) = \mathbf{u}^T \mathbf{A} \mathbf{v}$, where \mathbf{u} and \mathbf{v} are $m \times 1$ and $n \times 1$ constant vectors, respectively, we have:

$$\frac{\partial}{\partial a_{ij}} [\mathbf{u}^T \mathbf{A} \mathbf{v}] = \frac{\partial}{\partial a_{ij}} [\sum_{i=1}^m \sum_{j=1}^n u_i a_{ij} v_j] = u_i v_j, \quad (i = 1, \dots, m, j = 1, \dots, n) \quad (12.72)$$

i.e.,

$$\frac{d}{d\mathbf{A}} (\mathbf{u}^T \mathbf{A} \mathbf{v}) = \mathbf{u} \mathbf{v}^T \quad (12.73)$$

13 Appendix 2: Review of Random Variables

13.1 Random Variables

- **Random Experiment and its Sample Space**

A *random experiment* is a procedure that can be carried out repeatedly with a random outcome generated each time. The *sample space* Ω of the random experiment is a set containing all of its possible outcomes. Ω may be finite, countable infinite, or uncountable.

For example, “Randomly pick a card from a deck of cards labeled 0, 1, 2, 3 and 4” is a random experiment. The sample space is a set of all of the possible outcomes: $\Omega = \{0, 1, 2, 3, 4\}$.

- **Random Events**

An *event* $A \subset \Omega$ is a subset of the sample space Ω . A can be an empty set \emptyset , a proper subset (e.g., a single outcome), or the entire sample space Ω . Event A occurs if the outcome is a member of A .

The *event space* \mathcal{F} is set of events. If Ω is finite and countable, then $\mathcal{F} = Pow(\Omega)$ is the power set of Ω (a set of all possible subsets of Ω). But if Ω is infinite or uncountable, \mathcal{F} is a σ -algebra on Ω satisfying the following:

- $\Omega \in \mathcal{F}$ (or $\emptyset \in \mathcal{F}$).
- closed to countable unions: if $A_i \in \mathcal{F}$ ($i = 1, 2, \dots$), then $\cup_i A_i \in \mathcal{F}$;
- closed to complements: if $A \in \mathcal{F}$, then $\overline{\Omega} = \Omega - A \in \mathcal{F}$.

The ordered pair (Ω, \mathcal{F}) is called a *measurable space*. The concept of σ -algebra is needed to introduce a probability measure for all events in \mathcal{F} .

For example, $\mathcal{F} = \{\emptyset, \{0, 1, 2\}, \{2, 3\}, \Omega = \{0, 1, 2, 3, 4\}\}$

- **Probability**

The *probability* is a measure on \mathcal{F} . Probability of any event $A \in \mathcal{F}$ is a function $P(A)$ from A to a real value in the range $[0, 1]$, satisfying the following:

- $0 \leq P(A) \leq 1$ for all $A \in \mathcal{F}$.
- $P(\emptyset) = 0$, and $P(\Omega) = 1$.
- $P(A \cup B) = P(A) + P(B)$ if $A \cap B = \emptyset$ for all $A, B \in \mathcal{F}$.

For example, “The randomly chosen card has a number smaller than 3” is a random event, which is represented by a subset $A = \{0, 1, 2\} \subset \Omega$. The probability of this event A is $P(A) = 3/5$. Event A occurs if the outcome ω is one of the members of A , $\omega \in A$, e.g., 2.

- **Probability Space**

The triple (Ω, \mathcal{F}, P) is called the *probability space*.

- **Random Variables**

A random variable $x(\omega)$ is a complex-valued (or real-valued as a special case) function $x : \Omega \rightarrow \mathbb{R}$ that maps every outcome $\omega \in \Omega$ into a complex number x . Formally, the function $x(\omega)$ is a random variable if

$$\{\omega : x(\omega) \leq r\} \in \mathcal{F}, \quad \forall r \in \mathbb{R} \quad (13.1)$$

Random variables x can be either continuous or discrete.

- **Cumulative Distribution Function**

The *cumulative distribution function* of a random variable x is defined as

$$F_x(u) = P(x < u) \quad (13.2)$$

and we have $F_x(\infty) = 1$ and $F_x(-\infty) = 0$.

- **Density Function**

The *density function* of a random variable x is defined by

$$p_x(x) = \frac{d}{du} F_x(u), \quad \text{i.e.,} \quad F_x(u) = \int_{-\infty}^u p_x(x) dx \quad (13.3)$$

We have

$$P(a \leq x < b) = F_x(b) - F_x(a) = \int_a^b p_x(x) dx \quad (13.4)$$

In particular

$$P(x < \infty) = F_x(\infty) - F_x(-\infty) = \int_{-\infty}^{\infty} p_x(x) dx = 1 \quad (13.5)$$

The subscript of p_x can be dropped if no confusion will be caused.

- **Discrete Random Variables**

If a random variable x can only take one of a set of N values $\{x_n \mid n = 1, \dots, N\}$, then its *probability distribution* is

$$P(x = x_n) = p_n \quad (n = 1, \dots, N) \quad (13.6)$$

where

$$0 \leq p_n \leq 1, \quad \text{and} \quad \sum_{i=1}^N p_i = 1 \quad (13.7)$$

The cumulative distribution function is

$$F_x(\xi) = P(x < \xi) = \sum_{x_n < \xi} p_n \quad (13.8)$$

- **Expectation**

The *expectation* is the mathematical mean of a random variable x . If x is continuous,

$$\mu_x = E(x) = \int_{-\infty}^{\infty} x p(x) dx \quad (13.9)$$

If x is discrete,

$$\mu_x = E(x) = \sum_{n=1}^N x_n p_n \quad (13.10)$$

- **Variance**

The *variance* represents the statistical variability of a random variable x . If x is continuous,

$$\sigma_x^2 = Var(x) = E[|x - \mu_x|^2] = \int_{-\infty}^{\infty} |x - \mu_x|^2 p(x) dx \quad (13.11)$$

If x is discrete,

$$\sigma_x^2 = Var(x) = E[|x - \mu_x|^2] = \sum_{n=1}^N |x_n - \mu_x|^2 p_n \quad (13.12)$$

We also have

$$\begin{aligned} \sigma_x^2 &= Var(x) = E(|x - \mu_x|^2) = E[(x - \mu_x)(\overline{x - \mu_x})] \\ &= E(|x|^2) - \mu_x E(\overline{x}) - E(x)\overline{\mu} + |\mu_x|^2 = E(|x|^2) - |\mu_x|^2 \end{aligned} \quad (13.13)$$

The *standard deviation* of x is defined as

$$\sigma_x = \sqrt{Var(x)} \quad (13.14)$$

- **Normal (Gaussian) Distribution**

Random variable x has a *normal distribution* if its density function is

$$p(x) = N(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{\sigma^2}} \quad (13.15)$$

It can be shown that

$$\int_{-\infty}^{\infty} N(x, \mu, \sigma) dx = 1 \quad (13.16)$$

$$E(x) = \int_{-\infty}^{\infty} x N(x, \mu, \sigma) dx = \mu \quad (13.17)$$

and

$$Var(x) = \int_{-\infty}^{\infty} (x - \mu)^2 N(x, \mu, \sigma) dx = \sigma^2 \quad (13.18)$$

13.2 Multivariate Random Variables

- **Multivariate Random Variables**

A set of N *multivariate random variables* can be represented as a *random vector* $\mathbf{x} = [x_1, \dots, x_N]^T$ of component x_n ($n = 1, \dots, N$) is a random variable. When a *stochastic* or *random process* (to be discussed later) $x(t)$ is sampled, it can be represented as a random vector \mathbf{x} .

- **Joint Distribution Function and Density Function**

The *joint distribution function* of a random vector \mathbf{x} is defined as

$$\begin{aligned} F_{\mathbf{x}}(u_1, \dots, u_N) &= P_{\mathbf{x}}(x_1 < u_1, \dots, x_n < u_N) \\ &= \int_{-\infty}^{u_1} \cdots \int_{-\infty}^{u_N} p_{\mathbf{x}}(x_1, \dots, x_N) dx_1 \cdots dx_N = \int_{-\infty}^{\mathbf{u}} p(\mathbf{x}) d\mathbf{x} \end{aligned} \quad (13.19)$$

where $p(\mathbf{x}) = p_{\mathbf{x}}(x_1, \dots, x_N)$ is the *joint density function* of the random vector \mathbf{x} .

- **Mean Vector**

The *expectation* or *mean* of random variable x_n is defined as

$$\mu_n = E(x_n) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} x_n p_{\mathbf{x}}(x_1, \dots, x_N) dx_1 \cdots dx_N \quad (13.20)$$

The *mean vector* of random vector \mathbf{x} is defined as:

$$\boldsymbol{\mu}_x = E(\mathbf{x}) = \int_{-\infty}^{\infty} \mathbf{x} p(\mathbf{x}) d\mathbf{x} = [E(x_1), \dots, E(x_N)]^T = [\mu_1, \dots, \mu_N]^T \quad (13.21)$$

which can be interpreted as the center of gravity of an N-dimensional object with $p_{\mathbf{x}}(x_1, \dots, x_N)$ being the density function.

- **Covariance Matrix**

The *variance* of random variable x_n measures its variability and is defined as:

$$\begin{aligned} \sigma_n^2 &= Var(x_n) = E[|x_n - \mu_n|^2] = E(|x_n|^2) - |\mu_n|^2 \\ &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} |x_n - \mu_n|^2 p_{\mathbf{x}}(x_n, \dots, x_N) dx_1 \cdots dx_N \end{aligned} \quad (13.22)$$

The *covariance* of x_m and x_n ($m, n = 1, \dots, N$) measures their similarity and is defined as:

$$\begin{aligned} \sigma_{mn}^2 &= Cov(x_m, x_n) = E[(x_m - \mu_m)(\overline{x_n - \mu_n})] \\ &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} (x_m - \mu_m)(\overline{x_n - \mu_n}) p_{\mathbf{x}}(x_1, \dots, x_N) dx_1 \cdots dx_N \end{aligned} \quad (13.23)$$

Note that:

$$\begin{aligned} \sigma_{mn}^2 &= E[(x_m - \mu_m)(\overline{x_n - \mu_n})] = E(x_m \overline{x_n}) - E(x_m) \overline{\mu_n} - \mu_m E(\overline{x_n}) + \mu_m \overline{\mu_n} \\ &= E(x_m \overline{x_n}) - \mu_m \overline{\mu_n} \end{aligned} \quad (13.24)$$

The *covariance matrix* of a random vector \mathbf{x} is defined as

$$\begin{aligned} \boldsymbol{\Sigma}_x &= \int_{-\infty}^{\infty} (\mathbf{x} - \boldsymbol{\mu}_x)(\mathbf{x} - \boldsymbol{\mu}_x)^* p(\mathbf{x}) d\mathbf{x} \\ &= E[(\mathbf{x} - \boldsymbol{\mu}_x)(\mathbf{x} - \boldsymbol{\mu}_x)^*] = E(\mathbf{x}\mathbf{x}^*) - \boldsymbol{\mu}_x \boldsymbol{\mu}_x^* \\ &= \begin{bmatrix} \sigma_{11}^2 & \cdots & \sigma_{1N}^2 \\ \vdots & \ddots & \vdots \\ \sigma_{N1}^2 & \cdots & \sigma_{NN}^2 \end{bmatrix} \end{aligned} \quad (13.25)$$

When $m = n$, $\sigma_n^2 = E(|x_n|^2) - |\mu_n|^2$ is the variance of x_n , which can be interpreted as the amount of information, or energy, contained in the n th component x_n of the signal \mathbf{x} . Therefore the total information or energy contained in \mathbf{x} is:

$$\text{tr } \Sigma_x = \sum_{n=1}^N \sigma_n^2 \quad (13.26)$$

Obviously Σ is symmetric as $\sigma_{mn}^2 = \sigma_{nm}^2$. Moreover, it can be shown that Σ is also *positive definite*, and all its eigenvalues $\lambda_n > 0$ ($n = 1, \dots, N$) are positive and we have:

$$\text{tr } \Sigma_x = \sum_{n=1}^N \lambda_n > 0, \quad \det \Sigma_x = \prod_{n=1}^N \lambda_n > 0 \quad (13.27)$$

- **Correlation Coefficient**

The covariance σ_{mn}^2 of two random variables x_m and x_n represents the statistical similarity between them. If $\sigma_{mn}^2 > 0$, x_m and x_n are positively correlated; if $\sigma_{mn}^2 < 0$, they are negatively correlated, if $\sigma_{mn}^2 = 0$, they are *uncorrelated* or *decorrelated*. The normalized covariance is called the *correlation coefficient*:

$$r_{mn} = \frac{\sigma_{mn}^2}{\sigma_m \sigma_n} = \frac{E(x_m \bar{x}_n) - \mu_m \bar{\mu}_n}{\sqrt{E(|x_m|^2) - |\mu_m|^2} \sqrt{E(|x_n|^2) - |\mu_n|^2}} \quad (13.28)$$

The correlation coefficient $-1 \leq r_{mn} \leq 1$ measures the similarity between the two random variables x_m and x_n . They are either positively correlated if $r_{mn} > 0$, negatively correlated if $r_{mn} < 0$, or uncorrelated if $r_{mn} = 0$.

The correlation matrix of a random vector is therefore defined as:

$$\mathbf{R} = \begin{bmatrix} r_{11} & \cdots & r_{1N} \\ \vdots & \ddots & \vdots \\ r_{N1} & \vdots & r_{NN} \end{bmatrix} \quad (13.29)$$

Obviously all elements $r_{nn} = 1$ ($n = 1, \dots, N$) along the main diagonal of \mathbf{R} are 1, and all off-diagonal elements $|r_{mn}| < 1$ ($m \neq n$).

- **Correlation and Independence**

A set of N random variables x_n ($n = 1, \dots, N$) are independent if and only if

$$p(\mathbf{x}) = p_{\mathbf{x}}(x_1, \dots, x_N) = p(x_1) p(x_2) \cdots p(x_N) \quad (13.30)$$

Two random variables x_m and x_n are uncorrelated if $r_{mn} = 0$, i.e.,

$$\sigma_{mn}^2 = E(x_m \bar{x}_n) - \mu_m \bar{\mu}_n = 0, \quad \text{or} \quad E(x_m \bar{x}_n) - \mu_m \bar{\mu}_n = 0 \quad (13.31)$$

Obviously if x_m and x_n are independent, we have $E(x_m \bar{x}_n) = E(x_m)E(\bar{x}_n) = \mu_m \bar{\mu}_n$ they are uncorrelated. However, if they are uncorrelated, they are not necessarily independent, unless they are normally distributed.

A random vector $\mathbf{x} = [x_1, \dots, x_N]^T$ is uncorrelated or decorrelated if $r_{mn} = 0$ for all $m \neq n$, and both its covariance Σ and correlation \mathbf{R} become diagonal matrices with only non-zero σ_n^2 ($n = 1, \dots, N$) on its diagonal.

- **Mean and Covariance under Unitary Transforms**

If the inverse of a matrix is the same as its conjugate transpose: $\mathbf{A}^{-1} = \mathbf{A}^*$, then it is a unitary matrix. Given any unitary matrix \mathbf{A} , an orthogonal transform of a random vector \mathbf{x} can be defined as

$$\begin{cases} \mathbf{X} = \mathbf{A}^* \mathbf{x} \\ \mathbf{x} = \mathbf{A} \mathbf{X} \end{cases} \quad (13.32)$$

The mean vector $\boldsymbol{\mu}_X$ and the covariance matrix Σ_X of \mathbf{X} are related to the $\boldsymbol{\mu}_x$ and Σ_x of \mathbf{x} by:

$$\boldsymbol{\mu}_X = E(\mathbf{X}) = E(\mathbf{A}^* \mathbf{x}) = \mathbf{A}^* E(\mathbf{x}) = \mathbf{A}^* \boldsymbol{\mu}_x \quad (13.33)$$

$$\begin{aligned} \Sigma_X &= E(\mathbf{X} \mathbf{X}^*) - \boldsymbol{\mu}_X \boldsymbol{\mu}_X^* = E(\mathbf{A}^* \mathbf{x} \mathbf{x}^* \mathbf{A}) - \mathbf{A}^* \boldsymbol{\mu}_x \boldsymbol{\mu}_x^* \mathbf{A} \\ &= \mathbf{A}^* E(\mathbf{x} \mathbf{x}^*) \mathbf{A} - \mathbf{A}^* \boldsymbol{\mu}_x \boldsymbol{\mu}_x^* \mathbf{A} = \mathbf{A}^* [E(\mathbf{x} \mathbf{x}^*) - \boldsymbol{\mu}_x \boldsymbol{\mu}_x^*] \mathbf{A} \\ &= \mathbf{A}^* \Sigma_x \mathbf{A} \end{aligned} \quad (13.34)$$

Unitary transform does not change the trace of Σ :

$$\text{tr } \Sigma_X = \text{tr } \Sigma_x \quad (13.35)$$

which means the total amount of energy or information contained in \mathbf{x} is not changed after a unitary transform $\mathbf{X} = \mathbf{A}^* \mathbf{x}$ (although the distribution of energy among the components may be changed).

- **Normal Distribution**

The density function of a normally distributed random vector \mathbf{x} is:

$$p(\mathbf{x}) = N(\mathbf{x}, \boldsymbol{\mu}_x, \Sigma_x) = \frac{1}{(2\pi)^{n/2} |\Sigma_x|^{1/2}} \exp\left[-\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu}_x)^T \Sigma_x^{-1} (\mathbf{x} - \boldsymbol{\mu}_x)\right] \quad (13.36)$$

When $n = 1$, Σ_x and $\boldsymbol{\mu}_x$ become σ_x and μ_x , respectively, and the density function becomes single variable normal distribution.

To find the shape of a normal distribution, consider the iso-value hyper surface in the N-dimensional space determined by equation

$$N(\mathbf{x}, \boldsymbol{\mu}_x, \Sigma_x) = c_0 \quad (13.37)$$

where c_0 is a constant. This equation can be written as

$$(\mathbf{x} - \boldsymbol{\mu}_x)^T \Sigma_x^{-1} (\mathbf{x} - \boldsymbol{\mu}_x) = c_1 \quad (13.38)$$

where c_1 is another constant related to c_0 , $\boldsymbol{\mu}_x$ and Σ_x . This equation represents a hyper ellipsoid in the N-dimensional space. The center and spatial distribution of this ellipsoid are determined by $\boldsymbol{\mu}_x$ and Σ_x , respectively.

In particular, when $\mathbf{x} = [x_1, \dots, x_N]^T$ is decorrelated, i.e., $\sigma_{mn}^2 = 0$ for all $m \neq n$, Σ_x becomes a diagonal matrix

$$\Sigma_x = \text{diag}[\sigma_1^2, \dots, \sigma_N^2] = \begin{bmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_N^2 \end{bmatrix} \quad (13.39)$$

and equation $N(\mathbf{x}, \boldsymbol{\mu}_x, \Sigma_x) = c_0$ can be written as

$$(\mathbf{x} - \boldsymbol{\mu}_x)^T \Sigma_x^{-1} (\mathbf{x} - \boldsymbol{\mu}_x) = \sum_{n=1}^N \frac{(x_n - \mu_n)^2}{\sigma_n^2} = c_1 \quad (13.40)$$

which represents a standard ellipsoid with all its axes are in parallel with those of the coordinate system.

- **Estimation of $\boldsymbol{\mu}_x$ and Σ_x**

When $p(\mathbf{x}) = p(x_1, \dots, x_n)$ is not known, $\boldsymbol{\mu}_x$ and Σ_x cannot be found by their definitions, but they can be estimated if a set of K outcomes $(\mathbf{x}^{(k)}, k = 1, \dots, K)$ of the random experiment can be observed. Then the mean vector can be estimated as

$$\hat{\boldsymbol{\mu}}_x = \frac{1}{K} \sum_{k=1}^K \mathbf{x}^{(k)} \quad (13.41)$$

i.e., the nth element of $\hat{\boldsymbol{\mu}}_x$ is estimated as

$$\hat{\mu}_n = \frac{1}{K} \sum_{k=1}^K x_n^{(k)}, \quad (n = 1, \dots, N) \quad (13.42)$$

where $x_n^{(k)}$ is the nth element of the kth outcome $\mathbf{x}^{(k)}$. The covariance matrix Σ_x can be estimated as

$$\hat{\Sigma}_x = \frac{1}{K-1} \sum_{k=1}^K (\mathbf{x}^{(k)} - \hat{\boldsymbol{\mu}}_x)(\mathbf{x}^{(k)} - \hat{\boldsymbol{\mu}}_x)^T = \frac{1}{K-1} \sum_{k=1}^K \mathbf{x}^{(k)} \mathbf{x}^{(k)T} - \hat{\boldsymbol{\mu}}_x \hat{\boldsymbol{\mu}}_x^T \quad (13.43)$$

i.e., the mn-th element of $\hat{\Sigma}_x$ is

$$\hat{\sigma}_{mn} = \frac{1}{K-1} \sum_{k=1}^K (x_m^{(k)} - \hat{\mu}_m)(x_n^{(k)} - \hat{\mu}_n) = \frac{1}{K-1} \sum_{k=1}^K x_m^{(k)} x_n^{(k)} - \hat{\mu}_m \hat{\mu}_n \quad (13.44)$$

Note that for the estimation of the covariance to be unbiased, i.e., $E(\hat{\Sigma}_x) = \Sigma_x$, the coefficient $1/(K-1)$, instead of $1/K$, needs to be used. Obviously this makes little difference when the number of samples K is large.

13.3 Stochastic Models

A physical signal can be modeled as a time function $x(t)$ which takes a complex value (or real value as a special case) $x(t_0)$ at each time moment $t = t_0$. This value may be either deterministic or random with a certain probability distribution. In the latter case the time function is called a *stochastic process* or *random process*.

Recall that a random variable $x(\omega)$ is a function that maps the outcomes $\omega \in \Omega$ in the sample space Ω of a random experiment to a real number between 0 and 1. Here a stochastic process can be considered as a function $x(\omega, t)$ of two arguments of time t as well as the outcome $\omega \in \Omega$.

If the mean and covariance functions of a random process $x(t)$ do not change over time, i.e.,

$$\mu_x(t) = \mu_x(t - \tau), \quad R_x(t, \tau) = R_x(t - \tau), \quad \Sigma_x(t, \tau) = \Sigma_x(t - \tau) \quad (13.45)$$

then $x(t)$ is a *stationary process*, in the weak or wide sense (*weak-sense* or *wide-sense stationarity (WSS)*). If the probability distribution of $x(t)$ does not change over time, it is said to have *strict* or *strong stationarity*. We will only consider stationary processes.

- The *mean function* of $x(t)$ is the expectation defined as:

$$\mu_x(t) = E[x(t)] \quad (13.46)$$

If $\mu_x(t) = 0$ for all t , then $x(t)$ is a zero-mean or centered stochastic process, which can be easily obtained by subtracting the mean function $\mu_x(t)$ from the original process $x(t)$. If the stochastic process is stationary, then $\mu_x(t) = \mu_x$ is a constant.

- The *auto-covariance function* of $x(t)$ is defined as

$$\begin{aligned} \sigma_x^2(t, \tau) &= Cov[x(t), x(\tau)] = E[(x(t) - \mu_x)(x(\tau) - \mu_x)] \\ &= E[x(t)x(\tau)] - \mu_x(t)\mu_x(\tau) \end{aligned} \quad (13.47)$$

If the stochastic process is stationary, then $\sigma_x^2(t) = \sigma_x^2(\tau) = \sigma_x^2$, $\mu_x(t) = \mu_x(\tau) = \mu_x$, and $\sigma_x^2(t, \tau) = \sigma_x^2(t - \tau)$, the above can be expressed as

$$\sigma_x^2(t - \tau) = E[(x(t) - \mu_x)(x(\tau) - \mu_x)] = E[x(t)x(\tau)] - \mu_x^2 \quad (13.48)$$

- The *autocorrelation function* of $x(t)$ is defined as

$$r_x(t, \tau) = \frac{\sigma_x^2(t, \tau)}{\sigma_x(t)\sigma_x(\tau)} \quad (13.49)$$

If the stochastic process is stationary, then $\sigma_x^2(t) = \sigma_x^2(\tau) = \sigma_x^2$, and $\sigma_x^2(t, \tau) = \sigma_x^2(t - \tau)$, the above can be expressed as

$$r_x(t - \tau) = \frac{\sigma_x^2(t - \tau)}{\sigma_x^2} \quad (13.50)$$

- When two stochastic processes $x(t)$ and $y(t)$ are of interest, then their *cross-covariance* and *cross-correlation functions* are defined respectively as:

$$\begin{aligned}\sigma_{xy}^2(t, \tau) &= Cov[x(t), y(\tau)] = E[(x(t) - \mu_x(t))(y(\tau) - \mu_y(\tau))] \\ &= E[x(t)y(\tau)] - \mu_x(t)\mu_y(\tau)\end{aligned}\quad (13.51)$$

and

$$r_{xy}(t, \tau) = \frac{\sigma_{xy}^2(t, \tau)}{\sigma_x(t)\sigma_y(\tau)} \quad (13.52)$$

When only one stochastic process $x(t)$ is concerned, $\mu_x(t)$ and σ_x^2 can be simply referred to as its mean and covariance. If a stochastic process $x(t)$ has a zero mean, i.e., $\mu_x(t) = 0$ for all t , then it is said to be centered. Any stochastic process can be centered by a simple subtraction:

$$x'(t) = x(t) - \mu_x(t) \quad (13.53)$$

so that $\mu_{x'} = 0$. Without loss of generality, any stochastic process can be assumed to be centered. In this case, its covariance becomes

$$\sigma_x^2 = E[x^2(t)] \quad (13.54)$$

A *Markov process* $x(t)$ is a particular type of stochastic process whose future values depend only on its present value, but independent of any past values. In other words, the probability of any future value conditioned on present and all past values is equal to the probability conditioned only on the present value:

$$Pr(x(t+h) = y | x(s) = \xi(s), \forall s \leq t) = Pr[x(t+h) = y | x(t) = \xi(t)], \quad \forall h > 0 \quad (13.55)$$

When a stochastic process is sampled it becomes a time sequence of random variables $x[n]$ ($n = 1, \dots, N$), which can be represented by a random vector $\mathbf{x} = [x[0], \dots, x[N-1]]^T$. A *Markov chain* is defined as:

$$\begin{aligned}Pr(x[n] = y | x[m] = \xi[m], \forall m < n) \\ = Pr(x[n] = y | x[n-m] = \xi[n-m], m = 1, \dots, k)\end{aligned}\quad (13.56)$$

i.e., the value $x[n]$ depends only on the k prior values. In particular, when $k = 1$, this is a first order Markov chain:

$$Pr(x[n] = y | x[m] = \xi[m], \forall m < n) = Pr(x[n] = y | x[n-1] = \xi[n-1]) \quad (13.57)$$

Let $-1 < r < 1$ be the correlation coefficient between any two consecutive values $x[n]$ and $x[n-1]$ of a stationary first order Markov chain of size N , then the correlation matrix is:

$$\mathbf{R}_x = \begin{bmatrix} 1 & r & r^2 & \dots & r^{N-1} \\ r & 1 & r & \dots & r^{N-2} \\ r^2 & r & 1 & \dots & r^{N-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r^{N-1} & r^{N-2} & r^{N-3} & \dots & 1 \end{bmatrix}_{N \times N} \quad (13.58)$$

We see that the correlation between two variables $x[m]$ and $x[n]$ is $\rho^{|m-n|}$, which decays exponentially as a function of the distance $|m - n|$ between the two variables. This matrix \mathbf{R} is a Toeplitz matrix.

Moreover, when $k = 0$, we get a memoryless zero order Markov chain of which any value $x[n]$ is a random variable independent of any other value $x[m]$, in other words, all elements of the chain are totally decorrelated, i.e., $r_{mn} = \delta[m - n]$, and the correlation matrix is the identity matrix $\mathbf{R} = \mathbf{I} = \text{diag}(1, \dots, 1)$. Also, let σ^2 be the variance of any $x[n]$ of a stationary zero order Markov chain, then the covariance matrix is:

$$\Sigma_x = \begin{bmatrix} \sigma^2 & 0 & 0 & \cdots & 0 \\ 0 & \sigma^2 & 0 & \cdots & 0 \\ 0 & 0 & \sigma^2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \sigma^2 \end{bmatrix} = \sigma^2 \mathbf{I} \quad (13.59)$$

14 Bibliography

1. Roman, S. *Advanced Linear Algebra (Graduate Texts in Mathematics)*, 3rd ed., Vol. 135, Springer, 2008
2. Hirsch, F. and Lacombe, G. *Elements of Functional Analysis (Graduate Texts in Mathematics)*, Vol. 192, Springer, 1999
3. Christensen, O. *An Introduction to Frames and Riesz Bases*, Birkhäuser, 2003
4. Love, M. *Probability theory*, 4th ed., Graduate Texts in Mathematics, Vol. 46, Springer-Verlag, 1978.
5. Oppenheim, A.V. and Willsky A.S., *Signals and Systems*, 2nd ed., Prentice Hall, 1997
6. Poularikas, A.D. and Seely, S. *Signals and Systems*, PWS-KENT Publishing Company, 1991
7. Oppenheim, A.V. and Schafer, R.W. *Digital Signal Processing*,
8. Britanak, V., Yip, P.C., and Rao, K.R. *Discrete Cosine and Sine Transforms*, Academic Press, 2007
9. Rao, K.R. (ed.), *Discrete transforms and their applications*, Van Nostrand Reinhold, 1985
10. Rao, K.R. and Yip, P.C. *The Transform and Data Compression Handbook*, CRC Press LLC, 2001
11. Ahmed, N., and Rao, K.R. *Orthogonal Transforms for Digital Signal Processing*, Spring-Verlag, 1975
12. Jain, A. K. *Fundamentals of Digital Image Processing*, Prentice Hall, 1989
13. Bracewell, R. N. *The Fourier Transform and Its Applications*, McGraw Hill, 2000
14. Brigham, E.O. *The Fast Fourier Transform and Its Applications*, Prentice Hall, 1988
15. Rao, K.R., Kim, D.N., and Hwang, J.J. *Fast Fourier Transform – Algorithms and Applications (Signals and Communication Technology)*, Springer, 2010
16. Jolliffe, I.T. *Principal Component Analysis* 2nd ed. Springer, 2002
17. Strang, G., and Nguyen, T. *Wavelets and Filter Banks*, Wellesley-Cambridge Press, 1996
18. Mallat, S. *A Wavelet Tour of Signal Processing*, Academic Press, 1998
19. Vetterli, M. and Kovacevic, J. *Wavelets and Subband Coding*, Prentice Hall, 1995

Index

- adjoint transformation, 56
- admissibility condition, 450
- amplitude modulation (AM), 231
- analysis filter bank, 507
- analytic signal, 244
- autocorrelation, 10
- band-limited signal, 154
- basis of vector space, 41
- bilateral Laplace transform, 263
- biorthogonal bases, 86
- biorthogonal MRA, 492
- biorthogonal transformation, 86
- Bode plot, 224
 - first order systems, 279
 - second order systems, 288
- circular convolution, 179
- compact support, 481, 493
- conformal mapping, 297
- continuous convolution, 17
- continuous-time wavelet transform (CTWT), 447
- convolution
 - continuous, 3
 - discrete, 2
- convolution theorem
 - CTFT, 124
 - DFT, 177
 - DTFT, 146
- cross correlation, 10
- cross power spectral density, 146
- daughter wavelet, 448
- Delta function
 - continuous, 3
 - discrete, 2
- digital filter, 310
- Dirac delta, *see* Delta function, continuous
- Discrete cosine transform, 338–361
- discrete Fourier transform (DFT), 166
- Discrete sine transform, 338–361
- discrete wavelet transform (DWT), 503–520
- dual vector, 80
- dyadic wavelet transform, 458
- eigenface, 427
- eigenvalue problem, 57
 - eigen functions, 57
 - eigen values, 57
 - eigen vectors, 57
- energy signal, 9
- entropy, 400, 406
- Euclidean space, 35
- fast Fourier transform (FFT), 185–188
- father wavelet, 478
- finite impulse response (FIR) filters, 310
- Fourier spectrum, 115
- Fourier transform
 - continuous-time Fourier transform (CTFT), 114–136
 - discrete Fourier transform (DFT), 166–188
 - discrete-time Fourier transform (DTFT), 140–165
 - Fourier series expansion, 102–114
- frame, 80
 - dual frames, 81
 - frame transformation, 80
- frequency response function (FRF)
 - continuous, 18, 212
 - discrete, 21, 214
- Gabor transform, 445
- generalized Fourier expansion, 47
- Gram-Schmidt orthogonalization, 50
- Gray code, 369
- group delay, 224
- Haar transform, 382–391
- Hartley transform, 324–338
- Heaviside step function, *see* unit step function, continuous
- Heisenberg Box (or Heisenberg cell), 446
- Heisenberg uncertainty, *see* uncertainty principle

- Hermitian transformation, *see* self-adjoint transformation
 Hilbert space, 45
 Hilbert transform, 242–246
 Huffman coding, 407
- impulse response function
 continuous, 17
 discrete, 20
 infinite impulse response (IIR) filters, 311
 inner product, 34
 inner product space, 35
 integral transform, 97
- Karhunen-Loeve theorem
 continuous, 94, 402
 discrete, 95, 403
 Karhunen-Loeve transform, 402–411
 kernel function, 90
 kernel operator, 91
 Kronecker delta, *see* Delta function, discrete
 Kronecker product, 363
- Laplace transform, 262–295
 linear constant coefficient differential equation (LCCDE), 57
 linear operator, 55
 linear span, 41
 Linear time-invariant (LTI) system, 15
 linear transformation, 55
 local correlation, 400
- Markov chain, 400, 550
 Markov process, 550
 Mercer’s theorem, 92
 metric space, 40
 mother wavelet, 448, 483
 moving average filter
 continuous, 226
 discrete, 311
 multiresolution analysis (MRA), 477
 multiresolution analysis (MRA), 476–481
- nascent delta function, 5
 Nyquist frequency, 155
- Orthogonal Frequency Division Multiplexing (OFDM), 256
 orthogonal projection, 38
- Parseval’s identity
 CTFT, 136
 DFT, 173
 DTFT, 145
 Fourier series, 104
 Parseval’s theorem, *see* Parseval’s identity
- Fourier, 107
 phase delay, 224
 Plancherel theorem, 46
 power density spectrum (PDS), 121
 power signal, 10
 principal component analysis (PCA), 410
 probability density function (pdf), 397
 projection theorem, 69
 pseudo-inverse
 matrix, 71
 transformation, 81, 86
- quality factor Q , 452
- Radon transform, 247–255
 region of convergence (ROC)
 s-plane, 263
 z-plane, 296
 regular functions, 499
 remote sensing, 424, 430
 reproducing kernel, 454
 Riesz basis, 86, 477
- sampling, 140
 sampling theorem, 155
 self-adjoint transformation, 56
 sequency, 364
 short-time Fourier transform (STFT), 444
 singular value decomposition (SVD), 433–436
 singular value decomposition theorem, 434
 slant transform, 375–382
 spectrum centralization, 182
 stochastic process, 10, 397
 synthesis filter bank, 509
- transfer function
 continuous, 18, 271
 discrete, 21, 307
 two-channel filter bank, 509
 two-dimensional transforms
 DCT, 356–361
 Fourier, 192–208
 Hartley, 334–338
 Walsh-Hadamard, slant, Haar, 392–395
- uncertainty principle, 133–136, 445–447
 uncertainty theorem, 135
 unit step function
 continuous, 5
 discrete, 4
 unitary space, 35
 unitary transformation, 63
- vanishing moments, 499
 vector space, 33

Walsh-Hadamard transform (WHT),
363–375
wavelet transform
continuous-time (CTWT), 447–457
discrete-time (DTWT), 457–474
wavelets
Derivative of Gaussian, 456
Difference of Gaussians, 457
Marr wavelet (Mexican hat), 456
Morlet, 455
Shannon, 455
windowed Fourier transform, 444

z-transform, 295–319