Introduction to Orthogonal Transforms

with Applications in Data Processing and Analysis

Introduction to Orthogonal Transforms

with Applications in Data Processing and Analysis

Ruye Wang
June 2, 2010

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

First published 2007

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

Contents

Preface

What Is the Book 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", from which a more general concept of "orthogonal" is derived. Although in this book we will be mostly concerned with orthogonal vectors or functions, they are essentially no different from two perpendicular straight lines, as described 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 in particular. This book is about a set of computational methods, known collectively as orthogonal transforms, that enables us to take advantage of orthogonality. As we will see through out the book, orthogonality is a much desired property that will keep things untangled and nicely separated for ease of manipulation, and an orthogonal transform can rotate a signal, represented as a vector in Euclidean space, or more generally, in Hilbert space, in such a way that the signal components tend to become, approximately or accurately, orthogonal to each other. These orthogonal transforms, such as the Fourier transform and discrete cosine 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, and the wavelet transforms that may not be strictly orthogonal but still closely related to those that are.

In the last few decades the scale of data collection across many fields has 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. 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 engineering, there is always the need to process and compress the data in some meaningful way for the purpose of effective and efficient data analysis and interpretation, by various

computational methods and algorithms. The transform methods covered 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, or extract a particular kind of features of interest, and very often it is desirable to 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 processing. To address all such needs, this book presents a thorough introduction to the mathematical background common to a set of transform methods used in a wide variety of data processing problems, 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 transform method discussed. Certainly many existing books touch upon the topics of orthogonal transform and wavelet transforms, from either mathematical or engineering point of view. Some of them may concentrate on the theories of these transform 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 to be able to convert the mathematical theory ultimately into computer code so that the algorithm can be actually implemented and tested. This idea has been the guiding principle through out the writing of the book. For each of the orthogonal and wavelet transform method covered, we will first derive the theory mathematically, then present the corresponding computational algorithm, and finally provide the 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 orthogonal and wavelet transform methods covered 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 accompanying CD as well as a website dedicated to the book at: http://fourier.eng.hmc.edu/book/programs. 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 method now called wavelet transform. While all of these ideas were originally

developed with different goals and applications in mind, for either solving the heat equation or 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 on the data to start with? As the measurement of a certain variable such as the temperature or pressure of some physical process, a signal tends to vary continuously and smoothly, as the energy associated with the physical process tends to be distributed relatively evenly in both space and time. Most of such temporal or spatial 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 ruled by the principles of minimum energy and maximum entropy, in which any abruption and discontinuities, typically caused by energy surge of some kind, are relatively rare and unlikely events (except in the macroscopic world ruled by the 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 becomes difficult to decompose the signal, which is needed in various applications such as information extraction, noise reduction and data compression. The issue therefore becomes, how can the signal be converted so that it is less correlated and its energy is less evenly distributed, and to what extent can such a conversion be carried out to achieve such goals.

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 a some vector space which can be represented under different orthogonal bases that all span the same space. Each of such representations corresponds to a different decomposition of the signal. Moreover, these representations are all equivalent in the sense that they are related to each other by certain rotation in the space which conserves the total energy or information contained in the signal. 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 algorithms for the same general purpose.

While all representations of a given signal corresponding to different transform methods are equivalent in terms of the total signal energy which is always conserved, they may be different in terms of how much the signal components after the transform are still correlated, and how the total energy or informa-

tion in the signal is redistributed among the components. If, after a properly chosen orthogonal transform, the signal will be 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 first two chapters establish the mathematical foundation for the subsequent chapters each discussing a specific type of transform methods. Chapter 1 is a brief summary of the basic concepts of signals and linear time-invariant (LTI) systems. For readers with engineering background, most of this chapter may be a quick review which could even be 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 vector space, or more strictly speaking, Hilbert space, and the linear transformations in such a space. Here a usual N-dimensional space is further 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 the 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 the later chapters will be viewed and studied in such a frame work.

In Chapters 3 and 4, we study the classical Fourier methods for continuous and discrete signals respectively. 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 ways to categorize a signal: 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 briefly 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 into another 1-dimensional function along a different variable, typically, frequency $\omega = 2\pi f$. Instead, the Laplace and Z-transforms convert a 1-dimensional signal from time domain into a 2-dimensional function in a complex plane $s = \sigma + j\omega$, called s-plane for a continuous signal, or $z = e^s$ called z-plane for a discrete signal. 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 the toolbox.

Chapter 6 discusses the Hartly and cosine transforms, both of which are closely related to the Fourier transform. As real transforms, both Hartly and cosine transforms have the advantage of reduced computational cost when compared with the Fourier transform, which by definition is complex. If the signal in question is real, i.e., its imaginary part is all zero, then half of the computation in its Fourier transform is redundant and therefore wasted. However, this redundancy is avoided by real transforms such as the cosine transform, which, for this reason, 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, due to the fact that the Haar transform possesses the basic characteristics of the general wavelet transform method, and also due to its simplicity, the Haar transform can also serve as a bridge between the two camps of the orthogonal transforms and the wavelet transforms, and a natural transition leading 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 methods, and the associated data analysis method, 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-dimensional time signal into a 2-dimensional function of two variables, one for different levels of details corresponding to different frequencies in the Fourier transform, while the other for different temporal positions, which is completely absent in the Fourier or any other orthogonal transform. While redundancy is inevitably introduced into the 2-dimensional 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. Such a capability is the main advantage of the wavelet transform method over orthogonal transforms like the Fourier transform in some applications such as data compression.

Finally in Chapter 10, we introduce the basic concept of multiresolution analysis (MA), 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.

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 systematically 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 book 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 filed. 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 project 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 years but till feel the need to update and expand his/her knowledge. Such readers could certainly study whichever chapters of interest, instead of systematically reading through out the chapters from beginning to end. It is hoped that the book can serve as a toolbox from which some pertinent transform methods can be chosen and applied, in combination with the reader's expertise in his/her own field, to develop a solution to the specific data processing/analysis problems at hand.

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." Explicit efforts are made in this book to discuss the practical uses of the orthogonal transforms as well as the mathematics behind them, 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 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 what he learns from the book, much more than three pence.

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 $u + jv$
$\angle(u + jv) = \tan^{-1}(v/u)$	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
$x(t)$	continuous time signal
$x[n]$	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 $x(t)$

Unless otherwise noted, a bold-faced lower case letter $\boldsymbol{x}$ represents a vector, and a bold-faced upper case letter $\boldsymbol{A}$ represents a matrix.

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 of the concepts, instead of the mathematical rigor.

1.1 Continuous and Discrete Signals

A physical signal, always assumed to be a real or complex-valued function of time, unless otherwise specified (e.g., a spatial function), can be recorded as a continuous time function $x(t)$, or sampled at a certain rate (number of samples per unit time) to produce a discrete time function $x[n]$. In either case, the duration of the signal is finite in practice but could also be considered infinite in theory, i.e., $-\infty < t < \infty$ for $x(t)$ and $-\infty < n < \infty$ for $x[n]$.

A given continuous signal $x(t)$ can be discretized to generate a set of discrete samples $x[n]$. Assume the time interval between two consecutive samples, is $\triangle$ seconds, then the nth sample is:

$$x[n] = x(t)\big|_{t=n\triangle} = x(n\triangle) \tag{1.1}$$

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

We define the *discrete unit impulse or Kronecker delta function* as:

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

and represent a discrete signal as:

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

This equation can be interpreted in two conceptually different ways.

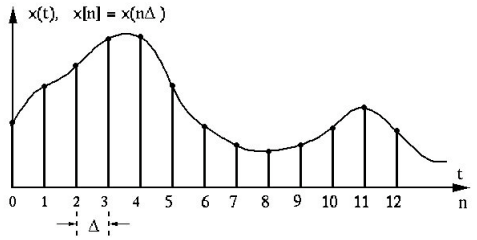

Figure 1.1 Sampling of a continuous signal

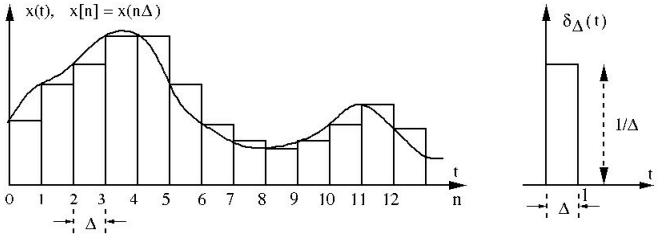

Figure 1.2 Approximation of a continuous signal

- 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 $n = m$, as shown in Fig.1.1.
- Second, the Kronecker delta $\delta[n - m]$ sifts out one particular value of the signal $x[n]$ at $m = n$ from a sequence of signal samples. This is the *sifting property* of the function.

On the other hand, a continuous signal can also be approximated by a set of its samples. To do so, we first define a unit square impulse function:

$$\delta_{\triangle}(t) = \begin{cases} 1/\Delta & 0 \le t < \triangle \\ 0 & \text{otherwise} \end{cases} \tag{1.4}$$

As the width and height of this square impulse are respectively $\triangle$ and $1/\triangle$, i.e, it covers a unit area $\triangle \times 1/\triangle = 1$, independent of $\triangle$. Now the continuous signal $x(t)$ can be approximated by its samples $x[n]$ as a sequence of weighted square impulses:

$$x(t) \approx \hat{x}(t) = \sum_{n=-\infty}^{\infty} x\,[n]\,\delta_{\triangle}(t - n\triangle)\triangle \tag{1.5}$$

This approximation is composed of a sequence of square impulses $x[n]\delta_{\triangle}(t - n\triangle)$, which is weighted by the sample value $x[n]$ for the amplitude of the signal at the moment $t = n\triangle$, as shown in Fig.1.2. If we let $\triangle \to 0$, the square impulse function will have infinitesimally narrow width and infinite height. At the limit, the summation in Eq.1.5 becomes an integral and the approximation becomes

exact:

$$x(t) = \lim_{\triangle \to 0} \sum_{n=-\infty}^{\infty} x[n]\delta_{\triangle}(t - n\triangle)\triangle = \int_{-\infty}^{\infty} x(\tau)\delta(t-\tau)d\tau \tag{1.6}$$

where

$$\delta(t) = \lim_{\triangle \to 0} \delta_{\triangle}(t) = \begin{cases} \infty & t = 0 \\ 0 & t \neq 0 \end{cases} \tag{1.7}$$

is the *continuous unit impulse or Dirac delta function*. The Dirac delta $\delta(t)$ has an infinite height but zero width at $t = 0$, and it still covers a unit area:

$$\int_{-\infty}^{\infty} \delta(t)dt = \lim_{\Delta \to 0} [\triangle \cdot 1/\triangle] = 1 \tag{1.8}$$

In particular, when $t = 0$, Eq.1.6 becomes:

$$x(0) = \int_{-\infty}^{\infty} x(\tau)\delta(\tau)d\tau \tag{1.9}$$

Eq.1.6 can be interpreted in two conceptually different ways.

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

Note that the discrete impulse function $\delta[n]$ has unit height, while the continuous impulse function $\delta(t)$ has a unit area, the height multiplied by width (time), and they have different dimensions. The dimension of the discrete impulse function is the same as that of the signal (e.g., voltage), while the dimension of the latter 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 integrated over time will its dimension become the same as the signal $x(t)$.

In summary, 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 signal is not unique. The signal can also be decomposed in different domains other than time, such as frequency, and these representations of the signal in different domains are related by certain orthogonal transformations.

1.2 The Dirac Delta and Unit Step Function

The impulse function $\delta(t)$ is closely related to the unit step function (also called Heaviside step function) $u(t)$, another important functions to be heavily used in

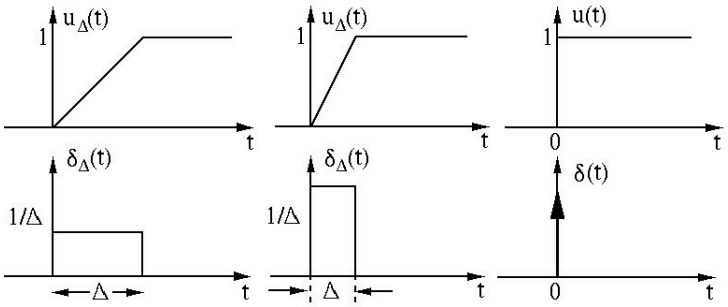

Figure 1.3 Generation of unit step and unit impulse

future discussions. First, we define a piece-wise linear function as:

$$u_{\triangle}(t) = \begin{cases} 0 & t < 0 \\ t/\triangle & 0 \le t < \triangle \\ 1 & t \ge \triangle \end{cases} \tag{1.10}$$

Taking the time derivative of this function, we get

$$\delta_{\triangle}(t) = \frac{d}{dt} u_{\triangle}(t) = \begin{cases} 0 & t < 0 \\ 1/\triangle & 0 \le t < \triangle \\ 0 & t \ge \triangle \end{cases} \tag{1.11}$$

which is the square impulse considered before in Eq.1.4. As its width and height are respectively $\triangle$ and $1/\triangle$, the area underneath the function is 1:

$$\int_{-\infty}^{\infty} \delta_{\triangle}(t)dt = \frac{1}{\triangle}\triangle = 1 \tag{1.12}$$

At the limit $\triangle \to 0$, $u_{\triangle}(t)$ becomes the *unit step function*:

$$\lim_{\triangle \to 0} u_{\triangle}(t) = u(t) = \begin{cases} 1 & t > 0 \\ 0 & t < 0 \end{cases} \tag{1.13}$$

Note that the value $u(0)$ of the step function at $t = 0$ is not specifically defined in this process. Although either $u(0) = 0$ or $u(0) = 1$ is used in various literatures, we will define $u(0) = 1/2$, for reason to be discussed in the future. Also, at the limit $\triangle \to 0$, $\delta_{\triangle}(t)$ becomes Dirac delta discussed above:

$$\lim_{\triangle \to 0} \delta_{\triangle}(t) = \delta(t) = \begin{cases} \infty & t = 0 \\ 0 & t \ne 0 \end{cases} \tag{1.14}$$

which of course still satisfies the unit area condition:

$$\int_{-\infty}^{\infty} \delta(t)dt = 1 \tag{1.15}$$

In addition to the square impulse $\delta_{\triangle}(t)$, the Dirac delta can also be generated from a variety of different *nascent delta functions* as the limit when a certain parameter of the function approaches either zero or infinity. As an example of

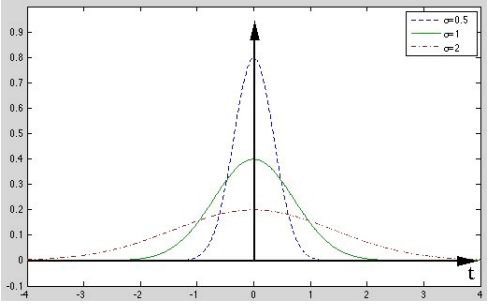

Figure 1.4 Gaussian functions with different σ values

such a function, we consider the Gaussian function:

$$g(t) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-t^2/2\sigma^2} \tag{1.16}$$

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:

$$\int_{-\infty}^{\infty} g(t)dt = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} e^{-t^2/2\sigma^2} dt = 1 \tag{1.17}$$

At the limit $\sigma \to 0$, 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:

$$\lim_{\sigma \to 0} \frac{1}{\sqrt{2\pi\sigma^2}} e^{-t^2/2\sigma^2} dt = \delta(t) \tag{1.18}$$

The argument t of a Dirac delta $\delta(t)$ may be scaled so that it becomes $\delta(at)$. In this case Eq.1.9 becomes:

$$\int_{-\infty}^{\infty} x(\tau)\delta(a\tau)\ d\tau = \int_{-\infty}^{\infty} x(\frac{u}{a})\delta(u)\frac{1}{|a|}\ du = \frac{1}{|a|}x(0) \tag{1.19}$$

where we have defined $u = a\tau$. Comparing this result with Eq.1.9, we see that

$$\delta(at) = \frac{1}{|a|}\delta(t), \quad \text{i.e.} \quad |a|\delta(at) = \delta(t) \tag{1.20}$$

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

$$\delta(f) = 2\pi\delta(\omega) \tag{1.21}$$

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 is one of the roots t_k of $f(t)$ (so that $f(t_k) = 0$). To see how such an impulse is scaled, consider the following integral:

$$\int_{-\infty}^{\infty} x(\tau)\delta(f(\tau))\ d\tau = \int_{-\infty}^{\infty} x(\tau)\delta(u)\ \frac{1}{|f'(\tau)|}\ du \tag{1.22}$$

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:

$$\int_{-\infty}^{\infty} x(\tau)\delta(f(\tau))\, d\tau = \frac{x(\tau_0)}{|f'(\tau_0)|} \tag{1.23}$$

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

$$\int_{-\infty}^{\infty} x(\tau)\delta(f(\tau))\, d\tau = \sum_k \frac{x(\tau_k)}{|f'(\tau_k)|} \tag{1.24}$$

This is the generalized sifting property of the impulse function. Based on these results, we can express the delta function as:

$$\delta(f(t)) = \sum_k \frac{\delta(t - t_k)}{|f'(\tau_k)|} \tag{1.25}$$

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.

Before leaving this section let us consider four important relationships showing that the Kronecker and Dirac delta functions can be generated respectively as the sum and integral of certain complex exponential functions. These formulas are useful in the future discussions of the different forms of the Fourier transform.

- Dirac delta as an integral of a complex exponential:

$$\int_{-\infty}^{\infty} e^{\pm j2\pi ft} dt = \delta(f) \tag{1.26}$$

- Kronecker delta as an integral of a complex exponential:

$$\frac{1}{T}\int_{T} e^{\pm j2\pi kt/T} dt = \delta[k] \tag{1.27}$$

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

$$\frac{1}{F}\sum_{k=-\infty}^{\infty} e^{\pm j2k\pi f/F} = \sum_{n=-\infty}^{\infty} \delta(f - nF) \tag{1.28}$$

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

$$\begin{aligned}\frac{1}{N}\sum_{n=0}^{N-1} e^{\pm j2\pi nm/N} &= \frac{1}{N}\sum_{n=0}^{N-1} \cos(2\pi nm/N) \pm \frac{j}{N}\sum_{n=0}^{N-1} \sin(2\pi nm/N) \\ &= \sum_{k=-\infty}^{\infty} \delta[m - kN]\end{aligned} \tag{1.29}$$

and

$$\sum_{n=0}^{N-1} \sin(2\pi nm/N) = 0$$

The proof of these four important identities is left as homework problems for the reader.

The integral in Eq.1.26 can also be written as:

$$\int_{-\infty}^{\infty} e^{\pm j2\pi ft} dt = \int_{-\infty}^{\infty} [\cos(2\pi ft) \pm j \sin(2\pi ft)] dt$$
$$= 2\int_{0}^{\infty} \cos(2\pi ft) dt = \delta(f) \tag{1.30}$$

The second equal sign is due to the fact that $\sin(2\pi ft) = -\sin(-2\pi ft)$ is odd and its integral over all time $-\infty < t < \infty$ is zero, and $\cos(2\pi ft) = \cos(-2\pi ft)$ is even and its integral over all time is twice the integral over half time $0 < t < \infty$.

This result can be interpreted intuitively. The integral of any sinusoid over all time $-\infty < t < \infty$ is always zero, except if $f = 0$ then $\sin(0) = 0$ but $\cos(0) = 1$, and the integral over all time becomes infinity. Alternatively, if we integrate the complex exponential with respect to f, we get:

$$\int_{-\infty}^{\infty} e^{j2\pi ft} df = 2\int_{0}^{\infty} \cos(2\pi ft) df = \delta(t) \tag{1.31}$$

which can also be interpreted intuitively as a superposition of infinitely many cosine functions with progressively higher frequency f. These sinusoids cancel each other at any time $t \neq 0$ except when $t = 0$, where all cosine functions equal to 1 and their superposition becomes infinity. Similar arguments can also be made for the other three cases.

1.3 Attributes of Signals

A time signal can be characterized by the following parameters:

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

$$\mathcal{E} = \int_{-\infty}^{\infty} |x(t)|^2 dt \tag{1.32}$$

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

$$\mathcal{E} = \sum_{n=-\infty}^{\infty} |x[n]|^2 \tag{1.33}$$

where $|x(t)|^2$ or $|x[n]|^2$ represents the power of the signal. If the energy $\mathcal{E} < \infty$ contained in a signal $x(t)$ or $x[n]$ is finite, then it 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 discussed later, either

continuous or discrete, will be assumed to be such energy signals and are therefore square-integrable/summable.

- The *average power* of the signal:

$$\mathcal{P} = \lim_{T\to\infty} \frac{1}{T} \int_0^T |x(t)|^2 dt \tag{1.34}$$

or for a discrete signal:

$$\mathcal{P} = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} |x[n]|^2 \tag{1.35}$$

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* measures the similarity between two signals and is defined as:

$$\begin{aligned} r_{xy}(\tau) = x(t) \star y(t) &= \int_{-\infty}^{\infty} x(t)\ \overline{y}(t-\tau)dt = \int_{-\infty}^{\infty} x(t+\tau)\ \overline{y}(t)dt \\ &\neq \int_{-\infty}^{\infty} x(t-\tau)\ \overline{y}(t)dt = y(t) \star x(t) \end{aligned} \tag{1.36}$$

Note that $x(t) \star y(t) \neq y(t) \star x(t)$, i.e., the cross-correlation is not commutative. For discrete signal, we have

$$r_{xy}[m] = x[n] \star y[n] = \sum_{n=-\infty}^{\infty} x[n]\ y[n-m] = \sum_{n=-\infty}^{\infty} x[n+m]\ y[n] \tag{1.37}$$

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

$$r_x(\tau) = \int_{-\infty}^{\infty} x(t)\overline{x}(t-\tau)dt = \int_{-\infty}^{\infty} x(t+\tau)\overline{x}(t)dt \tag{1.38}$$

or

$$r_x[m] = \sum_{n=-\infty}^{\infty} x[n]\ \overline{x}[n-m] = \sum_{n=-\infty}^{\infty} x[n+m]\ \overline{x}[n] \tag{1.39}$$

- A random time signal $x(t)$ is called a *stochastic process* and its auto-covariance is

$$Cov_x(t,\tau) = \sigma_x^2 = E[(x(t) - \mu_x(t))\ (\overline{x}(\tau) - \overline{\mu}_x(\tau))] \tag{1.40}$$

The cross-covariance of two stochastic processes is

$$Cov_{xy}(t,\tau) = \sigma_{xy}^2 = E[(x(t) - \mu_x(t))\ (\overline{y}(\tau) - \overline{\mu}_y(\tau))] \tag{1.41}$$

Here $E(x)$ is the expectation of a random variable x as defined in Appendix B.

1.4 Signal Arithmetics and Transformations

Any of the arithmetic operations 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 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. Also, the addition and subtraction of two discrete signals can be carried out as vector operations $\boldsymbol{z} = \boldsymbol{x} \pm \boldsymbol{y}$. Obviously this kind of vector operations do not apply to multiplication or division.

Next we consider the transformations of a give continuous signal. Both the amplitude and the argument of a time function $x(t)$ can be modified by a linear transformation $y = ax + b$:

- Transformation of signal amplitude (vertical in time plot):

$$y(t) = ax(t) + x_0 = a[x(t) + x_0/a] \tag{1.42}$$

  - Translation:
    $y(t) = x(t) + x_0$ is moved either upward if $x_0 > 0$ or downward if $x_0 < 0$.
  - Scaling:
    $y(t) = ax(t)$ is either up-scaled if $|a| > 1$ or down-scaled if $|a| < 1$. The signal is also flipped (upside-down) if $a < 0$.
- Transformation of time argument t (horizontal in time plot):

$$\tau = at + t_0 = a(t + t_0/a)), \qquad y(\tau) = x(at + t_0) = x(a(t + t_0/a)) \tag{1.43}$$

  - Translation (or shifts):
    $y(t) = x(t + t_0)$ is either right-shifted if $t_0 < 0$, or left-shifted if $t_0 < 0$.
  - Scaling:
    $y(t) = x(at)$ is either compressed if $|a| > 1$, expanded if $|a| < 1$. The signal is also reversed in time if $a < 0$.

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

- Method 1:
  This is 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.

- Method 2:
  First find the values of the original signal $v_1 = x(t_1)$ and $v_2 = x(t_2)$ at two arbitrarily chosen time points $t = t_1$ and $t = t_2$. The transformed signal $y(t) = x(at + t_0)$ will take the same values v_1 and v_2 when its argument is $at + t_0 = t_1$ and $at + t_0 = t_2$, respectively. Solving these two equations for t, we get $t = (t_1 - t_0)/a$ and $t = (t_2 - t_0)/a$, at which $y(t)$ will take the value v_1 and v_2, respectively. As the time transformation $at + t_0$ is linear, the value $y(t)$ 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 signal in time:

$$x(t) = \begin{cases} t & 0 < t < 2 \\ 0 & \text{otherwise} \end{cases} \tag{1.44}$$

- Translation (Fig.1.5 (a)):

$$y(t) = x(t + 3), \qquad z(t) = x(t - 1) \tag{1.45}$$

- Expansion/compression (Fig.1.5 (b)):

$$y(t) = x(2t/3), \qquad z(t) = x(2t) \tag{1.46}$$

- Time reversal (Fig.1.5 (c)):

$$y(t) = x(-t), \qquad z(t) = x(-2t) \tag{1.47}$$

- Combination of translation, scaling and reversal:

$$y(t) = x(-2t + 3) = x(-2(t - \frac{3}{2})) \tag{1.48}$$

  – Method 1: based on the first expression $y(t) = x(-2t + 3)$ we get (Fig.1.5 (d)):

$$z(t) = x(t + 3), \quad y(t) = z(-2t) \tag{1.49}$$

  alternatively, based on the second expression of $y(t) = x(-2(t - 3/2))$ we get (Fig.1.5 (e)):

$$z(t) = x(-2t), \quad y(t) = z(t - \frac{3}{2}) \tag{1.50}$$

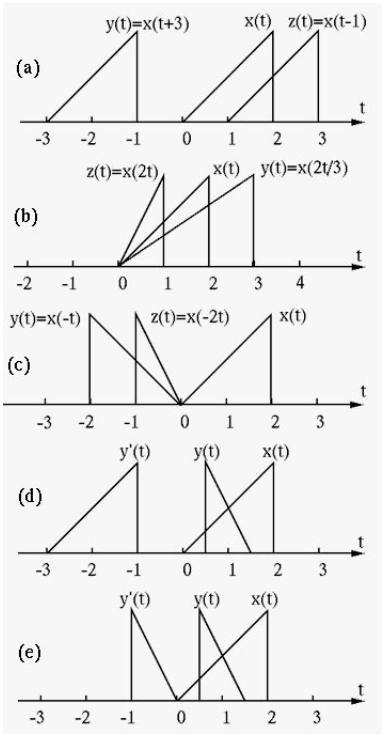

Figure 1.5 Transformation of continuous signal

– 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:

$$-2t + 3 = 0 \quad \Longrightarrow \quad t = \frac{3}{2}$$
$$-2t + 3 = 2 \quad \Longrightarrow \quad t = \frac{1}{2}$$

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.1.5(d) and (e).

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

- Down-sampling (decimation):
  keep every Nth sample and discard the rest. Signal size becomes $1/N$ of the original one.

$$x_{(N)}[n] = x[nN] \tag{1.51}$$

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

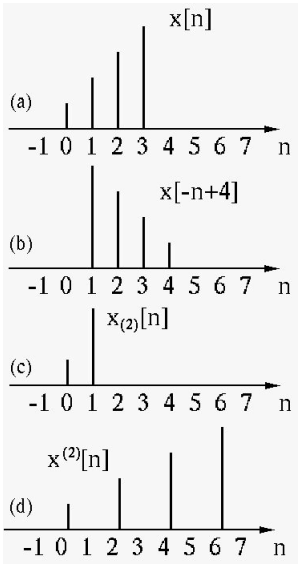

Figure 1.6 Transformation of discrete signal

- Up-sampling (interpolation by zero stuffing):
  insert $N-1$ zeros between every two consecutive samples. Signal size becomes N times the original one.

$$x^{(N)}[n] = \begin{cases} x[n/N] & n = 0, \pm N, \pm 2N, \cdots \\ 0 & \text{otherwise} \end{cases} \tag{1.52}$$

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

Example 1.2: Given $x[n]$ as shown in Fig.1.6(a), a transformation $y[n] = x[-n+4]$, shown in Fig.1.6(b), can be obtained based on two time points:

$$\begin{aligned} -n+4 = 0 \quad &\Longrightarrow \quad n = 4 \\ -n+4 = 3 \quad &\Longrightarrow \quad n = 1 \end{aligned} \tag{1.53}$$

The up and down sampling of the signal in Fig.1.6(a) can be obtained in the following table and shown in Fig.1.6(c) and (d), respectively.

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

(1.54)

1.5 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):

$$y(t) = \mathcal{O}[x(t)] \tag{1.55}$$

where the symbol $\mathcal{O}[\ ]$ represents the operation applied by the system to the input $x(t)$ to produce the output $y(t)$.

A system is *linear* if its input-output relationship satisfies both *homogeneity* and *superposition*. Let $y(t)$ be the response of a system to an input $x(t)$:

$$\mathcal{O}[x(t)] = y(t) \tag{1.56}$$

then the system is linear if the following two conditions are satisfied:

- Homogeneity:

$$\mathcal{O}\left[ax(t)\right] = a\mathcal{O}[x(t)] = ay(t) \tag{1.57}$$

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

$$\mathcal{O}\left[\sum_{n=1}^{N} x_i(t)\right] = \sum_{n=1}^{N} \mathcal{O}[x_n(t)] = \sum_{n=1}^{N} y_n(t) \tag{1.58}$$

or

$$\mathcal{O}\left[\int_{-\infty}^{\infty} x(\tau)d\tau\right] = \int_{-\infty}^{\infty} \mathcal{O}[x(\tau)]d\tau = \int_{-\infty}^{\infty} y(\tau)d\tau \tag{1.59}$$

Combining these two properties together, we have

$$\mathcal{O}\left[\sum_{n=1}^{N} a_n x_n(t)\right] = \sum_{n=1}^{N} a_n \mathcal{O}[x_n(t)] = \sum_{n=1}^{N} a_n y_n(t) \tag{1.60}$$

or

$$\mathcal{O}\left[\int_{-\infty}^{\infty} a(\tau)x(\tau)d\tau\right] = \int_{-\infty}^{\infty} a(\tau)\mathcal{O}[x(\tau)]d\tau = \int_{-\infty}^{\infty} a(\tau)y(\tau)d\tau \tag{1.61}$$

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

$$\text{if} \quad \mathcal{O}[x(t)] = y(t), \quad \text{then} \quad \mathcal{O}[x(t-\tau)] = y(t-\tau) \tag{1.62}$$

A system which is both linear and time-invariant is referred to as a *linear and time-invariant (LTI) system.*

If an LTI system's response to some input $x(t)$ is $y(t) = \mathcal{O}[x(t)]$, then its response to $dx(t)/dt$ is $dy(t)/dt$.

Proof: As this is an LTI system, we have

$$\mathcal{O}[\frac{1}{\triangle}[x(t+\triangle) - x(t)] = \frac{1}{\triangle}[y(t+\triangle t) - y(t)] \tag{1.63}$$

At the limit $\triangle \to 0$, the above becomes

$$\mathcal{O}\left[\frac{d}{dt}x(t)\right] = \frac{d}{dt}y(t) \tag{1.64}$$

Example 1.3: Check to see 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 passing R:

  $$y(t) = \mathcal{O}[x(t)] = \frac{1}{R}x(t)$$

  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:

  $$y(t) = \mathcal{O}[x(t)] = \frac{1}{R}x^2(t)$$

  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:

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

  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 $x(t)$:

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

  Consider

  $$\mathcal{O}[x_1(t) + x_2(t)] = x_1(t) + x_2(t) + a \neq \mathcal{O}[x_1(t)] + \mathcal{O}[x_2(t)] = x_1(t) + x_2(t) + 2a$$

  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 $y(t) = l - l_0 = \Delta l$ is the change of length l of the spring, or the displacement of the moving end of the spring.
  According to Hooke's law,

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

  This system is linear.
- Same as above except the output $y(t) = l$ is the length of the spring.

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

  This is not a linear system.

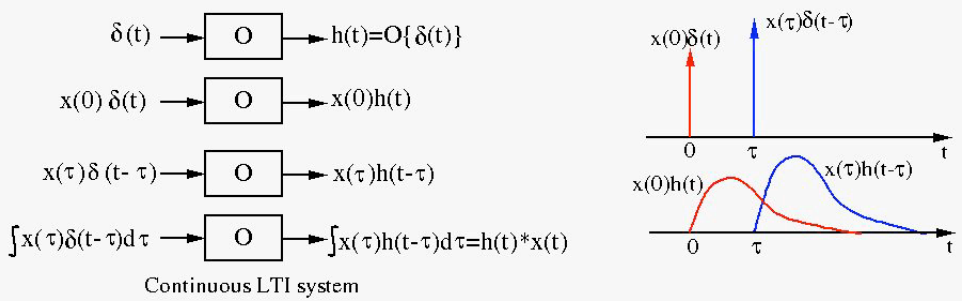

Figure 1.7 Response of a continuous LTI system

1.6 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

$$h(t) = \mathcal{O}[\delta(t)] \tag{1.65}$$

is called the *impulse response function*. Given the impulse response $h(t)$ of an LTI system, we can find its response to any input $x(t)$ that can be expressed in Eq. 1.6:

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

According to Eq. 1.59, we have

$$\begin{aligned} y(t) = \mathcal{O}[x(t)] &= \mathcal{O}\left[\int_{-\infty}^{\infty} x(\tau)\delta(t-\tau)d\tau\right] \\ &= \int_{-\infty}^{\infty} x(\tau)\mathcal{O}[\delta(t-\tau)]d\tau = \int_{-\infty}^{\infty} x(\tau)h(t-\tau)]d\tau \end{aligned} \tag{1.66}$$

This process is illustrated in Fig.1.7. 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 $y(t)$:

$$z(t) = x(t) * y(t) = \int_{-\infty}^{\infty} x(\tau)y(t-\tau)d\tau = \int_{-\infty}^{\infty} y(\tau)x(t-\tau)d\tau = y(t) * x(t) \tag{1.67}$$

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:

$$x(t) = e^{st} = e^{(\sigma+j\omega)t} = [\cos(\omega t) + j\ \sin(\omega t)]e^{\sigma t} \tag{1.68}$$

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

$$y(t) = \mathcal{O}[e^{st}] = \int_{-\infty}^{\infty} h(\tau)e^{s(t-\tau)}d\tau = e^{st}\int_{-\infty}^{\infty} h(\tau)e^{-s\tau}d\tau = H(s)e^{st} \tag{1.69}$$

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

$$H(s) = \int_{-\infty}^{\infty} h(\tau) e^{-s\tau} d\tau \tag{1.70}$$

This is called the *transfer function* 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 5. We note that Eq.1.69 is an *eigenequation*, where the constant $H(s)$ and the complex exponential e^{st} are, respectively, the *eigenvalue* and the *corresponding eigenfunction* of the LTI system. 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:

$$H(j\omega) = \int_{-\infty}^{\infty} h(\tau) e^{-j\omega\tau} d\tau \quad \text{or} \quad H(f) = \int_{-\infty}^{\infty} h(\tau) e^{-j2\pi f\tau} d\tau \tag{1.71}$$

This is called the *frequency response function* of the system, which is the *Fourier transform* of the impulse response function $h(t)$, to be discussed in Chapter 3.

Given $H(j\omega)$ of a system, its response to an input $x(t) = e^{j\omega_0 t}$ can be found by evaluating Eq.1.69 at $s = j\omega_0$:

$$y(t) = \mathcal{O}[e^{j\omega_0 t}] = H(j\omega_0) e^{j\omega_0 t} \tag{1.72}$$

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

$$x(t) = \sum_{k=-\infty}^{\infty} X_k e^{jk\omega_0 t} \tag{1.73}$$

where X_k ($-\infty < k < \infty$) are a set of constant coefficients, then, due to the linearity of the system, its output is:

$$\begin{aligned} y(t) &= \mathcal{O}[\sum_{k=-\infty}^{\infty} X_k e^{jk\omega_0 t}] = \sum_{k=-\infty}^{\infty} X_k \mathcal{O}[e^{jk\omega_0 t}] \\ &= \sum_{k=-\infty}^{\infty} X_k H(jk\omega_0) e^{jk\omega_0 t} = \sum_{k=-\infty}^{\infty} Y_k e^{jk\omega_0 t} \end{aligned} \tag{1.74}$$

where the kth coefficient Y_k is defined as $Y_k = X_k H(jk\omega_0)$. The significance of this result is that we can obtain the response of an LTI system described by $H(s)$ to any input $x(t)$ in the form of a linear combination of a set of complex exponentials. This is an important conclusion of the Fourier transform theory considered in Chapter 3.

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

$$\text{if } |x(t)| < B_x \quad \text{then} \quad |y(t)| < B_y \tag{1.75}$$

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

$$y(t) = h(t) * x(t) = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau \tag{1.76}$$

we have:

$$\begin{aligned} |y(t)| &= \Big| \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau \Big| \leq \int_{-\infty}^{\infty} |h(\tau)||x(t-\tau)|d\tau \\ &< B_x \int_{-\infty}^{\infty} |h(\tau)|d\tau < B_y \end{aligned} \tag{1.77}$$

which obviously requires:

$$\int_{-\infty}^{\infty} |h(\tau)|d\tau < \infty \tag{1.78}$$

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.1.78 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 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). Assuming 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:

$$y(t) = h(t) * x(t) = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau = \int_{0}^{\infty} h(\tau)x(t-\tau)d\tau \tag{1.79}$$

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

$$y(t) = h(t) * x(t) = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau = \int_{0}^{t} h(\tau)x(t-\tau)d\tau \tag{1.80}$$

1.7 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, a discrete signal can be written as:

$$x[n] = \sum_{m=-\infty}^{\infty} x[m]\delta[n-m] \tag{1.81}$$

If the impulse response of a discrete LTI system is

$$h[n] = \mathcal{O}[\delta[n]] \tag{1.82}$$

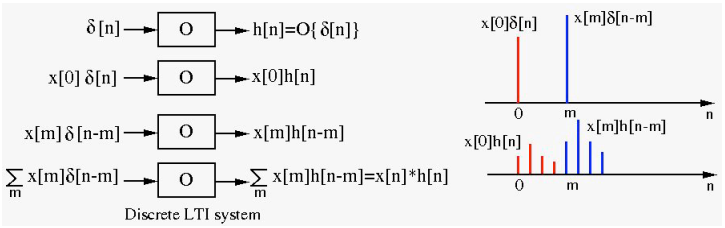

Figure 1.8 Response of a discrete LTI system

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

$$y[n] = \mathcal{O}[x[n]] = \mathcal{O}\left[\sum_{m=-\infty}^{\infty} x[m]\delta[n-m]\right] = \sum_{m=-\infty}^{\infty} x[m]\mathcal{O}[\delta[n-m]]$$
$$= \sum_{m=-\infty}^{\infty} x[m]h[n-m] \tag{1.83}$$

This process is illustrated in Fig.1.8.

The last summation in Eq.1.83 is defined called the *discrete convolution*, which is generally defined as an operation of two discrete functions $x[n]$ and $h[n]$:

$$z[n] = x[n] * y[n] = \sum_{m=-\infty}^{\infty} x[m]y[n-m] = \sum_{m=-\infty}^{\infty} y[m]x[n-m] = y[n] * x[n] \tag{1.84}$$

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 is zero until $t = 0$, we have:

$$y[n] = \sum_{m=0}^{n} x[m]h[n-m] \tag{1.85}$$

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

$$x[n] = e^{sn} = (e^s)^n = z^n \tag{1.86}$$

where $s = \sigma + j\omega$ as defined above, and z is defined as $z = e^s$, the corresponding output is

$$y[n] = \mathcal{O}[z^n] = \sum_{k=-\infty}^{\infty} h[k]z^{n-k} = z^n \sum_{k=-\infty}^{\infty} h[k]z^{-k} = H(z)z^n \tag{1.87}$$

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

$$H(z) = \sum_{k=-\infty}^{\infty} h[k]z^{-k} \tag{1.88}$$

This is called the *transfer function* of the discrete LTI system, which is the *Z-transform* of the impulse response $h[n]$ of the system, to be discussed in Chapter 5. We note that Eq.1.87 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 $s = j\omega$ ($\sigma = 0$), $z = e^s = e^{j\omega}$ and $H(z)$ becomes:

$$H(e^{j\omega}) = \sum_{k=-\infty}^{\infty} h[k]e^{-jk\omega} \tag{1.89}$$

This is the *frequency response function* of the system, which is the Fourier transform of the discrete impulse response function $h[n]$, to be discussed in Chapter 4.

Given $H(e^{j\omega})$ of a discrete system, its response to a discrete input $x[n] = z^n = e^{j\omega_0 n}$ can be found to by evaluating Eq.1.87 at $z = e^{j\omega_0}$:

$$y[n] = \mathcal{O}[e^{j\omega_0 n}] = H(e^{j\omega_0})e^{j\omega_0 n} \tag{1.90}$$

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

$$x[n] = \sum_{k=0}^{N-1} X_k e^{jk\omega_0 n/N} \tag{1.91}$$

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

$$\begin{aligned} y[n] &= \mathcal{O}[\sum_{k=0}^{N-1} X_k e^{jk\omega_0 n/N}] = \sum_{k=0}^{N-1} X_k \mathcal{O}[e^{jk\omega_0 n}] \\ &= \sum_{k=0}^{N-1} X_k H(e^{jk\omega_0})e^{jk\omega_0 n} = \sum_{k=0}^{N-1} Y_k e^{jk\omega_0 n} \end{aligned} \tag{1.92}$$

where the kth coefficient Y_k is defined as $Y_k = X_k H(e^{jk\omega_0})$. The significance of this result is that we can obtain the response of a discrete LTI system described by $H(z)$ 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 Fourier transform theory 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:

$$\text{if } |x[n]| < B_x \quad \text{then} \quad |y[n]| < B_y \tag{1.93}$$

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

$$y[n] = h[n] * x[n] = \sum_{m=-\infty}^{\infty} h[m]x[n-m] \tag{1.94}$$

we have:

$$|y[n]| = \Big| \sum_{m=-\infty}^{\infty} h[m]x[n-m] \Big| \leq \sum_{m=-\infty}^{\infty} |h[m]||x[n-n]|$$
$$< B_x \sum_{m=-\infty}^{\infty} |h[m]|d\tau < B_y \tag{1.95}$$

which obviously requires:

$$\sum_{m=-\infty}^{\infty} |h[m]| < \infty \tag{1.96}$$

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.1.96 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:

$$y[n] = h[n] * x[n] = \sum_{m=-\infty}^{\infty} h[m]x[n-m] = \sum_{m=0}^{\infty} h[m]x[n-m] \tag{1.97}$$

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

$$y[n] = h[n] * x[n] = \sum_{m=-\infty}^{\infty} h[m]x[n-m] = \sum_{m=0}^{n} h[m]x[n-m] \tag{1.98}$$

1.8 Continuous and discrete convolutions

The operation of convolution in both continuous and discrete cases defined respectively in Eqs.1.67 and 1.84 is of great importance in the future discussions. Here we further consider how such a convolution can be specifically carried out. First we consider the continuous convolution:

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

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 $y(t-\tau)$ along the τ axis as t goes from $-\infty$ to ∞;

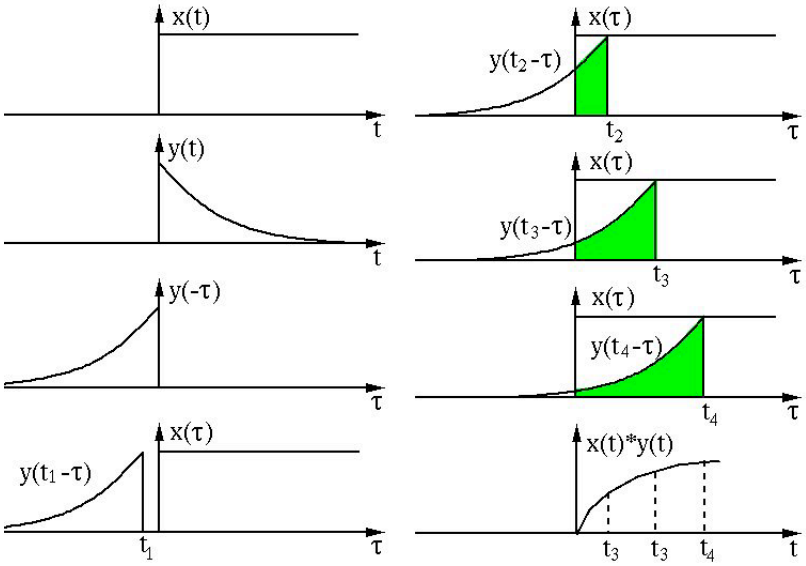

Figure 1.9 The convolution of two functions
The three steps are shown top-down, then left to right. The shaded area represents the convolution evaluated at a specific time moment such as $t = t_2$, $t = t_3$, and $t = t_4$.

3. For each time moment t, find the integral of the product $x(\tau)y(t-\tau)$ over all τ, i.e., find the area of overlap between $x(\tau)$ and $y(t-\tau)$, which is proportional to the convolution $z(t)$ at t.

Example 1.4: Let $x(t) = u(t)$ and $y(t) = e^{-at}u(t)$, the convolution of the two functions is

$$y(t) * x(t) = \int_{-\infty}^{\infty} y(\tau)x(t-\tau)d\tau = \int_0^t y(\tau)d\tau = \int_0^t e^{-a\tau}d\tau$$
$$= -\frac{1}{a}e^{-a\tau}\Big|_0^t = \frac{1}{a}(1-e^{-at})$$

This process is shown in Fig.1.9. Alternatively, the convolution can also be written as:

$$x(t) * y(t) = \int_{-\infty}^{\infty} x(\tau)y(t-\tau)d\tau = \int_0^t y(t-\tau)d\tau = \int_0^t e^{-a(t-\tau)}d\tau$$
$$= \frac{1}{a}e^{-at}e^{a\tau}\Big|_0^t = \frac{1}{a}e^{-at}(e^{at}-1) = \frac{1}{a}(1-e^{-at})$$

Although convolution and cross-correlation (Eq.1.36) are conceptually 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.

$$x(t) * y(-t) = \int_{-\infty}^{\infty} x(\tau)y(\tau - t)d\tau = r_{xy}(t) = x(t) \star y(t) \tag{1.99}$$

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

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:

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

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

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

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 $x(t) \star y(t)$:

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

Consider two cases:

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

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

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

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

Example 1.6: Let $x[n] = u[n]$ and $y[n] = a^n u[n]$ (assuming $|a| < 1$), the convolution of the two functions is:

$$y[n] * x[n] = \sum_{m=-\infty}^{\infty} y[m]x[n - m] = \sum_{m=0}^{n} y[m] = \sum_{m=0}^{n} a^m = \frac{1 - a^{n+1}}{1 - a}$$

Here we have used the geometric series formula:

$$\sum_{n=0}^{N} x^n = \frac{1 - x^{N+1}}{1 - x}$$

Alternatively, the convolution can also be written as:

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

1.9 Problems

1. Prove the identity in Eq.1.26:

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

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 $a\ sinc(af)$ of frequency f with a parameter a. A sinc function is defined as $sinc(x) = \sin(\pi x)/\pi x$.

b. Show that the following integral of this sinc function $a\ sinc(af)$ is 1 (independent of a):

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

based on the integral formula:

$$\int_{0}^{\infty} \frac{sin(x)}{x} dx = \frac{\pi}{2}$$

c. Let $a \to \infty$ and show that $a\ sinc(af)$ approaches a unit impulse:

$$\lim_{a \to \infty} s(f, a) = \delta(f)$$

Proof:

Consider the *sinc function* which can be obtained by the following integral:

$$\int_{-a/2}^{a/2} e^{\pm j2\pi ft} dt = \frac{1}{\pm j2\pi f} e^{\pm j2\pi ft} \Big|_{-a/2}^{a/2} = \frac{\sin(\pi fa)}{\pi f} = a\ sinc(af)$$

where $sinc(x)$ is the sinc function commonly defined as:

$$sinc(x) = \frac{\sin(\pi x)}{\pi x}$$

In particular when $x = 0$, we have $\lim_{x\to 0} sinc(x) = 1$. When a is increased, the function $a\ sinc(af)$ becomes narrower but taller, until when $a \to \infty$, it becomes infinity at $f = 0$ but zero everywhere else. Also, as the integral of this sinc function is unity:

$$\int_{-\infty}^{\infty} \frac{\sin(\pi f a)}{\pi f} df = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{\sin(\pi f a)}{af} d(af) = \frac{1}{\pi}\int_{-\infty}^{\infty} \frac{\sin(x)}{x} dx = 1$$

Now we see that Eq.1.26 represents a Dirac delta:

$$\int_{-\infty}^{\infty} e^{\pm j2\pi f t} dt = \lim_{a\to\infty} a\ sinc(af) = \delta(f)$$

2. Prove the identity in Eq.1.27:

$$\frac{1}{T}\int_{T} e^{\pm j2\pi kt/T} dt = \delta[k]$$

Hint: According to Euler's formula, the integrand can be expressed as:

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

Proof:

$$\frac{1}{T}\int_{T} e^{j2\pi kt/T} dt = \frac{1}{T}[\int_{T} \cos(\frac{2\pi}{T/k}t)dt \pm j\int_{T} \sin(\frac{2\pi}{T/k}t)dt]$$

The sinusoids have period T/k and their integral over T is zero, except if $k = 0$ then $\cos 0 = 1$ and $\int_T dt/T = 1$, i.e., it is a Kronecker delta.

3. Prove the identity in Eq.1.28:

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

Hint: Follow these steps:

a. Find the summation of the following series:

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

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

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

b. Obtain the value for the sum above when $a = 1$.

c. Apply the result to the left-hand side of the equation you are trying to prove, and show it is an impulse at every $f = nF$ for all integer n, i.e., it is a series of infinite impulses separated by an interval F. (Hint: consider its value for two cases: (a) $f = nF$ and (b) $f \neq nF$.)

d. Show that each of these impulses is a unit impulse by showing that the integral over any F with respect to f is 1, as shown on the right-hand side.

Proof:
First we note that if $f = nF$ is any multiple of F, then $e^{\pm j2k\pi f/F} = e^{\pm j2\pi nk} = 1$, and the summation on the left-hand side is infinity. Next we consider the following summation with the assumptions that $|a| < 1$ and $f \neq nF$:

$$\sum_{k=-\infty}^{\infty}(ae^{jx})^k = \sum_{k=0}^{\infty}(ae^{jx})^k + \sum_{k=0}^{\infty}(ae^{-jx})^k - (e^{jx})^0$$
$$= \frac{1}{1-ae^{jx}} + \frac{1}{1-ae^{-jx}} - 1 = \frac{2-a(e^{jx}+e^{-jx})}{1-a(e^{jx}+e^{-jx})+a^2} - 1$$

The first equal sign is due to the fact that the following power series converges when $|a| < 1$:

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

When $a \to 1$, this summation becomes $1 - 1 = 0$. Now we see that the summation on the left-hand side of Eq.1.28 is zero except when $f = nF$, in which case it is infinity. In other words, the summation is actually an impulse train with a gap F. Moreover, we can further show that the integral of each impulse with respect to f over one period F is 1:

$$\int_F \frac{1}{F}\sum_{k=-\infty}^{\infty} e^{\pm j2k\pi f/F}df = \frac{1}{F}\sum_{k=-\infty}^{\infty}\int_F e^{\pm j2k\pi f/F}df = \sum_{k=-\infty}^{\infty}\delta[k]$$
$$= \cdots\delta[-1] + \delta[0] + \delta[1]\cdots = \cdots + 0 + 1 + 0 + \cdots = 1$$

Here we have used the result of Eq.1.27.

4. Prove the identity in Eq.1.29:

$$\frac{1}{N}\sum_{n=0}^{N-1} e^{\pm j2\pi nm/N}$$
$$= \frac{1}{N}\sum_{n=0}^{N-1}\cos(2\pi\, nm/N) \pm \frac{1}{N}\sum_{n=0}^{N-1}\sin(2\pi\, nm/N)$$
$$= \frac{1}{N}\sum_{n=0}^{N-1}\cos(2\pi\, nm/N) = \sum_{k=-\infty}^{\infty}\delta[m-kN]$$

and

$$\sum_{n=0}^{N-1}\sin(2\pi\, nm/N) = 0$$

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

a. First, show that the summation is equal to 1 when m is any multiple of N, i.e., $m = kN$ for all $-\infty < k < \infty$.

b. Second, show that when $m \neq kN$, the summation is equal to 0 based on this formula:

$$\sum_{n=0}^{N-1} x^n = \frac{1-x^N}{1-x}$$

Proof:
The summation is obviously equal to 1 if $m = kNi$ for all integer k, otherwise the above becomes

$$\frac{1}{N}\sum_{n=0}^{N-1} e^{\pm j2\pi nk/N} = \frac{1}{N}\sum_{n=0}^{N-1}(e^{\pm j2\pi k/N})^n = \frac{1}{N}\,\frac{1-e^{\pm j2\pi kN/N}}{(1-e^{\pm j2\pi k/N})} = 0$$

The second equal sign is due to the geometric series formula

$$\sum_{n=0}^{N-1} x^n = \frac{1-x^N}{1-x}$$

and $e^{\pm j2\pi k} = 1$.

2 Vector Spaces and Signal Representation

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 $\boldsymbol{x} = [\cdots, x[n], \cdots]^T$, 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* vector addition *and* scalar multiplication *defined for its members, referred to as* vectors.

1. Vector addition *maps any two vectors $\boldsymbol{x}, \boldsymbol{y} \in V$ to another vector $\boldsymbol{x} + \boldsymbol{y} \in V$ satisfying the following properties:*
   - *Commutativity:*
   $$\boldsymbol{x} + \boldsymbol{y} = \boldsymbol{y} + \boldsymbol{x} \tag{2.1}$$
   - *Associativity:*
   $$\boldsymbol{x} + (\boldsymbol{y} + \boldsymbol{z}) = (\boldsymbol{x} + \boldsymbol{y}) + \boldsymbol{z} \tag{2.2}$$
   - *Existence of zero: there is a vector $\mathbf{0} \in V$ such that:*
   $$\mathbf{0} + \boldsymbol{x} = \boldsymbol{x} + \mathbf{0} = \boldsymbol{x} \tag{2.3}$$
   - *Existence of inverse: for any vector $\boldsymbol{x} \in V$, there is another vector $-\boldsymbol{x} \in V$ such that*
   $$\boldsymbol{x} + (-\boldsymbol{x}) = \mathbf{0} \tag{2.4}$$
2. Scalar multiplication *maps a vector $\boldsymbol{x} \in V$ and a scalar $a \in \mathbb{C}$ to another vector $a\boldsymbol{x} \in V$ with the following properties:*
   - $a(\boldsymbol{x} + \boldsymbol{y}) = a\boldsymbol{x} + a\boldsymbol{y}$
   - $(a + b)\boldsymbol{x} = a\boldsymbol{x} + b\boldsymbol{x}$
   - $ab\boldsymbol{x} = a(b\boldsymbol{x})$
   - $1\boldsymbol{x} = \boldsymbol{x}$

For example, an N-dimensional space, denoted by $\mathbb{R}^N$ or $\mathbb{C}^N$, is a vector space, whose element vector $\boldsymbol{x}$ can be written as an n-tuple, an ordered list of n elements:

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} = [x_1, x_2, \cdots, x_N]^T \tag{2.5}$$

where $x_n \in \mathbb{C}$ $(n = 1, \cdots, N)$ is a real or complex scalar. An alternative range for the index $n = 0, \cdots, N-1$ may also be used in the future for convenience. The dimensionality n may be extended to infinity for an infinite-dimensional space. Through out the future discussion, a vector is always represented as a column vector, or the transpose of a row vector.

A vector space can be defined to contain all M by N matrices composed of n column (or row) vectors:

$$\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N] = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1N} \\ x_{21} & x_{22} & \cdots & x_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ x_{M1} & x_{M2} & \cdots & x_{MN} \end{bmatrix} \tag{2.6}$$

where the nth column is an M-D vector $\boldsymbol{x}_n = [x_{1n}, \cdots, x_{Mn}]^T$. Such a matrix can be converted to an MN-D vector by cascading all the column (or row) vectors.

In the N-D space $\mathbb{R}^N$ or $\mathbb{C}^N$, the two operations generally defined above take the following forms:

- A vector $\boldsymbol{x}$ can be multiplied by a real or complex scalar factor a to become

$$a\boldsymbol{x} = [ax_1, ax_2, \cdots, ax_N]^T \tag{2.7}$$

  If $a = 1/b$, the above can also be written as $\boldsymbol{x}/b$.
- The *sum* of two vectors is defined as

$$\boldsymbol{x} + \boldsymbol{y} = [x_1 + y_1, x_2 + y_2, \cdots, x_N + y_N]^T \tag{2.8}$$

  Based on this operation, the difference between the two vectors can also be defined:

$$\boldsymbol{x} - \boldsymbol{y} = \boldsymbol{x} + (-\boldsymbol{y}) = [x_1 - y_1, x_2 - y_2, \cdots, x_n - y_N]^T \tag{2.9}$$

- The *zero vector* is a special vector with all components equal to zero:

$$\boldsymbol{0} = [0, 0, \cdots, 0]^T \tag{2.10}$$

As another example, a vector space V can also be a set containing all continuous functions $x(t)$ (real or complex valued) defined over a specific range $a \le t \le b$, which could be infinite if $a = -\infty$ and/or $b = \infty$. Any function $\boldsymbol{x} = x(t) \in V$ can be added to another one $\boldsymbol{y} = y(t) \in V$ to get $\boldsymbol{x} + \boldsymbol{y} = x(t) + y(t) \in V$, or multiplied by a scalar a to get $a\boldsymbol{x} = ax(t) \in V$. It can be shown that these operations satisfy all the conditions in the definition of a vector space.

Note that in our discussion, the term "vector", represented by $\boldsymbol{x}$, may have two interpretations. First, it can be an element of any vector space V in the general sense, such as a function vector $\boldsymbol{x} = x(t)$. Second, it can also mean specifically an N-D vector $\boldsymbol{x} = [x_1, \cdots, x_N]^T$, a tuple of n discrete elements (where n may be infinity). However, it should be clear what a vector $\boldsymbol{x}$ indicates in a specific discussion from the context.

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*

$$S_1 + S_2 = \{\boldsymbol{s}_1 + \boldsymbol{s}_2 | \boldsymbol{s}_1 \in S_1, \boldsymbol{s}_2 \in S_2\} \tag{2.11}$$

In particular, if S_1 and S_2 are mutually exclusive:

$$S_1 \cap S_2 = \{0\} \tag{2.12}$$

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:

$$V = \oplus_{i=1}^{n} S_i = S_1 \oplus \cdots \oplus S_n \tag{2.13}$$

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

$$S_i \cap \left(\sum_{i \neq j} S_j \right) = \{0\} \tag{2.14}$$

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*

$$\boldsymbol{p}_{S_1,S_2}(\boldsymbol{s}_1 + \boldsymbol{s}_2) = \boldsymbol{s}_1, \quad (\boldsymbol{s}_1 \in S_1,\ \boldsymbol{s}_2 \in S_2) \tag{2.15}$$

is called the projection of $s_1 + s_2$ 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 $\boldsymbol{x}, \boldsymbol{y} \in V$ to a scalar $< \boldsymbol{x}, \boldsymbol{y} > \in \mathbb{C}$ and satisfies the following conditions:*

- *Positive definiteness:*

$$< \boldsymbol{x}, \boldsymbol{x} > \geq 0, \qquad < \boldsymbol{x}, \boldsymbol{x} >= 0 \quad \textit{iff } \boldsymbol{x} = \boldsymbol{0} \tag{2.16}$$

- *Conjugate symmetry:* [1]

$$< \boldsymbol{x}, \boldsymbol{y} >= \overline{< \boldsymbol{y}, \boldsymbol{x} >} \tag{2.17}$$

[1] The over-line indicates the complex conjugate of a complex value, i.e., $\overline{u + jv} = u - jv$ (where $j = \sqrt{-1}$ is the imaginary unit).

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

$$< \boldsymbol{x}, \boldsymbol{y} >=< \boldsymbol{y}, \boldsymbol{x} > \tag{2.18}$$

- *Linearity in the first variable:*

$$< a\boldsymbol{x} + b\boldsymbol{y}, \boldsymbol{z} >= a < \boldsymbol{x}, \boldsymbol{z} > +b < \boldsymbol{y}, \boldsymbol{z} > \tag{2.19}$$

where $a, b \in \mathbb{C}$*. As a special case, when* $b = 0$*, we have:*

$$< a\boldsymbol{x}, \boldsymbol{y} >= a < \boldsymbol{x}, \boldsymbol{y} >, \quad < \boldsymbol{x}, a\boldsymbol{y} >= \overline{a} < \boldsymbol{x}, \boldsymbol{y} > \tag{2.20}$$

Note that in general the linearity does not apply to the second variable (unless the coefficients are real $a, b \in \mathbb{R}$):

$$\begin{aligned} < \boldsymbol{x}, a\boldsymbol{y} + b\boldsymbol{z} >= \overline{< a\boldsymbol{y} + b\boldsymbol{z}, \boldsymbol{x} >} = \overline{a < \boldsymbol{y}, \boldsymbol{x} > +b < \boldsymbol{z}, \boldsymbol{x} >} \\ = \overline{a} < \boldsymbol{x}, \boldsymbol{y} > +\overline{b} < \boldsymbol{x}, \boldsymbol{z} >\neq a < \boldsymbol{x}, \boldsymbol{y} > +b < \boldsymbol{x}, \boldsymbol{z} > \end{aligned} \tag{2.21}$$

In general, we have:

$$\begin{aligned} < \sum_i c_i \boldsymbol{x}_i, \boldsymbol{y} >= \sum_i c_i < \boldsymbol{x}_i, \boldsymbol{y} > \\ < \boldsymbol{x}, \sum_i c_i \boldsymbol{y}_i >= \sum_i \overline{c}_i < \boldsymbol{x}, \boldsymbol{y}_i > \end{aligned} \tag{2.22}$$

All vector spaces discussed in the future will be assumed to be inner product spaces. Some examples of the inner product include:

- In an N-D vector space, the inner product, also called the *dot product*, of two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ is defined as:

$$< \boldsymbol{x}, \boldsymbol{y} >= \boldsymbol{x}^T \overline{\boldsymbol{y}} = \boldsymbol{y}^* \boldsymbol{x} = [x_1, x_2, \cdots, x_N] \begin{bmatrix} \overline{y}_1 \\ \overline{y}_2 \\ \cdots \\ \overline{y}_N \end{bmatrix} = \sum_{n=1}^{N} x_n \overline{y}_n \tag{2.23}$$

  where $\boldsymbol{y}^* = \overline{\boldsymbol{y}}^T$ is the conjugate transpose of $\boldsymbol{y}$.
- In a space of 2-D matrices $\boldsymbol{X}_{M \times N}$ containing $M \times N$ elements x_{mn} $(m = 1, \cdots, M,\ n = 1, \cdots, N)$, the inner product of two such matrices $\boldsymbol{X}$ and $\boldsymbol{Y}$ is defined as:

$$< \boldsymbol{X}, \boldsymbol{Y} >= \sum_{m=1}^{M} \sum_{n=1}^{N} x_{mn} \overline{y}_{mn} \tag{2.24}$$

  This inner product is equivalent to Eq.2.23 if we cascade the column (or row) vectors of two arrays $\boldsymbol{X}$ and $\boldsymbol{Y}$ to form two MN-D vectors.
- In a function space, the inner product of two function vectors $x(t)$ and $y(t)$ is defined as:

$$< x(t), y(t) >= \int_a^b x(t)\overline{y(t)}dt = \overline{\int_a^b \overline{x(t)} y(t) dt} = \overline{< y(t), x(t) >} \tag{2.25}$$

In particular, Eq.1.9 for the sifting property of the delta function $\delta(t)$ is an inner product:

$$< x(t), \delta(t) >= \int_{-\infty}^{\infty} x(\tau)\delta(\tau)d\tau = x(0)$$

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

$$< x, y >= E[x\overline{y}] \tag{2.26}$$

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

$$\sigma_{xy}^2 = E[(x-\mu_x)\overline{(y-\mu_y)}] = E(x\overline{y}) - \mu_x\overline{\mu}_y = E(x\overline{y}) =< x, y > \tag{2.27}$$

Based on inner product, the following important concepts can be defined:

Definition 2.5. *If the inner product of two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ is zero, $< \boldsymbol{x}, \boldsymbol{y} >= 0$, they are orthogonal (perpendicular) to each other, denoted by $\boldsymbol{x} \perp \boldsymbol{y}$.*

Definition 2.6. *A vector space with inner product defined is called an inner product space. When the inner product is defined, $\mathbb{C}^N$ is called a unitary space and $\mathbb{R}^N$ is called a Euclidean space.*

Definition 2.7. *The norm of a vector $\boldsymbol{x} \in V$ is defined as:*

$$||\boldsymbol{x}|| = \sqrt{< \boldsymbol{x}, \boldsymbol{x} >} =< \boldsymbol{x}, \boldsymbol{x} >^{1/2}, \qquad or \qquad ||\boldsymbol{x}||^2 =< \boldsymbol{x}, \boldsymbol{x} > \tag{2.28}$$

The norm $||\boldsymbol{x}||$ is nonnegative and it is zero if and only if $\boldsymbol{x} = \boldsymbol{0}$. In particular, if $||\boldsymbol{x}|| = 1$, $\boldsymbol{x}$ is *normalized* and is called *a unit vector*. Any vector can be normalized if divided by its own norm: $\boldsymbol{x}/||\boldsymbol{x}||$. The vector norm squared $||\boldsymbol{x}||^2 =< \boldsymbol{x}, \boldsymbol{x} >$ can be considered as the energy of the vector.

In an N-D unitary space, the norm of a vector $\boldsymbol{x} = [x_1, \cdots, x_N]^T \in \mathbb{C}^N$ is:

$$||\boldsymbol{x}|| = \sqrt{< \boldsymbol{x}, \boldsymbol{x} >} = \sqrt{\boldsymbol{x}^T\overline{\boldsymbol{x}}} = \left[\sum_{n=1}^{N} x_n\overline{x}_n\right]^{1/2} = \left[\sum_{n=1}^{N} |x_n|^2\right]^{1/2} \tag{2.29}$$

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

$$\mathcal{E} = ||\boldsymbol{x}||^2 =< \boldsymbol{x}, \boldsymbol{x} >= \sum_{n=1}^{N} |x_n|^2 \tag{2.30}$$

The concept of N-D 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 to converge to a finite value, i.e., the vector $\boldsymbol{x}$ is an energy signal

containing finite energy:

$$\sum_{n=-\infty}^{\infty} |x_n|^2 < \infty \tag{2.31}$$

All such vectors $\boldsymbol{x}$ satisfying the above are said to be *square-summable* and they form a vector space called l^2 space denoted as $l^2(\mathbb{Z})$.

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

$$||\boldsymbol{x}|| = \left(\int_a^b x(t)\overline{x(t)}\, dt \right)^{1/2} = \left(\int_a^b |x(t)|^2\, dt \right)^{1/2} \tag{2.32}$$

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:

$$\int_{-\infty}^{\infty} |x(t)|^2 dt < \infty \tag{2.33}$$

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

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

Theorem 2.1. Cauchy-Schwarz inequality

$$| < \boldsymbol{x}, \boldsymbol{y} > |^2 \ \leq < \boldsymbol{x}, \boldsymbol{x} >< \boldsymbol{y}, \boldsymbol{y} >, \qquad \textit{i.e.,} \quad | < \boldsymbol{x}, \boldsymbol{y} > | \ \leq \ ||\boldsymbol{x}|| \cdot ||\boldsymbol{y}|| \tag{2.34}$$

Proof: Let $\lambda \in \mathbb{C}$ be an arbitrary complex number and we have:

$$< \boldsymbol{x} - \lambda \boldsymbol{y}, \boldsymbol{x} - \lambda \boldsymbol{y} >= ||\boldsymbol{x}||^2 - \overline{\lambda} < \boldsymbol{x}, \boldsymbol{y} > -\lambda < \boldsymbol{y}, \boldsymbol{x} > +|\lambda|^2 ||\boldsymbol{y}||^2 \geq 0 \tag{2.35}$$

Obviously the Cauchy-Schwarz inequality holds if $\boldsymbol{y} = \boldsymbol{0}$. Otherwise, we let

$$\lambda = \frac{< \boldsymbol{x}, \boldsymbol{y} >}{||\boldsymbol{y}||^2}, \quad \text{then} \quad \overline{\lambda} = \frac{< \boldsymbol{y}, \boldsymbol{x} >}{||\boldsymbol{y}||^2}, \quad |\lambda|^2 = \frac{| < \boldsymbol{x}, \boldsymbol{y} > |^2}{||\boldsymbol{y}||^4} \tag{2.36}$$

Substitute it in the inequality above, we get

$$||\boldsymbol{x}||^2 - \frac{| < \boldsymbol{x}, \boldsymbol{y} > |^2}{||\boldsymbol{y}||^2} \geq 0, \qquad \text{i.e.,} \quad | < \boldsymbol{x}, \boldsymbol{y} >\leq ||\boldsymbol{x}|| \cdot ||\boldsymbol{y}|| \tag{2.37}$$

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

$$\theta = \cos^{-1}\left(\frac{< \boldsymbol{x}, \boldsymbol{y} >}{||\boldsymbol{x}||\ ||\boldsymbol{y}||} \right) = \cos^{-1}\left(\frac{< \boldsymbol{x}, \boldsymbol{y} >}{\sqrt{< \boldsymbol{x}, \boldsymbol{x} >}\sqrt{< \boldsymbol{y}, \boldsymbol{y} >}} \right) \tag{2.38}$$

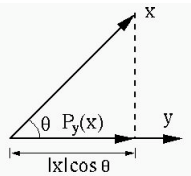

Figure 2.1 Orthogonal Projection

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

$$< \boldsymbol{x}, \boldsymbol{y} >= ||\boldsymbol{x}|| \cdot ||\boldsymbol{y}|| \cos\theta \tag{2.39}$$

If $\theta = 0$ or $\cos\theta = 1$, i.e., the two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ are collinear, the Cauchy-Schwarz inequality becomes an equality. On the other hand, if $< \boldsymbol{x}, \boldsymbol{y} >= 0$, i.e., $\boldsymbol{x}$ and $\boldsymbol{y}$ are orthogonal or perpendicular to each other, then the angle between them becomes $\theta = cos^{-1}0 = \pi/2$.

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

$$\boldsymbol{p_y}(\boldsymbol{x}) = \frac{< \boldsymbol{x}, \boldsymbol{y} >}{||\boldsymbol{y}||}\frac{\boldsymbol{y}}{||\boldsymbol{y}||} = \frac{< \boldsymbol{x}, \boldsymbol{y} >}{< \boldsymbol{y}, \boldsymbol{y} >}\boldsymbol{y} = ||\boldsymbol{x}|| \cos\theta \frac{\boldsymbol{y}}{||\boldsymbol{y}||} \tag{2.40}$$

where θ is the angle between the two vectors.

The projection $\boldsymbol{p_y}(\boldsymbol{x})$ is a vector and its norm or length is a scalar denoted by:

$$p_{\boldsymbol{y}}(\boldsymbol{x}) = ||\boldsymbol{p_y}(\boldsymbol{x})|| = \frac{< \boldsymbol{x}, \boldsymbol{y} >}{||\boldsymbol{y}||} = ||\boldsymbol{x}|| \cos\theta \tag{2.41}$$

which is sometimes also referred to as the projection. The projection $\boldsymbol{p_y}(\boldsymbol{x})$ is illustrated in Fig.2.1. In particular, if $\boldsymbol{y}$ is a unit vector, i.e., $||\boldsymbol{y}|| = 1$, we have

$$\boldsymbol{p_y}(\boldsymbol{x}) =< \boldsymbol{x}, \boldsymbol{y} > \boldsymbol{y}, \quad ||\boldsymbol{p_y}(\boldsymbol{x})|| =< \boldsymbol{x}, \boldsymbol{y} > \tag{2.42}$$

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

Definition 2.10. *Let $S \subset V$ and $S \oplus S^{\perp} = V$. Then*

$$\boldsymbol{p}_S(\boldsymbol{s} + \boldsymbol{r}) = \boldsymbol{s}, \quad (\boldsymbol{s} \in S,\ \boldsymbol{r}_2 \in S^{\perp}) \tag{2.43}$$

is called the orthogonal projection of $\boldsymbol{s} + \boldsymbol{r}$ onto S.

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

The angle between the two vectors is

$$\theta = \cos^{-1}(\frac{< \boldsymbol{x}, \boldsymbol{y} >}{\sqrt{< \boldsymbol{x}, \boldsymbol{x} >< \boldsymbol{y}, \boldsymbol{y} >}}) = \cos^{-1}(\frac{5}{\sqrt{5 \times 10}}) = \cos^{-1} 0.707 = 45^{\circ} \tag{2.44}$$

The projection of $\boldsymbol{x}$ on $\boldsymbol{y}$ is:

$$\boldsymbol{p}_{\boldsymbol{y}}(\boldsymbol{x}) = \frac{<\boldsymbol{x},\boldsymbol{y}>}{<\boldsymbol{y},\boldsymbol{y}>}\boldsymbol{y} = \frac{5}{10}\begin{bmatrix}3\\1\end{bmatrix} = \begin{bmatrix}1.5\\0.5\end{bmatrix} \tag{2.45}$$

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

$$p_{\boldsymbol{z}}(\boldsymbol{x}) = ||\boldsymbol{p}_{\boldsymbol{z}}(\boldsymbol{x})|| = <\boldsymbol{x},\boldsymbol{z}> = [1,\ 2]\begin{bmatrix}3\\1\end{bmatrix}/\sqrt{10} = 5/\sqrt{10} \approx 1.58 \tag{2.46}$$

Definition 2.11. *The distance between two vectors $\boldsymbol{x},\boldsymbol{y}$ is defined as*

$$d(\boldsymbol{x},\boldsymbol{y}) = ||\boldsymbol{x}-\boldsymbol{y}|| \tag{2.47}$$

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

- *Nonnegative:* $d(\boldsymbol{x},\boldsymbol{y}) \geq 0$. $d(\boldsymbol{x},\boldsymbol{y}) = 0$ *iff* $\boldsymbol{x} = \boldsymbol{y}$.
- *Symmetric:* $d(\boldsymbol{x},\boldsymbol{y}) = d(\boldsymbol{y},\boldsymbol{x})$.
- *Triangle inequality:* $d(\boldsymbol{x},\boldsymbol{y}) \leq d(\boldsymbol{x},\boldsymbol{z}) + d(\boldsymbol{z},\boldsymbol{y})$

Proof: The first two conditions are self-evident based on the definition. We now show that the distance $d(\boldsymbol{x},\boldsymbol{y})$ does indeed satisfy the third condition. Consider the following:

$$\begin{aligned}
&||\boldsymbol{u}+\boldsymbol{v}||^2 = <\boldsymbol{u}+\boldsymbol{v},\boldsymbol{u}+\boldsymbol{v}> = ||\boldsymbol{u}||^2 + <\boldsymbol{u},\boldsymbol{v}> + <\boldsymbol{v},\boldsymbol{u}> + ||\boldsymbol{v}||^2 \\
&= ||\boldsymbol{u}||^2 + 2\,Re<\boldsymbol{u},\boldsymbol{v}> + ||\boldsymbol{v}||^2 \leq ||\boldsymbol{u}||^2 + 2\,|<\boldsymbol{u},\boldsymbol{v}>| + ||\boldsymbol{v}||^2 \\
&\leq ||\boldsymbol{u}||^2 + 2\,||\boldsymbol{u}||\,||\boldsymbol{v}|| + ||\boldsymbol{v}||^2 = (||\boldsymbol{u}|| + ||\boldsymbol{v}||)^2
\end{aligned} \tag{2.48}$$

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

$$||\boldsymbol{u}+\boldsymbol{v}|| \leq ||\boldsymbol{u}|| + ||\boldsymbol{v}|| \tag{2.49}$$

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

$$||\boldsymbol{x}-\boldsymbol{y}|| \leq ||\boldsymbol{x}-\boldsymbol{z}|| + ||\boldsymbol{z}-\boldsymbol{y}||, \quad \text{i.e.,} \quad d(\boldsymbol{x},\boldsymbol{y}) \leq d(\boldsymbol{x},\boldsymbol{z}) + d(\boldsymbol{z},\boldsymbol{y}) \tag{2.50}$$

This completes the proof.

When distance is defined between any two elements of a vector space, the space becomes a *metric space*. In a unitary space $\mathbb{C}^N$, the distance between $\boldsymbol{x}$ and $\boldsymbol{y}$ is:

$$d(\boldsymbol{x},\boldsymbol{y}) = ||\boldsymbol{x}-\boldsymbol{y}|| = \left(\sum_{n=1}^{N}|x_n - y_n|^2\right)^{1/2} \tag{2.51}$$

This is the *Euclidean distance*, which can be considered as a special case of the more general *p-norm distance* defined as:

$$d_p(\boldsymbol{x}, \boldsymbol{y}) = \left(\sum_{n=1}^{N} |x_n - y_n|^p \right)^{1/p} \tag{2.52}$$

Obviously the Euclidean distance is the p-norm distance when $p = 2$. Also, other commonly used p-norm distances include:

$$d_1(\boldsymbol{x}, \boldsymbol{y}) = \sum_{n=1}^{N} |x_n - y_n| \tag{2.53}$$

$$d_\infty(\boldsymbol{x}, \boldsymbol{y}) = max(|x_1 - y_1|, \cdots, |x_N - y_N|) \tag{2.54}$$

In a function space, the distance between two functions $x(t)$ and $y(t)$ is:

$$d(x(t), y(t)) = ||x(t) - y(t)|| = \left(\int_a^b |x(t) - y(t)|^2 \, dt \right)^{1/2} \tag{2.55}$$

Definition 2.12. *Two subspaces $S_1 \subset V$ and $S_2 \subset V$ of a vector space V are orthogonal, denoted by $S_1 \perp S_2$, if $\boldsymbol{s}_1 \perp \boldsymbol{s}_2$ for any $\boldsymbol{s}_1 \in S_1$ and $\boldsymbol{s}_2 \in S_2$. In particular, if one of the subsets contains only one vector $S_1 = \{\boldsymbol{s}_1\}$, then the vector is orthogonal to the other subset, i.e., $\boldsymbol{s}_1 \perp S_2$.*

Definition 2.13. *The orthogonal complement of a subspace $S \subset V$ is the set of all vectors in V that are orthogonal to S:*

$$S^\perp = \{\boldsymbol{v} \in V \mid \boldsymbol{v} \perp S\} \tag{2.56}$$

Obviously we have

$$S \cap S^\perp = \{0\}, \quad \textit{and} \quad S \oplus S^\perp = V \tag{2.57}$$

In general, more than two subspaces $S_i \subset V$ $(i = 1, \cdots, n)$ are orthogonal complement if

$$V = S_1 \oplus \cdots \oplus S_n, \quad \textit{and} \quad S_i \perp S_j, \quad (i \neq j) \tag{2.58}$$

2.1.3 Bases of a Vector Space

Definition 2.14. *The linear span of a set of vectors $\boldsymbol{b}_i$, $(i = 1, \cdots, n)$ in space V is a subspace $W \subset V$:*

$$W = span(\boldsymbol{b}_1, \cdots, \boldsymbol{b}_N) = \{\sum_{n=1}^{N} c_n \boldsymbol{b}_n | c_n \in \mathbb{C}\} \tag{2.59}$$

Definition 2.15. *A set of vectors $\boldsymbol{v}_n$, $(n = 1, \cdots, N)$ in an N-D space V forms a basis of the space if they are linearly independent, i.e., they span the space*

so that any vector $\boldsymbol{x} \in V$ in the space can be uniquely expressed as a linear combination of these basis vectors:

$$\boldsymbol{x} = \sum_{n=1}^{N} c_n \boldsymbol{b}_n \tag{2.60}$$

The basis vectors are linearly independent, i.e., none of them can be represented as a linear combination of the rest, and including any extra vector in the basis it would no longer be linearly independent. These basis vectors are also said to be complete as 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 that can represent any vector in the space. Also it is obvious that there are infinitely many bases that can all span the same space, as, any rotation of a given basis will result in a different basis. This idea is of great importance in our future discussion.

Consider as an example the N-D unitary space $\mathbb{C}^N$. How many linearly independent vectors does a basis need to contain for it to span the entire space? Let us assume for now this basis consists of M linearly independent vectors $\{\boldsymbol{b}_1, \cdots, \boldsymbol{b}_M\}$, where each vector $\boldsymbol{b}_i$ is an N-D vector. Then any $\boldsymbol{x} \in \mathbb{C}^N$ can be represented as a linear combination of these basis vectors:

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix}_{N\times 1} = \sum_{m=1}^{M} c_m \boldsymbol{b}_m = [\boldsymbol{b}_1, \cdots, \boldsymbol{b}_M]_{N\times M} \begin{bmatrix} c_1 \\ \vdots \\ c_M \end{bmatrix}_{M\times 1} = \boldsymbol{B}\boldsymbol{c} \tag{2.61}$$

where $\boldsymbol{B} = [\boldsymbol{b}_1, \cdots, \boldsymbol{b}_M]$ is an N by M matrix composed of the M N-D basis vectors as its columns, and $\boldsymbol{c} = [c_1, \cdots, c_M]^T$ is a vector composed of M *coefficients*, which can be found by solving this linear equation system.

Obviously for the solution to exist, the number of equations N can be no greater than the number of unknowns M. On the other hand, M can be no greater then N as there can be no more than N independent basis vectors in this N-D space. It is therefore clear that a basis of an N-D space must have exactly $M = N$ basis vectors. Now the matrix $\boldsymbol{B}$ becomes an N by N square matrix with full rank (as all column vectors are independent), and the coefficients can be obtained by solving the system with N unknowns and N equations:

$$\boldsymbol{c} = \begin{bmatrix} c_1 \\ \vdots \\ c_N \end{bmatrix} = [\boldsymbol{b}_1, \cdots, \boldsymbol{b}_N]^{-1} \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} = \boldsymbol{B}^{-1}\boldsymbol{x} \tag{2.62}$$

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

As a very simple example for the basis of a vector space, recall that in a 3-D space $\mathbb{R}^3$ a vector $\boldsymbol{v} = [x, y, z]^T$ is conventionally represented as:

$$\boldsymbol{v} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = x \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + y \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} + z \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = x\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k} \tag{2.63}$$

where $\boldsymbol{i} = [1, 0, 0]^T$, $\boldsymbol{j} = [0, 1, 0]^T$, and $\boldsymbol{k} = [0, 0, 1]^T$ are the three *standard* or *canonical basis* vectors along each of the three mutually perpendicular axes. The concept of the standard basis $\{\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k}\}$ in the 3-D space can be generalized to an N-D unitary space $\mathbb{C}^N$, where the standard basis is composed of a set of n vectors defined as:

$$\boldsymbol{e}_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \boldsymbol{e}_2 = \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \quad \cdots, \quad \boldsymbol{e}_N = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \tag{2.64}$$

In general, all components of the nth standard basis vector $\boldsymbol{e}_n$ are zero except the nth one which is 1. If we denote the mth component of the nth vector $\boldsymbol{e}_n$ by e_{mn}, then we have $e_{mn} = \delta[m - n]$. Now we see that in the N-D space a vector $\boldsymbol{x} = [x_1, \cdots, x_N]^T$ is expressed as a linear combination of the N standard basis vectors:

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} = x_1\boldsymbol{e}_1 + x_2\boldsymbol{e}_2 + \cdots + x_N\boldsymbol{e}_N = \sum_{n=1}^{N} x_n\boldsymbol{e}_n \tag{2.65}$$

where the nth coordinate or component x_n is the coefficient for the ith vector $\boldsymbol{e}_n$ of the standard basis. In other words, whenever a vector is presented in the form of a tuple or column vector, it is actually represented in terms of the standard basis, which is always implicitly used to specify a vector, unless a different basis is explicitly specified.

Example 2.2:

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

$$\boldsymbol{a}_1 = 1\boldsymbol{e}_1 + 0\boldsymbol{e}_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad \boldsymbol{a}_2 = -1\boldsymbol{e}_1 + 2\boldsymbol{e}_2 = \begin{bmatrix} -1 \\ 2 \end{bmatrix}$$

As $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ are independent (none of the two can be obtained by scaling the other), they can be used as the basis vectors to span the space. Any given vector

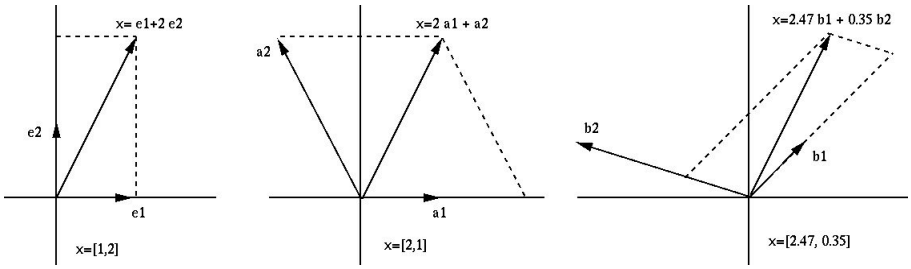

Figure 2.2 Different basis vectors of a 2-D space

such as

$$\boldsymbol{x} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} = 1\boldsymbol{e}_1 + 2\boldsymbol{e}_2 = 1 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + 2 \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

can be expressed by these basis vectors as:

$$\boldsymbol{x} = c_1\boldsymbol{a}_1 + c_2\boldsymbol{a}_2 = [\boldsymbol{a}_1, \boldsymbol{a}_2] \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}$$

i.e.,

$$c_1 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + c_2 \begin{bmatrix} -1 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 & -1 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

Solving this we get $c_1 = 2$ and $c_2 = 1$, so that $\boldsymbol{x}$ can be expressed by $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ as:

$$\boldsymbol{x} = 1\boldsymbol{a}_1 + 2\boldsymbol{a}_2 = 2 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + 1 \begin{bmatrix} -1 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

This example is illustrated in Fig.2.2

Example 2.3: The example above can also be extended to the function space spanned by two basis functions defined over $[0, 2]$:

$$a_1(t) = \begin{cases} 1 \ (0 \le t < 1) \\ 0 \ (1 \le t < 2) \end{cases}, \quad a_2(t) = \begin{cases} -1 \ (0 \le t < 1) \\ 2 \ (1 \le t < 2) \end{cases} \tag{2.66}$$

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

$$x(t) = \begin{cases} 1 \ (0 \le t < 1) \\ 2 \ (1 \le t < 2) \end{cases} \tag{2.67}$$

can be represented by the two basis functions as:

$$x(t) = c_1 a_1(t) + c_2 a_2(t) \tag{2.68}$$

where the coefficients $c_1 = 2$ and $c_2 = 1$, same as before.

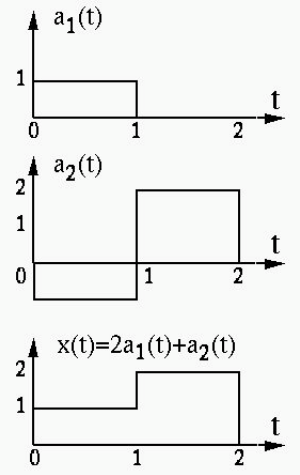

Figure 2.3 Representation of a time function by basis functions

2.1.4 Orthogonal Bases

A vector space V can be spanned by a set of orthogonal basis vectors $\{\cdots, \boldsymbol{v}_n, \cdots\}$ satisfying:

$$< \boldsymbol{v}_m, \boldsymbol{v}_n >= \delta[m-n]\,||\boldsymbol{v}_n||^2 \tag{2.69}$$

A given signal vector $\boldsymbol{x} \in V$ in the space can be expressed as:

$$\boldsymbol{x} = \sum_n c_n \boldsymbol{v}_n \tag{2.70}$$

Taking the inner product with $\boldsymbol{v}_m$ on both sides we get:

$$< \boldsymbol{x}, \boldsymbol{v}_m >=< \sum_n c_n \boldsymbol{v}_n, \boldsymbol{v}_m >= \sum_n c_n < \boldsymbol{v}_n, \boldsymbol{v}_m >= c_m ||\boldsymbol{v}_m||^2 \tag{2.71}$$

and the coefficients can be obtained as:

$$c_m = \frac{1}{||\boldsymbol{v}_m||^2} < \boldsymbol{x}, \boldsymbol{v}_m >, \qquad (m = 1, \cdots, N) \tag{2.72}$$

Now the vector can be expressed as:

$$\boldsymbol{x} = \sum_n c_n \boldsymbol{v}_n = \sum_n \frac{1}{||\boldsymbol{v}_n||^2} < \boldsymbol{x}, \boldsymbol{v}_n > \boldsymbol{v}_n = \sum_n \boldsymbol{p}_{\boldsymbol{v}_n}(\boldsymbol{x}) \tag{2.73}$$

We see that $\boldsymbol{x}$ is expressed as the vector sum of its projections $\boldsymbol{p}_{\boldsymbol{v}_n}(\boldsymbol{x})$ (Eq.2.40) onto each of the basis vectors $\boldsymbol{v}_n$. We can further normalize the orthogonal basis to get a set of orthonormal basis vectors:

$$\boldsymbol{u}_n = \frac{\boldsymbol{v}_n}{||\boldsymbol{v}_n||}, \qquad (n = 1, \cdots, N) \tag{2.74}$$

so that they satisfy $< \boldsymbol{u}_m, \boldsymbol{u}_n >= \delta[m-n]$. Now a vector $\boldsymbol{x}$ can be expressed as:

$$\boldsymbol{x} = \sum_n c_n \boldsymbol{u}_n = \sum_n < \boldsymbol{x}, \boldsymbol{u}_n > \boldsymbol{u}_n \tag{2.75}$$

where

$$c_n =< \boldsymbol{x}, \boldsymbol{u}_n >= p_{\boldsymbol{u}_n}(\boldsymbol{x}) \tag{2.76}$$

These two equations form a unitary transform pair.

In particular, in an N-D unitary space $\mathbb{C}^N$ a vector is an N-tuple, represented as a column vector containing N components: $\boldsymbol{x} = [x_1, \cdots, x_N]^T$, and the space can be spanned by a set of N orthonormal vectors $\{\boldsymbol{u}_1, \cdots, \boldsymbol{u}_N\}$:

$$< \boldsymbol{u}_m, \boldsymbol{u}_n >= \boldsymbol{u}_m^T \overline{\boldsymbol{u}}_n = \sum_{k=1}^{N} u_{km} \overline{u}_{kn} = \delta[m-n] \tag{2.77}$$

and any vector $\boldsymbol{x} \in \mathbb{C}^N$ can be expressed as a linear combination of these basis vectors:

$$\boldsymbol{x} = \sum_{n=1}^{N} c_n \boldsymbol{u}_n = [\boldsymbol{u}_1, \cdots, \boldsymbol{u}_N] \begin{bmatrix} c_1 \\ \vdots \\ c_N \end{bmatrix} = \boldsymbol{U}\boldsymbol{c} \tag{2.78}$$

where $\boldsymbol{c} = [c_1, \cdots, c_N]^T$ is an N-D coefficient vector containing the N coefficients, and $\boldsymbol{U} = [\boldsymbol{u}_1, \cdots, \boldsymbol{u}_N]$ is a unitary matrix that satisfies

$$\boldsymbol{U}^{-1} = \boldsymbol{U}^*, \quad \text{i.e.,} \quad \boldsymbol{U}\boldsymbol{U}^* = \boldsymbol{U}^*\boldsymbol{U} = \boldsymbol{I}, \tag{2.79}$$

Premultiplying $\boldsymbol{U}^{-1} = \boldsymbol{U}^*$ on both sides Eq.2.78, we get the coefficient vector:

$$\boldsymbol{U}^{-1}\boldsymbol{x} = \boldsymbol{U}^{-1}\boldsymbol{U}\boldsymbol{c} = \boldsymbol{U}^*\boldsymbol{U}\boldsymbol{c} = \boldsymbol{c} \tag{2.80}$$

Eqs. 2.78 and 2.80 can be rewritten as a transform pair:

$$\begin{cases} \boldsymbol{c} = \boldsymbol{U}^*\boldsymbol{x} \\ \boldsymbol{x} = \boldsymbol{U}\boldsymbol{c} \end{cases} \tag{2.81}$$

Alternatively, each coefficient c_n in Eq.2.78 can be obtained by taking an inner product with $\boldsymbol{u}_m$ on both sides of the equation to get:

$$< \boldsymbol{x}, \boldsymbol{u}_m >=< \sum_{n=1}^{N} c_n \boldsymbol{u}_n, \boldsymbol{u}_m >= \sum_{n=1}^{N} c_n < \boldsymbol{u}_n, \boldsymbol{u}_m >= \sum_{n=1}^{N} c_n \delta[m-n] = c_m \tag{2.82}$$

Now the transform pair above can also be written as:

$$\begin{cases} c_n =< \boldsymbol{x}, \boldsymbol{u}_n > \quad (n = 1, \cdots, N) \\ \boldsymbol{x} = \sum_{n=1}^{N} c_n \boldsymbol{u}_n = \sum_{n=1}^{N} < \boldsymbol{x}, \boldsymbol{u}_n > \boldsymbol{u}_n \end{cases} \tag{2.83}$$

or in component form as:

$$\begin{cases} c_n = \sum_{m=1}^{N} x_m \overline{u}_{mn}, \quad (n = 1, \cdots, N) \\ x_m = \sum_{n=1}^{N} c_n u_{nm}, \quad (m = 1, \cdots, N) \end{cases} \tag{2.84}$$

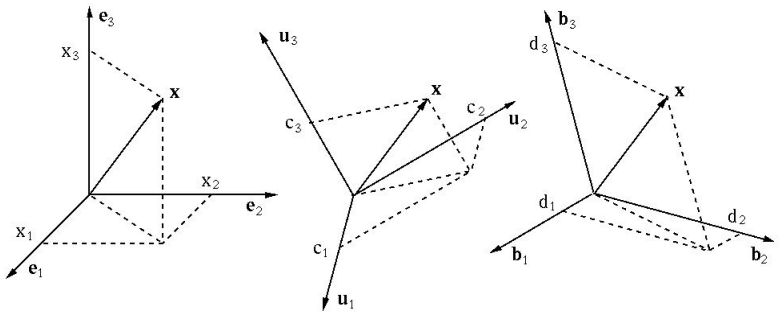

Figure 2.4 Representations of the same vector under different bases

The two equations in Eqs.2.81 (or 2.83) form a pair of discrete *unitary transforms*, the first one for the forward transform while the second one for the inverse transform. The computational complexity for either of them is $O(N^2)$, in comparison to $O(N^3)$ needed in Eq.2.62 for an arbitrary basis. The reduced complexity is certainly a main advantage of the orthogonal bases.

We see that the vector $\boldsymbol{x}$ and its coefficient vector $\boldsymbol{c}$ are related by a unitary matrix $\boldsymbol{U}$, representing a rotation in the space. Different unitary matrices represent different rotations, each corresponding to a particular set of basis vectors, the column (or row) vectors of matrix. Moreover, as the product of two unitary matrices is another unitary matrix, any two orthonormal bases are also related by a certain rotation.

The standard basis $\{\boldsymbol{e}_1, \cdots, \boldsymbol{e}_N\}$ is obviously a special orthonormal basis:

$$< \boldsymbol{e}_m, \boldsymbol{e}_n >= \boldsymbol{e}_m^T \boldsymbol{e}_n = \delta[m-n] \tag{2.85}$$

These standard basis vectors form a matrix:

$$[\boldsymbol{e}_1, \cdots, \boldsymbol{e}_N] = \begin{bmatrix} 1\,0\cdots 0 \\ 0\,1\cdots 0 \\ \vdots\,\vdots\,\ddots\,\vdots \\ 0\,0\cdots 1 \end{bmatrix} = \boldsymbol{I} \tag{2.86}$$

As a special unitary matrix $\boldsymbol{I}^{-1} = \boldsymbol{I}^T = \boldsymbol{I}$, the identity matrix corresponding unitary transform $\boldsymbol{x} = \boldsymbol{I}\boldsymbol{x}$, an identity transform, i.e., the representation of the vector is not changed.

Any given signal vector $\boldsymbol{x}$ can be equivalently represented by different bases of the space, such as the standard basis, an orthogonal basis, or some arbitrary basis, as illustrated in Fig. 2.4.

Due to the advantages of orthogonal bases, it is often desirable to convert a given non-orthogonal basis $\{\boldsymbol{a}_1, \cdots, \boldsymbol{a}_N\}$ into an orthogonal basis $\{\boldsymbol{u}_1, \cdots, \boldsymbol{u}_N\}$ by the following steps of the *Gram-Schmidt orthogonalization process*:

- $\boldsymbol{u}_1 = \boldsymbol{a}_1$
- $\boldsymbol{u}_2 = \boldsymbol{a}_2 - P_{\boldsymbol{u}_1}\boldsymbol{a}_2$

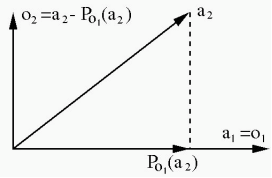

Figure 2.5 Gram-Schmidt orthogonalization

- $\boldsymbol{u}_3 = \boldsymbol{a}_3 - P_{\boldsymbol{u}_1}\boldsymbol{a}_3 - P_{\boldsymbol{u}_2}\boldsymbol{a}_3$
- .....
- $\boldsymbol{u}_N = \boldsymbol{a}_N - \sum_{n=1}^{N-1} P_{\boldsymbol{u}_n}\boldsymbol{a}_N$

Example 2.4: In Example 2.2, a vector $\boldsymbol{x} = [1,2]^T$ in a 2-D space is represented under a basis composed of $\boldsymbol{a}_1 = [1,0]^T$ and $\boldsymbol{a}_2 = [-1,2]^T$. 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 $\boldsymbol{u}_1 = \boldsymbol{a}_1 = [1,0]^T$, and

$$\boldsymbol{u}_2 = \boldsymbol{a}_2 - P_{\boldsymbol{u}_1}\boldsymbol{a}_2 = \begin{bmatrix} -1 \\ 2 \end{bmatrix} - \begin{bmatrix} -1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \end{bmatrix}$$

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

$$\boldsymbol{x} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} = 1\boldsymbol{u}_1 + 1\boldsymbol{u}_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 2 \end{bmatrix}$$

In this particular case, the two coefficients both happen to be 1, as illustrated in Fig.2.5.

Example 2.5: (Homework) Two vectors $\boldsymbol{u}_1 = [2,1]^T/\sqrt{5}$ and $\boldsymbol{u}_2 = [-1,2]^T/\sqrt{5}$ in $\mathbb{R}^2$ space are orthogonal

$$< \boldsymbol{u}_1, \boldsymbol{u}_2 >= \frac{1}{5}[2,1]\begin{bmatrix} -1 \\ 2 \end{bmatrix} = 0 \tag{2.87}$$

and normalized:

$$< \boldsymbol{u}_1, \boldsymbol{u}_1 >= \frac{1}{5}[2,1]\begin{bmatrix} 1 \\ 1 \end{bmatrix} = 1, \quad < \boldsymbol{u}_2, \boldsymbol{u}_2 >= \frac{1}{5}[-1,2]\begin{bmatrix} -1 \\ 2 \end{bmatrix} = 1 \tag{2.88}$$

and they can therefore be used as orthonormal basis vectors. A given vector $\boldsymbol{x} = [1,2]^T$ can be expressed as:

$$\boldsymbol{x} = c_1\boldsymbol{u}_1 + c_2\boldsymbol{u}_2 \tag{2.89}$$

The coefficients c_1 and c_2 can now be found by the projection method above (instead of solving a linear equation system as in the previous example):

$$c_1 =< \boldsymbol{x}, \boldsymbol{u}_1 >= \frac{1}{\sqrt{5}}[1,2]\begin{bmatrix}2\\1\end{bmatrix} = \frac{4}{\sqrt{5}}, \quad c_2 =< \boldsymbol{x}, \boldsymbol{u}_2 >= \frac{1}{\sqrt{5}}[1,2]\begin{bmatrix}-1\\2\end{bmatrix} = \frac{3}{\sqrt{5}} \tag{2.90}$$

2.1.5 Signal Representation by Standard Basis

Previously we showed that a vector $\boldsymbol{x} = [\cdots, x_m, \cdots]^T$ can be expressed by the standard basis as:

$$\boldsymbol{x} = \sum_{n=1}^{N} x_n \boldsymbol{e}_n \tag{2.91}$$

and its mth component x_m of the vector as:

$$x_m = \sum_{n=1}^{N} x_n e_{mn} = \sum_{n=1}^{N} x_n \delta[m-n], \qquad (m = 1, \cdots, N) \tag{2.92}$$

If we assume this vector is a representation of a discrete time signal $x[1], \cdots, x[N]$, we see that the equation above is the exactly the same as Eq.1.3 shown in the previous chapter. Here $e_{mn} = \delta[m-n]$ is the mth component of the nth basis vector $\boldsymbol{e}_n$, which is 0 except $m = n$. Now we see that when a discrete time signal is represented by a vector $\boldsymbol{x}$ under the standard basis, the signal is actually decomposed in time in terms of a set of components $x[m]$ each corresponding to one particular time segment $\delta[m-n]$. While the signal representation by the standard basis and the corresponding signal decomposition in time seem most reasonable thing to do, we note that it is also possible, and often beneficial, to use other bases to represent the signal and correspondingly to decompose the signal into a set of components along some dimensions other than time.

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 $x(t)$ $(0 < t < T)$. To do so, we first define a function:

$$\delta_{\triangle}(t) = \begin{cases} 1/\triangle & 0 \le t < \triangle \\ 0 & \text{otherwise} \end{cases} \tag{2.93}$$

from which a set of functions $\delta_{\triangle}(t - n\triangle)$ $(n = 0, \cdots, N-1)$ can be obtained by translation in time, and they are obviously orthonormal:

$$< \delta_{\triangle}(t - m\triangle), \delta_{\triangle}(t - n\triangle) >= \int \delta_{\triangle}(t - m\triangle)\, \delta_{\triangle}(t - n\triangle) dt = 0, \quad (m \neq n) \tag{2.94}$$

Next, we sample the continuous time signal $x(t)$ with a sampling interval $\triangle = T/N$ to get a set of discrete samples $\{x_0, \cdots, x_{N-1}\}$, and approximate the signal

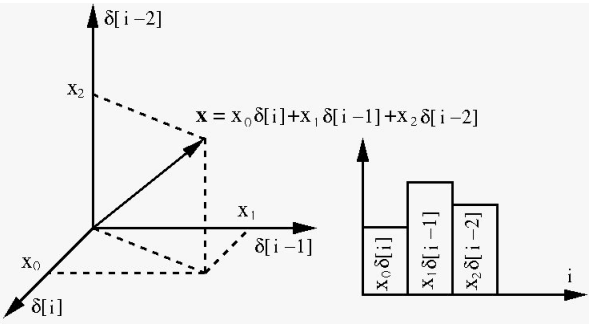

Figure 2.6 Vector representation of an N-D space (N=3)

as:

$$x(t) \approx \sum_{n=0}^{N-1} x_n \delta_{\triangle}(t - n\triangle)\ \triangle \tag{2.95}$$

Here $x_n\phi_n(t)$ represents the nth segment of the signal over the time duration $n\triangle < t < (n+1)\triangle$, as illustrated in Fig.2.6. We see that each of these functions $\delta_{\triangle}(t - n\triangle)$ represents a certain time segment, same as the standard basis $e_{mn} = \delta[m-n]$ in $\mathbb{C}^N$. However, we note that $\delta_{\triangle}(t - n\triangle)$ do not form a basis that spans the function space, as they are not complete, in the sense that they can only approximate but not precisely represent a continuous function $x(t)$ in the space. However, if we reduce the sampling interval by letting $\triangle \to 0$, we get the Dirac delta at the limit:

$$\lim_{\triangle \to 0} \delta_{\triangle}(t) = \delta(t) \tag{2.96}$$

and the summation above becomes an integral, by which the function $x(t)$ can be precisely represented:

$$x(t) = \int x(\tau)\delta(t-\tau)d\tau \tag{2.97}$$

This equation is the exactly the same as Eq. 1.6 in the previous chapter. Now we have obtained a set of basis functions $\phi_\tau(t) = \delta(t-\tau)$ (for all τ), which are complete as well as orthonormal, i.e., they form a standard basis of the function space, by which any continuous signal can be represented, just as the standard basis $\boldsymbol{e}_n$ in $\mathbb{C}^N$ by which any discrete signal can represented.

It may seem only natural to represent a discrete or continuous time signal by the corresponding standard basis representing a sequence of time segments, corresponding to the decomposition of the signal into time components. 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, so that the signal is decomposed along some different dimension other than time. Such alternative way of signal representation and decomposition may be desirable, as the signal can be more conveniently processed and analyzed, for whatever pur-

pose 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.

2.1.6 Hilbert Space

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

Definition 2.16. • *In a metric space V (an inner-product space with distance $d(\boldsymbol{x}, \boldsymbol{y})$ defined), a sequence $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots\}$ is a Cauchy sequence if for any $\epsilon > 0$, there exists an $N > 0$ such that for any $m, n > N$, $d(\boldsymbol{x}_m, \boldsymbol{x}_n) < \epsilon$.*

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

$$\lim_{m\to\infty} d(\boldsymbol{x}_m, \boldsymbol{x}) = ||\boldsymbol{x} - \boldsymbol{x}_m|| = 0 \tag{2.98}$$

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

$$\text{if } m > N, \quad \text{then}, \quad d(\boldsymbol{x}_m, \boldsymbol{x}) < \epsilon \tag{2.99}$$

- *An inner product space that is complete is a Hilbert space, denoted by H.*
- *Let $\boldsymbol{b}_n$ be a set of orthogonal vectors ($n = 1, 2, \cdots$) in H, and an arbitrary vector $\boldsymbol{x}$ is approximated in an M-D subspace by*

$$\hat{\boldsymbol{x}}_M = \sum_{n=1}^{m} c_n \boldsymbol{b}_n \tag{2.100}$$

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

$$\lim_{m\to\infty} ||\boldsymbol{x} - \hat{\boldsymbol{x}}_M||^2 = \lim_{m\to\infty} ||\boldsymbol{x} - \sum_{n=1}^{M} c_n \boldsymbol{b}_n||^2 = 0 \tag{2.101}$$

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

$$\lim_{m\to\infty} \sum_{n=1}^{M} c_n \boldsymbol{b}_n = \sum_{n=1}^{\infty} c_n \boldsymbol{b}_n = \boldsymbol{x} \tag{2.102}$$

In the following discussions, the lower and upper limits of a summation will not always be explicitly given, as the summation may be finite (e.g., from 1 to N) or infinite (e.g., from 1 or $-\infty$ to ∞), depending on each specific case.

Theorem 2.3. *Let $\{\boldsymbol{b}_n\}$ be a complete orthonormal system in a Hilbert space H:*

$$< \boldsymbol{b}_m, \boldsymbol{b}_n >= \delta[m - n] \tag{2.103}$$

Then

1. Any vector $\boldsymbol{x} \in H$ *can be expressed as:*

$$\boldsymbol{x} = \sum_n < \boldsymbol{x}, \boldsymbol{b}_n > \boldsymbol{b}_n \tag{2.104}$$

2.

$$< \boldsymbol{x}, \boldsymbol{y} >= \sum_n < \boldsymbol{x}, \boldsymbol{b}_n > \overline{< \boldsymbol{y}, \boldsymbol{b}_n >} \tag{2.105}$$

This is the Plancherel's or Parseval's theorem.

Proof: Under the basis $\{\boldsymbol{b}_n\}$, any $\boldsymbol{x} \in H$ can be written as:

$$\boldsymbol{x} = \sum_n c_n \boldsymbol{b}_n \tag{2.106}$$

Taking an inner product with $\boldsymbol{b}_m$ on both sides we get

$$< \boldsymbol{x}, \boldsymbol{b}_m >=< \sum_n c_n \boldsymbol{b}_n, \boldsymbol{b}_m >= \sum_n c_n < \boldsymbol{b}_n, \boldsymbol{b}_m >= \sum_n c_n \delta[m-n] = c_m \tag{2.107}$$

and we have

$$\boldsymbol{x} = \sum_n c_n \boldsymbol{b}_n = \sum_n < \boldsymbol{x}, \boldsymbol{b}_n > \boldsymbol{b}_n \tag{2.108}$$

This is the *generalized Fourier expansion* of a vector $\boldsymbol{x}$ in terms of basis $\{\boldsymbol{b}_n\}$, and Eq.2.107 is the *generalized Fourier coefficient*. Similarly, another vector $\boldsymbol{y} \in H$ can be written as:

$$\boldsymbol{y} = \sum_m d_m \boldsymbol{b}_m = \sum_m < \boldsymbol{y}, \boldsymbol{b}_m > \boldsymbol{b}_m \tag{2.109}$$

where $d_m =< \boldsymbol{y}, \boldsymbol{b}_m >$, then we have:

$$\begin{aligned} < \boldsymbol{x}, \boldsymbol{y} >=< \sum_n c_n \boldsymbol{b}_n, \sum_m d_m \boldsymbol{b}_m >= \sum_n c_n \sum_m \overline{d}_m < \boldsymbol{b}_n, \boldsymbol{b}_m > \\ = \sum_n c_n \sum_m \overline{d}_m \delta[m-n] = \sum_n c_n \overline{d}_n = \sum_n < \boldsymbol{x}, \boldsymbol{b}_n > \overline{< \boldsymbol{y}, \boldsymbol{b}_n >} \end{aligned} \tag{2.110}$$

This completes the proof.

In particular if $\boldsymbol{x} = \boldsymbol{y}$, then we have:

$$< \boldsymbol{x}, \boldsymbol{x} >= \sum_n |c_n|^2 = \sum_n | < \boldsymbol{x}, \boldsymbol{b}_n > |^2 \tag{2.111}$$

The coefficients c_n can be represented as vectors $\boldsymbol{c} = [\cdots, c_n, \cdots]^T$ of finite or infinite dimensions, and the equation above can also be written as:

$$||\boldsymbol{x}||^2 =< \boldsymbol{x}, \boldsymbol{x} >=< \boldsymbol{c}, \boldsymbol{c} >= ||\boldsymbol{c}||^2 \tag{2.112}$$

This is Parseval's identity indicating that a signal vector $\boldsymbol{x}$ can be equivalently represented by its Fourier coefficients with all its energy or information conserved.

The above discussion for Hilbert space can be applied to any space such as L^2-space composed of all square integrable functions $x(t)$ defined over $a < t < b$. Assume $\phi_n(t)$ is a set of complete orthonormal basis functions of the space:

$$< \phi_m(t), \phi_n(t) >= \int \phi_m(t)\overline{\phi}_n(t)dt = \delta[m-n] \tag{2.113}$$

then any continuous signal $x(t) \in L^2$ can be represented as a generalized Fourier expansion:

$$x(t) = \sum_n c_n \phi_n(t) \tag{2.114}$$

where the c_n is the generalized Fourier coefficient which can be found as

$$c_n =< x(t), \phi_n(t) >= \int x(t)\overline{\phi}_n(t)dt \tag{2.115}$$

and the squared norm of this function is:

$$||x(t)||^2 =< x(t), x(t) >= \int x(t)\overline{x}(t)dt = \sum_n |c_n|^2 = ||\boldsymbol{c}||^2 \tag{2.116}$$

2.2 Unitary Transformations and Signal Representation

2.2.1 Linear Transformations

Definition 2.17. *Let V and W be two vector spaces. A transformation is a function or mapping $T : V \to W$ that converts a vector $\boldsymbol{x} \in V$ to another vector $\boldsymbol{u} \in W$.*

If the transformation is invertible, then there exists an inverse transformation T^{-1} that converts $\boldsymbol{u} \in W$ back to $\boldsymbol{x} \in V$. The transformation and the inverse are denoted as:

$$T\boldsymbol{x} = \boldsymbol{u}, \qquad \text{and} \qquad \boldsymbol{x} = T^{-1}\boldsymbol{u} \tag{2.117}$$

$TT^{-1} = T^{-1} = I$ is an identity operator that maps a vector to itself: $TT^{-1}\boldsymbol{u} = I\boldsymbol{u}\boldsymbol{u}$ and $T^{-1}T\boldsymbol{x} = I\boldsymbol{x} = \boldsymbol{x}$.

A transformation T is linear if the following is true:

$$T(a\boldsymbol{x} + b\boldsymbol{y}) = aT\boldsymbol{x} + bT\boldsymbol{y} \tag{2.118}$$

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

If $W = V$, the linear transformation T is a linear operator.

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

$$T_d x(t) = \frac{d}{dt}x(t) = \dot{x}(t), \qquad T_i x(t) = \int x(\tau)d\tau \tag{2.119}$$

For another example, an M by N matrix $\boldsymbol{A}$ with elements $a_{mn} \in \mathbb{C}$ is a linear transformation $T_A : \mathbb{C}^N \to \mathbb{C}^M$ that maps $\boldsymbol{x} \in \mathbb{C}^N$ to $\boldsymbol{y} \in \mathbb{C}^M$:

$$T_A \boldsymbol{x} = \boldsymbol{A}\boldsymbol{x} = \boldsymbol{y} \tag{2.120}$$

or in component form:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_M \end{bmatrix}_{M\times 1} = \begin{bmatrix} a_{11} & a_{12} \cdots & a_{1N} \\ a_{21} & a_{22} \cdots & a_{2N} \\ \vdots & \vdots \ddots & \vdots \\ a_{M1} & a_{M2} \cdots & a_{MN} \end{bmatrix}_{M\times N} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}_{N\times 1} \tag{2.121}$$

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

The operation of translation $T_t \boldsymbol{x} = \boldsymbol{x} + \boldsymbol{t}$ is not a linear transformation:

$$T_t(a\boldsymbol{x} + b\boldsymbol{y}) = a\boldsymbol{x} + b\boldsymbol{y} + \boldsymbol{t} \neq aT_t\boldsymbol{x} + bT_t\boldsymbol{y} = a\boldsymbol{x} + b\boldsymbol{y} + (a+b)\boldsymbol{t} \tag{2.122}$$

Definition 2.18. *For a linear operator T in space V, if there is an operator T^* so that*

$$< T\boldsymbol{x}, \boldsymbol{y} >=< \boldsymbol{x}, T^*\boldsymbol{y} > \tag{2.123}$$

for any $\boldsymbol{x}, \boldsymbol{y} \in H$, the T^* is called the adjoint (or Hermitian adjoint) of T. If a linear operator T is its own adjoint, i.e.,

$$< T\boldsymbol{x}, \boldsymbol{y} >=< \boldsymbol{x}, T\boldsymbol{y} > \tag{2.124}$$

then T is called a self-adjoint or Hermitian operator.

In particular, in a unitary space $\mathbb{C}^N$, let $\boldsymbol{B} = \boldsymbol{A}^*$ be the adjoint of matrix $\boldsymbol{A}$, then we have:

$$< \boldsymbol{A}\boldsymbol{x}, \boldsymbol{y} >=< \boldsymbol{x}, \boldsymbol{B}\boldsymbol{y} >, \quad \text{i.e.} \quad (\boldsymbol{A}\boldsymbol{x})^T\overline{\boldsymbol{y}} = \boldsymbol{x}^T\boldsymbol{A}^T\overline{\boldsymbol{y}} = \boldsymbol{x}^T\overline{\boldsymbol{B}}\overline{\boldsymbol{y}} \tag{2.125}$$

Comparing the two sides of the last equal sign, we see that $\boldsymbol{A}^T = \overline{\boldsymbol{B}}$, i.e., the adjoint matrix $\boldsymbol{B} = \boldsymbol{A}^* = \overline{\boldsymbol{A}}^T$ is the conjugate transpose of $\boldsymbol{A}$:

$$\boldsymbol{A}^* = \overline{\boldsymbol{A}}^T \tag{2.126}$$

If $\boldsymbol{A} = \boldsymbol{A}^* = \overline{\boldsymbol{A}}^T$ is self-adjoint, it is also called a *Hermitian matrix*. In particular, when $\overline{\boldsymbol{A}} = \boldsymbol{A}$ is real, a self-adjoint matrix $\boldsymbol{A} = \boldsymbol{A}^* = \boldsymbol{A}^T$ is symmetric. Also note that we have always used $\boldsymbol{A}^*$ to denote the conjugate transpose of a matrix $\boldsymbol{A}$, but now we see that it is actually also the self-adjoint of $\boldsymbol{A}$, 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:

$$< Tx(t), y(t) >= \int Tx(t)\,\overline{y(t)}dt = \int x(t)\,\overline{T^*y(t)}dt \tag{2.127}$$

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

2.2.2 Eigenvalue problems

Definition 2.19. *If the application of an operator T to a vector $\boldsymbol{x} \in V$ results in another vector $\lambda\boldsymbol{x} \in V$, where $\lambda \in \mathbb{C}$ is a scalar:*

$$T\boldsymbol{x} = \lambda\boldsymbol{x} \tag{2.128}$$

then the scalar λ is an eigenvalue of T and vector $\boldsymbol{x}$ is the corresponding eigenvector or eigenfunctions of T, and the equation above is called the eigenequation of the operator T.

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

$$\boldsymbol{A}\boldsymbol{\phi}_n = \lambda_n\boldsymbol{\phi}_n, \qquad (n = 1, \cdots, N) \tag{2.129}$$

where λ and $\boldsymbol{\phi}$ are the eigenvalue and the corresponding eigenvector of $\boldsymbol{A}$, respectively.

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

$$D^n\ e^{st} = \frac{d^n}{dt^n}\ e^{st} = s^n\ e^{st} \tag{2.130}$$

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

$$\sum_{n=0}^{N} a_n D^n y(t) = x(t) \tag{2.131}$$

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

Perhaps the most well known eigenvalue problem in physics is the Schrodinger equation, that describes a particle in terms of its energy and the DeBroglie wave function:

$$\left[-\frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2} + V(x)\right]\psi(x) = \hat{\mathcal{H}}\psi(x) = \mathcal{E}\psi(x) \tag{2.132}$$

where

$$\hat{\mathcal{H}} = -\frac{\hbar^2}{2m}\frac{\partial^2}{\partial x^2} + V(x) \tag{2.133}$$

is the Hamiltonian operator, $\mathcal{E}$ is its eigenvalue representing the energy of the particle, and $\psi(x)$ is the corresponding eigenfunction, also called eigenstate, representing the wave function of the particle.

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 $\boldsymbol{x}$ and $\boldsymbol{y}$ be the corresponding eigenvectors:

$$T\boldsymbol{x} = \lambda\boldsymbol{x}, \quad T\boldsymbol{y} = \mu\boldsymbol{y} \tag{2.134}$$

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

$$< T\boldsymbol{x}, \boldsymbol{x} >=< \boldsymbol{x}, T^*\boldsymbol{x} >=< \boldsymbol{x}, T\boldsymbol{x} > \tag{2.135}$$

Substituting $T\boldsymbol{x} = \lambda\boldsymbol{x}$ we get

$$< \lambda\boldsymbol{x}, \boldsymbol{x} >=< \boldsymbol{x}, \lambda\boldsymbol{x} >, \quad \text{i.e.} \quad \lambda < \boldsymbol{x}, \boldsymbol{x} >= \overline{\lambda} < \boldsymbol{x}, \boldsymbol{x} > \tag{2.136}$$

As $< \boldsymbol{x}, \boldsymbol{x} >\neq 0$, we have $\lambda = \overline{\lambda}$ is real. Next, consider

$$< T\boldsymbol{x}, \boldsymbol{y} >=< \boldsymbol{x}, T\boldsymbol{y} > \tag{2.137}$$

Substituting $T\boldsymbol{x} = \lambda\boldsymbol{x}$ and $T\boldsymbol{y} = \mu\boldsymbol{y}$, we get

$$\lambda < \boldsymbol{x}, \boldsymbol{y} >= \overline{\mu} < \boldsymbol{x}, \boldsymbol{y} >= \mu < \boldsymbol{x}, \boldsymbol{y} > \tag{2.138}$$

As $\lambda \neq \mu$, we have $< \boldsymbol{x}, \boldsymbol{y} >= 0$, i.e., $\boldsymbol{x}$ and $\boldsymbol{y}$ are orthogonal. The proof of the third property is omitted.

The third property tells us 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.

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.

In an N-D space $\mathbb{C}^N$, if $\boldsymbol{A}$ is Hermitian matrix, then is satisfies $\boldsymbol{A} = \boldsymbol{A}^*$ i.e., $\boldsymbol{A}^T = \overline{\boldsymbol{A}}$, and we have:

$$< \boldsymbol{A}x, \boldsymbol{y} >= (\boldsymbol{A}x)^T\overline{\boldsymbol{y}} = \boldsymbol{x}^T\boldsymbol{A}^T\overline{\boldsymbol{y}} = \boldsymbol{x}^T\overline{\boldsymbol{A}}\overline{\boldsymbol{y}} =< \boldsymbol{x}, \boldsymbol{A}y > \tag{2.139}$$

Let λ_n and $\boldsymbol{\phi}_n$ $(n = 1, \cdots, N)$ be the eigenvalues and the corresponding eigenvectors of $\boldsymbol{A}$, then its eigenequation can be written as:

$$\boldsymbol{A}\boldsymbol{\phi}_n = \lambda_n\boldsymbol{\phi}_n \quad (n = 1, \cdots, N), \tag{2.140}$$

which can also be written as:

$$\boldsymbol{A}[\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_N] = [\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_N]\boldsymbol{\Lambda}, \quad \text{or} \quad \boldsymbol{A}\boldsymbol{\Phi} = \boldsymbol{\Phi}\boldsymbol{\Lambda} \tag{2.141}$$

where $\boldsymbol{\Phi}$ and $\boldsymbol{\Lambda}$ are two matrices defined as:

$$\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_N], \quad \boldsymbol{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_N \end{bmatrix} \tag{2.142}$$

As $\boldsymbol{A}$ is a self-adjoint operator, its eigenvalues λ_n are real, and their corresponding eigenvectors $\boldsymbol{\phi}_n$ are orthogonal:

$$< \boldsymbol{\phi}_m, \boldsymbol{\phi}_n >= \boldsymbol{\phi}_m^T \overline{\boldsymbol{\phi}}_n = \delta[m-n] \tag{2.143}$$

and they form a complete orthogonal system to span the N-D unitary space. Also $\boldsymbol{\Phi}$ a unitary matrix satisfying:

$$\boldsymbol{\Phi}^*\boldsymbol{\Phi} = \boldsymbol{I}, \quad \text{or} \quad \boldsymbol{\Phi}^* = \boldsymbol{\Phi}^{-1} \tag{2.144}$$

The eigenequation in Eq.2.141 can also be written in some other useful forms. First, pre-multiplying both sides of the equation by $\boldsymbol{\Phi}^{-1} = \boldsymbol{\Phi}^*$, we get:

$$\boldsymbol{\Phi}^{-1}\boldsymbol{A}\boldsymbol{\Phi} = \boldsymbol{\Phi}^*\boldsymbol{A}\boldsymbol{\Phi} = \boldsymbol{\Lambda} \tag{2.145}$$

i.e., the matrix $\boldsymbol{A}$ can be diagonalized by $\boldsymbol{\Phi}$. Alternatively, if we post-multiply both sides of Eq.2.141 by $\boldsymbol{\Phi}^*$, we get:

$$\boldsymbol{A} = \boldsymbol{\Phi}\boldsymbol{\Lambda}\boldsymbol{\Phi}^* = [\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \cdots, \boldsymbol{\phi}_N] \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_N \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}_1^* \\ \boldsymbol{\phi}_2^* \\ \vdots \\ \boldsymbol{\phi}_N^* \end{bmatrix} = \sum_{n=1}^{N} \lambda_n \boldsymbol{\phi}_n \boldsymbol{\phi}_n^* \tag{2.146}$$

i.e., the matrix $\boldsymbol{A}$ can be series expanded to become a linear combination of N *eigen-matrices* $\boldsymbol{\phi}_n\boldsymbol{\phi}_n^*$.

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 the function 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:

$$< D^2 x(t), y(t) >=< x(t), D^2 y(t) > \tag{2.147}$$

where $x(t)$ and $y(t)$ are two functions defined over a certain time interval such as $[0, T]$ or $[-T/2, T/2]$, and $D^2 x(t) = \ddot{x}(t)$ is the second derivative of function

$x(t)$. Using integration by parts, we can show that this equation does hold:

$$\begin{aligned} < D^2x(t), y(t) > &= \int_0^T \ddot{x}(t)\overline{y}(t)dt = \dot{x}(t)\overline{y}(t)\Big|_0^T - \int_0^T \dot{x}(t)\overline{\dot{y}}(t)dt \\ &= \dot{x}(t)\overline{y}(t)\Big|_0^T - x(t)\overline{\dot{y}}(t)\Big|_0^T + \int_0^T x(t)\overline{\ddot{y}}(t)dt =< x(t), D^2y(t) > \end{aligned} \tag{2.148}$$

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

$$[\dot{x}(t)\overline{y}(t) - x(t)\overline{\dot{y}}(t)]\Big|_0^T = 0 \tag{2.149}$$

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

$$\begin{cases} D^2\phi(t) = \lambda\phi(t), \quad \text{i.e.} \quad \ddot{\phi}(t) - \lambda\phi(t) = 0 \\ \text{subject to:} \quad \phi(0) = \phi(T), \qquad \dot{\phi}(0) = \dot{\phi}(T) \end{cases} \tag{2.150}$$

We 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 condition, we have:
   $$\phi(0) = c_2 = \phi(T) = c_1T + c_2 \tag{2.151}$$
   We get $c_1 = 0$ and $\phi(t) = c_2$, i.e., the eigenfunction corresponding to $\lambda = 0$ is any constant.
2. $\lambda > 0$:
   We assume $\phi(t) = e^{st}$ and substitute it into the equation to get
   $$(s^2 - \lambda)e^{st} = 0, \quad \text{i.e.} \quad s = \pm\sqrt{\lambda} \tag{2.152}$$
   The solution is $\phi(t) = c\, e^{\pm\sqrt{\lambda}t}$. Substituting this into the boundary condition, we have:
   $$\phi(0) = c = \phi(T) = c\, e^{\pm\sqrt{\lambda}T} \tag{2.153}$$
   Obviously the equation holds only if $\lambda = 0$, which is the same as the previous case.
3. $\lambda < 0$:
   We assume $\lambda = -\omega^2$, i.e., $\sqrt{\lambda} = \pm j\omega$, and the solution is
   $$\phi(t) = c\, e^{\pm\sqrt{\lambda}t} = c\, e^{\pm j\omega t} \tag{2.154}$$
   Substituting this into the boundary condition, we have:
   $$\phi(0) = c = \phi(T) = c\, e^{\pm j\omega T}, \quad \text{i.e.} \quad e^{\pm j\omega T} = 1 \tag{2.155}$$
   which can be solved to get
   $$\omega T = 2k\pi, \quad \text{i.e.} \quad \omega = \frac{2k\pi}{T} = 2k\pi f_0 = k\omega_0, \quad (k = 0, \pm1, \pm2, \cdots) \tag{2.156}$$

where we have defined

$$f_0 = \frac{1}{T}, \quad \omega_0 = 2\pi f_0 = \frac{2\pi}{T} \tag{2.157}$$

Now the solution is

$$\phi_n(t) = c\, e^{\pm j2n\pi/T} = c\, e^{\pm jn\omega_0}, \quad (n = 0, \pm 1, \pm 2, \cdots) \tag{2.158}$$

which includes the solution $\phi(t) = c$ corresponding to the zero eigenvalue $\lambda = 0$.

Summarizing the above, we see that the self-adjoint operator D^2 has infinitely many eigenvalues each corresponding to a different eigenfunction $\phi_n(t)$:

$$D^2\phi_n(t) = \frac{d^2}{dt^2}\phi_n(t) = \lambda_n\phi_n(t), \quad (n = 0, \pm 1, \pm 2, \cdots) \tag{2.159}$$

where the nth eigenvalue is

$$\lambda_n = -(n\omega_0)^2 = -(2n\pi/T)^2 \tag{2.160}$$

and the corresponding eigenfunction is:

$$\phi_n(t) = e^{j2n\pi t/T} = e^{j2n\pi f_0 t} = e^{jn\omega_0 t} \tag{2.161}$$

all of which are periodic with period T:

$$\phi_n(t+T) = e^{j2n\pi(t+T)/T} = e^{j2n\pi t/T}e^{j2n\pi} = \phi_n(t) \tag{2.162}$$

Here, the 0th eigenfunction $\phi_0(t) = c$ is a zero-frequency constant, the first eigenfunction

$$\phi_1(t) = e^{j\omega_0 t} = e^{j2\pi f_0 t} = \cos(2\pi f_0 t) + j\sin(2\pi f_0 t) \tag{2.163}$$

is a combination of two sinusoids of frequency $f_0 = 1/T$ or angular frequency $\omega_0 = 2\pi f_0 = 2\pi/T$, called *fundamental frequency*, and the nth ($|n| > 1$) eigenfunction

$$\phi_n(t) = e^{jn\omega_0 t} = e^{j2n\pi f_0 t} = \cos(2n\pi f_0 t) + j\sin(2n\pi f_0 t) \tag{2.164}$$

is a combination of two sinusoids of frequency nf_0 or angular frequency $n\omega_0$, n times the fundamental frequency.

These eigenvalues and their corresponding eigenfunctions have the following properties:

- All eigenvalues are discrete, there is a gap between two consecutive eigenvalues:

$$\triangle\lambda_n = \lambda_{n+1} - \lambda_n \tag{2.165}$$

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

$$\omega_0 = 2\pi f_0 = 2\pi/T \tag{2.166}$$

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

$$\phi_n(t+T) = e^{j2n\pi(t+T)/T} = e^{j2n\pi t/T}e^{j2n\pi} = e^{j2n\pi t/T} = \phi_n(t) \tag{2.167}$$

According to the properties of self-adjoint operators discussed above, the eigenfunctions $\phi_n(t)$ of D^2 form a complete orthogonal system. The orthogonality can be easily verified:

$$< \phi_m(t), \phi_n(t) >= \int_0^T e^{jm\omega_0 t}e^{-jn\omega_0 t}dt = \int_0^T e^{j2\pi(m-n)t/T}dt$$
$$= \int_0^T \cos(\frac{2\pi(m-n)t}{T})dt + j\int_0^T \sin(\frac{2\pi(m-n)t}{T})dt = \begin{cases} T \text{ if } m = n \\ 0 \ \ m \neq n \end{cases} \tag{2.168}$$

We could redefine

$$\phi_n(t) = \frac{1}{\sqrt{T}}e^{j2n\pi t/T} = \frac{1}{\sqrt{T}}e^{j2n\pi f_0 t} \tag{2.169}$$

these eigenfunctions become orthonormal:

$$< \phi_m(t), \phi_n(t) >= \frac{1}{T}\int_0^T e^{j2\pi(m-n)t/T}dt = \delta[m-n] \tag{2.170}$$

This is actually Eq.1.27.

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:

$$x_T(t) = \sum_{n=-\infty}^{\infty} X_n\phi_n(t) = \sum_{n=-\infty}^{\infty} X_n e^{j2n\pi f_0 t} = \sum_{n=-\infty}^{\infty} X_n e^{jn\omega_0} \tag{2.171}$$

where X_n $(n = 0, \pm 1, \pm 2, \cdots)$ are the coefficients. This is the Fourier expansion (no longer in the generalized sense as before), to be discussed in detail in the next chapter.

So far we have only focused on periodic functions, but what about non-periodic functions? What kind of basis functions can be used to represent a non-periodic function? To address this question, we increase the period T, and note that at the limit $T \to \infty$ a periodic function $x_T(t)$ will become non-periodic. At the limit, the following also take place:

- The discrete variables $n\omega_0 = 2n\pi/T$ $(n = 0, \pm 1, \pm 2, \cdots)$ becomes a continuous variable $-\infty < \omega < \infty$;
- The gap between two consecutive eigenvalues becomes zero, i.e., $\triangle\lambda_n \to 0$, the discrete eigenvalues $\lambda_n = -(2n\pi/T)^2$ become a continuous eigenvalue function $\lambda = -\omega^2$;
- The frequency gap ω_0 between two consecutive eigenfunctions becomes zero, the discrete eigenfunctions $\phi_n(t) = e^{j2n\pi t/T}$ $(n = 0, \pm 1, \pm 2, \cdots)$ become a uncountable set of non-periodic eigenfunctions $\phi(t, f) = e^{j2\pi ft}$ 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 $\phi(t) = e^{j\omega t} = e^{j2\pi ft}$ form a complete orthogonal system that spans the function space of all non-periodic functions, i.e., each non-periodic function $x(t)$ can be represented as a linear combination of these basis functions:

$$x(t) = \int_{-\infty}^{\infty} X(f)\phi(t,f)df = \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df \tag{2.172}$$

The condition for this integral to converge is $X(f)$ is square integrable. This is the Fourier transform, to be discussed in detail in the next chapter.

2.2.4 Unitary Transformations

Definition 2.20. *If a linear transformation $T : V \to W$ conserves inner products:*

$$< T\boldsymbol{x}, T\boldsymbol{y} >=< \boldsymbol{x}, \boldsymbol{y} > \tag{2.173}$$

then it is called a unitary transformation. In particularly, if V is real with symmetric inner product $< \boldsymbol{x}, \boldsymbol{y} >=< \boldsymbol{y}, \boldsymbol{x} >$, then U is an orthogonal transformation.

Theorem 2.5. *A linear transformation T is unitary if and only if its adjoint T^* is equal to its inverse T^{-1}:*

$$T^* = T^{-1}, \qquad i.e. \qquad T^*T = TT^* = I \tag{2.174}$$

Proof: According to Eq.2.173, a unitary operator T satisfies:

$$< T\boldsymbol{x}, T\boldsymbol{y} >=< \boldsymbol{x}, \boldsymbol{y} > \tag{2.175}$$

If we let $T\boldsymbol{y} = \boldsymbol{z}$, i.e., $\boldsymbol{y} = T^{-1}\boldsymbol{z}$, then the above becomes:

$$< T\boldsymbol{x}, \boldsymbol{z} >=< \boldsymbol{x}, T^{-1}\boldsymbol{z} > \tag{2.176}$$

i.e., $T^{-1} = T^*$. On the other hand, given $T^* = T^{-1}$, we can immediately derive Eq.2.175 from Eq.2.176. This completes the proof. Because of this theorem, Eq.2.174 can also be used as an alternative definition of unitary operator.

Based on Eq.2.174 in the definition of a unitary transformation, we can immediately conclude that:

1. A unitary transformation preserves any measurements based on the inner product, such as the norm of a vector, the distance and angle between two vectors, and the projection of one on the other.
2. Parseval's identity holds for any unitary transformation $\boldsymbol{c} = U\boldsymbol{x}$:

$$||\boldsymbol{c}||^2 = ||U\boldsymbol{x}||^2 =< U\boldsymbol{x}, U\boldsymbol{x} >=< \boldsymbol{x}, \boldsymbol{x} >= ||\boldsymbol{x}||^2 \tag{2.177}$$

3. A unitary transformation can be interpreted as a rotation $R : V \to V$ in the vector space. [2]
4. A unitary transformation, a rotation, of an orthonormal basis $\{\boldsymbol{b}_n\}$ spanning a vector space is another orthonormal basis $\{U\boldsymbol{b}_n\}$ spanning the same space:

$$< U\boldsymbol{b}_m, U\boldsymbol{b}_n >=< \boldsymbol{b}_m, \boldsymbol{b}_n >= \delta[m-n] \tag{2.178}$$

A unitary transformation can be used for signal representation. Any given signal (either continuous or discrete) can be considered as a vector $\boldsymbol{x}$ in a proper vector space, represented under a certain basis. Then a unitary transformation U can be applied:

$$\begin{cases} \boldsymbol{c} = U^{-1}\boldsymbol{x} = U^*\boldsymbol{x} \\ \boldsymbol{x} = U\boldsymbol{c} \end{cases} \tag{2.179}$$

where the first equation is the forward transformation that maps the signal vector $\boldsymbol{x}$ to a coefficient vector $\boldsymbol{c}$, and the second equation is the inverse transformation that reconstructs the signal from the coefficients. Here $U^{-1} = U^*$ (Eq. 2.174) is the inverse transformation of U, and as $U\boldsymbol{c} = UU^{-1}\boldsymbol{x} = I\boldsymbol{x} = \boldsymbol{x}$, we see that $UU^{-1} = I$ is an identity operator.

The unitary transformation finds many applications in a wide variety of areas where a large quantity of signal/data needs to be processed, analyzed, and/or compressed. The motivation is to represent the signal in some suitable way so that all these operations can be carried out effectively and easily. Here we first high light some of the most general and fundamental ideas, which will be discussed in details in the following chapters for various specific methods.

- Either a continuous signal or a discrete signal is always given initially as a time function, which can be considered as a linear combination of a set of weighted and shifted impulses, in the form of a vector $\boldsymbol{x}$ represented by the standard basis of the vector space.
- The signal vector can be alternatively represented by any of the infinitely many orthogonal bases obtained by applying a specific rotation, a unitary transformation, to the standard basis.
- The signal vector is always represented as a set of coefficients of the basis being used, either the standard basis implicitly used before the unitary transformation, or some basis obtained by rotating the standard basis after the transformation.
- All of these different representations of the same signal are equivalent in the sense that the vector norm, representing the total amount of energy/information contained in the signal, is preserved by the unitary transformations, due to Parseval's identity.

[2] Strictly speaking, a unitary transformation may also correspond to other norm-preserving operations such as reflection and inversion, we here treat all such operations as rotations in the general sense.

- Depending on the specific signal processing task at hand, a proper transformation most suitable can be used.

2.2.5 Unitary Transformations in N-D Space

We consider specifically the unitary transformation in the N-D unitary space $\mathbb{C}^N$, where a linear transformation from vector $\boldsymbol{x}$ to another vector $\boldsymbol{y}$ is realized as a matrix multiplication $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$ by an N by N matrix $\boldsymbol{A}$. If we further assume this is linear transformation preserves inner products, then it is unitary and the corresponding matrix is called a unitary matrix.

Definition 2.21. *A matrix $\boldsymbol{U}$ is unitary if it preserves inner products:*

$$< \boldsymbol{U}\boldsymbol{x}, \boldsymbol{U}\boldsymbol{y} >=< \boldsymbol{x}, \boldsymbol{y} > \tag{2.180}$$

Theorem 2.6. *A matrix $\boldsymbol{U}$ is unitary if and only if $\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}$, i.e., the following two statements are equivalent:*

$$(a) \quad < \boldsymbol{U}\boldsymbol{x}, \boldsymbol{U}\boldsymbol{y} >=< \boldsymbol{x}, \boldsymbol{y} > \tag{2.181}$$

$$(b) \quad \boldsymbol{U}^*\boldsymbol{U} = \boldsymbol{U}\boldsymbol{U}^* = \boldsymbol{I}, \quad \textit{i.e.,} \quad \boldsymbol{U}^{-1} = \boldsymbol{U}^* \tag{2.182}$$

Proof: We first show if (b) then (a):

$$< \boldsymbol{U}\boldsymbol{x}, \boldsymbol{U}\boldsymbol{y} >= (\boldsymbol{U}\boldsymbol{x})^T\overline{\boldsymbol{U}\boldsymbol{y}} = \boldsymbol{x}^T\boldsymbol{U}^T\overline{\boldsymbol{U}}\,\overline{\boldsymbol{y}} = \boldsymbol{x}^T\boldsymbol{I}\overline{\boldsymbol{y}} =< \boldsymbol{x}, \boldsymbol{y} > \tag{2.183}$$

Next we show if (a) then (b):

$$< \boldsymbol{U}\boldsymbol{x}, \boldsymbol{U}\boldsymbol{x} >=< \boldsymbol{x}, \boldsymbol{x} >, \quad \text{i.e.,} \quad (\boldsymbol{U}\boldsymbol{x})^T\overline{\boldsymbol{U}}\boldsymbol{x} = \boldsymbol{x}^T\boldsymbol{U}^T\overline{\boldsymbol{U}}\,\overline{\boldsymbol{x}} = \boldsymbol{x}^T\overline{\boldsymbol{x}} \tag{2.184}$$

The second equation can be written as:

$$\boldsymbol{x}^*(\boldsymbol{U}^*\boldsymbol{U} - \boldsymbol{I})\boldsymbol{x} = 0 \tag{2.185}$$

Since $\boldsymbol{x}$ is an arbitrary vector, we must have $\boldsymbol{U}^*\boldsymbol{U} - \boldsymbol{I} = 0$, i.e., $\boldsymbol{U}^*\boldsymbol{U} = \boldsymbol{I}$. Post-multiplying this equation by $\boldsymbol{U}^{-1}$, we get $\boldsymbol{U}^* = \boldsymbol{U}^{-1}$. Premultiplying this new equation by $\boldsymbol{U}$, we get $\boldsymbol{U}\boldsymbol{U}^* = \boldsymbol{I}$. This completes the proof.

As (a) or (b) in the theorem are equivalent, either of them can be used as the definition of a unitary matrix. If a unitary matrix $\overline{\boldsymbol{U}} = \boldsymbol{U}$ is real, i.e., $\boldsymbol{U}^{-1} = \boldsymbol{U}^T$, then it is called an *orthogonal matrix*.

A unitary matrix $\boldsymbol{U}$ has the following properties:

- Unitary transformation $\boldsymbol{U}\boldsymbol{x}$ conserves vector norm, i.e., $||\boldsymbol{U}\boldsymbol{x}|| = ||\boldsymbol{x}||$ for any $\boldsymbol{x} \in \mathbb{C}^N$;
- All eigenvalues $\{\lambda_1, \cdots, \lambda_N\}$ of $\boldsymbol{U}$ have an absolute value of 1: $|\lambda_i| = 1$, i.e., they lie on the unit circle in the complex plain.
- The determinant of $\boldsymbol{U}$ has an absolute value of 1: $|det(\boldsymbol{U})| = 1$. This can be easily seen as $det(\boldsymbol{U}) = \prod_{n=1}^{N} \lambda_n$.

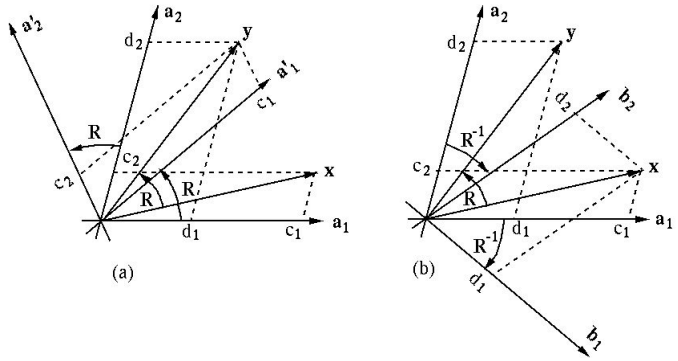

Figure 2.7 Rotation of vectors and bases

Previously in Eq.2.81 we showed that any $\boldsymbol{x} \in \mathbb{C}^N$ can be represented in terms of a set of orthonormal basis vectors that form a unitary matrix $\boldsymbol{U} = [\boldsymbol{u}_1, \cdots, \boldsymbol{u}_N]$:

$$\begin{cases} \boldsymbol{c} = \boldsymbol{U}^* \boldsymbol{x} \\ \boldsymbol{x} = \boldsymbol{U}\boldsymbol{c} \end{cases} \tag{2.186}$$

We realize this is actually the special case of the unitary transformation in Eq.2.179. Specifically here a discrete signal vector $\boldsymbol{x} = [x_1, \cdots, x_N]^T$, originally given under the implicit standard basis $\boldsymbol{I} = [\boldsymbol{e}_1, \cdots, \boldsymbol{e}_N]^T$ is converted to another vector $\boldsymbol{c} = [c_1, \cdots, c_N]^T$ representing the coefficients of a new basis $\boldsymbol{U} = [\boldsymbol{u}_1, \cdots, \boldsymbol{u}_N]^T$.

Also note that the property $||\boldsymbol{x}||^2 = ||\boldsymbol{U}\boldsymbol{x}||^2 = ||\boldsymbol{c}||^2$ is actually Parseval's identity in $\mathbb{C}^N$, indicating that a unitary transformation $\boldsymbol{U}^*\boldsymbol{x} = \boldsymbol{c}$ or $\boldsymbol{U}\boldsymbol{c} = \boldsymbol{x}$ always conserves the vector norm, which representing the signal energy or information. This unitary transformation can be considered as a rotation of the standard basis $\boldsymbol{I}$ to become another basis $\boldsymbol{U}$.

A unitary transformation can be considered as a rotation in the N-D unitary space $\mathbb{C}^N$, by which the vector norm is obviously conserved. Let $\{\boldsymbol{a}_n\}$ be a basis (not necessarily orthogonal) of the space, then any vector $\boldsymbol{x}$ can be represented in terms of a set of coefficients $\{c_n\}$:

$$\boldsymbol{x} = \sum_{n=1}^{N} c_n \boldsymbol{a}_n \tag{2.187}$$

Rotating this vector by a unitary matrix $\boldsymbol{U}$, we get a new vector:

$$\boldsymbol{y} = \boldsymbol{U}\boldsymbol{x} = \boldsymbol{U}\left[\sum_{n=1}^{N} c_n \boldsymbol{a}_n\right] = \sum_{n=1}^{N} c_n \boldsymbol{U}\boldsymbol{a}_n = \sum_{n=1}^{N} c_n \boldsymbol{a}'_n \tag{2.188}$$

This equation indicates that the rotated vector $\boldsymbol{y}$ can still be represented by the same set of coefficients $\{c_n\}$, if the basis $\{\boldsymbol{a}_n\}$ is also rotated the same way to become $\boldsymbol{a}'_n = \boldsymbol{U}\boldsymbol{a}_n$, as illustrated in Fig.2.7(a).

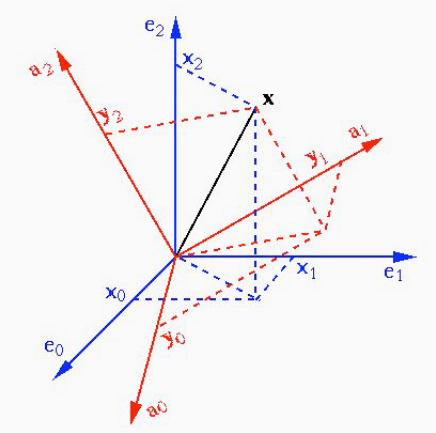

Figure 2.8 Rotation of coordinate system

Under the original basis $\{\boldsymbol{a}_n\}$, the rotated vector $\boldsymbol{y}$ can be represented in terms of a set of new coefficients $\{d_n\}$:

$$\boldsymbol{y} = \sum_{n=1}^{N} d_n \boldsymbol{a}_n = [\boldsymbol{a}_1, \cdots, \boldsymbol{a}_N] \begin{bmatrix} d_1 \\ \vdots \\ d_N \end{bmatrix} \tag{2.189}$$

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 $\boldsymbol{y}$ in the opposite direction by the inverse matrix $\boldsymbol{U}^{-1} = \boldsymbol{U}^*$, of course we get $\boldsymbol{x}$ back:

$$\boldsymbol{U}^{-1}\boldsymbol{y} = \boldsymbol{U}^{-1}\left[\sum_{n=1}^{N} d_n \boldsymbol{a}_n\right] = \sum_{n=1}^{N} d_n \boldsymbol{U}^{-1}\boldsymbol{a}_n = \sum_{n=1}^{N} d_n \boldsymbol{b}_n \tag{2.190}$$

where $\boldsymbol{b}_m = \boldsymbol{U}^{-1}\boldsymbol{a}_m = \boldsymbol{U}^*\boldsymbol{a}_m$ is a new basis obtained by rotating $\{\boldsymbol{a}_m\}$ in the opposite direction. Now we have:

$$P_{\boldsymbol{a}_n}(\boldsymbol{y}) = <\boldsymbol{y}, \boldsymbol{a}_n> = <\boldsymbol{U}\boldsymbol{x}, \boldsymbol{a}_n> = <\boldsymbol{x}, \boldsymbol{U}^*\boldsymbol{a}_n> = <\boldsymbol{x}, \boldsymbol{b}_n> = P_{\boldsymbol{b}_n}(\boldsymbol{x}) \tag{2.191}$$

We see that the projection of the new vector $\boldsymbol{y} = \boldsymbol{U}\boldsymbol{x}$ onto the old basis $\boldsymbol{a}_n$ is the same as the projection of old vector $\boldsymbol{x}$ onto the new basis $\boldsymbol{b}_n = \boldsymbol{U}^{-1}\boldsymbol{a}$. 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 an $N = 3$ dimensional space is illustrated in Fig.2.8.

Example 2.6: Consider two orthonormal basis vectors that span a 2-D space:

$$\boldsymbol{a}_1 = \frac{1}{2}\begin{bmatrix} \sqrt{3} \\ 1 \end{bmatrix}, \quad \boldsymbol{a}_2 = \frac{1}{2}\begin{bmatrix} -1 \\ \sqrt{3} \end{bmatrix} \tag{2.192}$$

Under this basis a given vector $\boldsymbol{x} = [1,\ 2]^T$ can be expressed as:

$$\boldsymbol{x} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} = c_1\boldsymbol{a}_1 + c_2\boldsymbol{a}_2 = \frac{c_1}{2}\begin{bmatrix} \sqrt{3} \\ 1 \end{bmatrix} + \frac{c_2}{2}\begin{bmatrix} -1 \\ \sqrt{3} \end{bmatrix} \tag{2.193}$$

where the two coefficients c_1 and c_2 can be obtained as the projection of $\boldsymbol{x}$ onto $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$, respectively:

$$\begin{aligned} c_1 &= <\boldsymbol{x}, \boldsymbol{a}_1> = \boldsymbol{x}^T\boldsymbol{a}_1 = \frac{1}{2}[1,\ 2]\begin{bmatrix} \sqrt{3} \\ 1 \end{bmatrix} = \frac{\sqrt{3}+2}{2} \\ c_2 &= <\boldsymbol{x}, \boldsymbol{a}_2> = \boldsymbol{x}^T\boldsymbol{a}_2 = \frac{1}{2}[1,\ 2]\begin{bmatrix} -1 \\ \sqrt{3} \end{bmatrix} = \frac{2\sqrt{3}-1}{2} \end{aligned} \tag{2.194}$$

A counter clockwise rotation of $\theta = 30^\circ$ is represented by a matrix:

$$\boldsymbol{R} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} = \frac{1}{2}\begin{bmatrix} \sqrt{3} & -1 \\ 1 & \sqrt{3} \end{bmatrix} \tag{2.195}$$

Pre-multiplied by this matrix, $\boldsymbol{x}$ will be rotated to become:

$$\boldsymbol{y} = \boldsymbol{R}\boldsymbol{x} = \frac{1}{2}\begin{bmatrix} \sqrt{3} & -1 \\ 1 & \sqrt{3} \end{bmatrix}\begin{bmatrix} 1 \\ 2 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} \sqrt{3}-2 \\ 2\sqrt{3}+1 \end{bmatrix} \tag{2.196}$$

This rotated vector $\boldsymbol{y}$ can be represented under the same basis $\{\boldsymbol{a}_1, \boldsymbol{a}_2\}$ by two new coefficients:

$$\begin{aligned} d_1 &= <\boldsymbol{y}, \boldsymbol{a}_1> = \frac{1}{4}[\sqrt{3}-2,\ 2\sqrt{3}+1]\begin{bmatrix} \sqrt{3} \\ 1 \end{bmatrix} = 1 \\ d_2 &= <\boldsymbol{y}, \boldsymbol{a}_2> = \frac{1}{4}[\sqrt{3}-2,\ 2\sqrt{3}+1]\begin{bmatrix} -1 \\ \sqrt{3} \end{bmatrix} = 2 \end{aligned} \tag{2.197}$$

On the other hand, the basis $\{\boldsymbol{a}_1, \boldsymbol{a}_2\}$ can be rotated in the opposite direction $-\theta = -30^\circ$ represented by:

$$\boldsymbol{R}^{-1} = \boldsymbol{R}^T = \frac{1}{2}\begin{bmatrix} \sqrt{3} & 1 \\ -1 & \sqrt{3} \end{bmatrix} \tag{2.198}$$

to become:

$$\begin{aligned} \boldsymbol{b}_1 &= \boldsymbol{R}^T\boldsymbol{a}_1 = \frac{1}{4}\begin{bmatrix} \sqrt{3} & 1 \\ -1 & \sqrt{3} \end{bmatrix}\begin{bmatrix} \sqrt{3} \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \boldsymbol{e}_1 \\ \boldsymbol{b}_2 &= \boldsymbol{R}^T\boldsymbol{a}_2 = \frac{1}{4}\begin{bmatrix} \sqrt{3} & 1 \\ -1 & \sqrt{3} \end{bmatrix}\begin{bmatrix} -1 \\ \sqrt{3} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \boldsymbol{e}_2 \end{aligned} \tag{2.199}$$

Under this new basis $\{\boldsymbol{b}_1, \boldsymbol{b}_2\}$ (which turns out to be the standard basis), the vector $\boldsymbol{x}$ is expressed as:

$$\boldsymbol{x} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} = d_1'\boldsymbol{b}_1 + d_2'\boldsymbol{b}_2 = 1\boldsymbol{b}_1 + 2\boldsymbol{b}_2 \tag{2.200}$$

We see that $d_1' = d_1 = 1$ and $d_2' = d_2 = 2$, in other words, the representation $\{d_1, d_2\}$ of the rotated vector $\boldsymbol{y}$ under the original basis $\{\boldsymbol{a}_1, \boldsymbol{a}_2\}$ is equivalent to

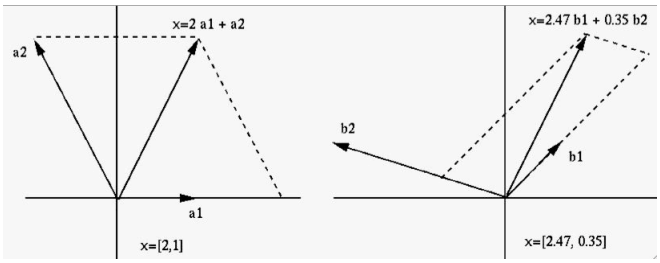

Figure 2.9 Rotation of a basis

the representation $\{d_1', d_2'\}$ of the original vector $\boldsymbol{x}$ under the inversely rotated basis $\{\boldsymbol{b}_1, \boldsymbol{b}_2\}$.

Example 2.7: In Example 2.2, a vector $\boldsymbol{x} = [1,\ 2]^T = 1\boldsymbol{e}_1 + 2\boldsymbol{e}_2$ is represented under a different basis $\boldsymbol{a}_1 = [1, 0]^T$ and $\boldsymbol{a}_2 = [-1, 2]^T$ as

$$\boldsymbol{x} = 1\boldsymbol{a}_1 + 2\boldsymbol{a}_2 = 2\begin{bmatrix}1\\0\end{bmatrix} + 1\begin{bmatrix}-1\\2\end{bmatrix} = \begin{bmatrix}1\\2\end{bmatrix} \tag{2.201}$$

This basis $\{\boldsymbol{a}_1, \boldsymbol{a}_2\}$ can be rotated by $\theta = 45^\circ$ with the rotation matrix:

$$\boldsymbol{R} = \begin{bmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{bmatrix} = 0.707\begin{bmatrix}1 & -1\\ 1 & 1\end{bmatrix} \tag{2.202}$$

to become a new basis $\{\boldsymbol{b}_1, \boldsymbol{b}_2\}$:

$$\boldsymbol{b}_1 = \boldsymbol{R}\boldsymbol{a}_1 = \boldsymbol{R}\begin{bmatrix}1\\0\end{bmatrix} = 0.707\begin{bmatrix}1\\1\end{bmatrix}, \quad \boldsymbol{b}_2 = \boldsymbol{R}\boldsymbol{a}_2 = \boldsymbol{R}\begin{bmatrix}-1\\2\end{bmatrix} = 0.707\begin{bmatrix}-3\\1\end{bmatrix} \tag{2.203}$$

Under this new basis, $\boldsymbol{x}$ can be represented as:

$$\boldsymbol{x} = c_1'\boldsymbol{b}_1 + c_2'\boldsymbol{b}_2 = c_1'\ 0.707\begin{bmatrix}1\\1\end{bmatrix} + c_2'\ 0.707\begin{bmatrix}-3\\1\end{bmatrix} = 0.707\begin{bmatrix}1 & -3\\1 & 1\end{bmatrix}\begin{bmatrix}c_1'\\c_2'\end{bmatrix} = \begin{bmatrix}1\\2\end{bmatrix} \tag{2.204}$$

Solving this we get $c_1' = 2.47$ and $c_2' = 0.35$, i.e., $\boldsymbol{x} = 2.47\boldsymbol{b}_1 + 0.35\boldsymbol{b}_2$. (Note that in this case the coefficients c_1' and c_2' cannot be found as the projections of $\boldsymbol{x}$ onto the basis vectors, as they are not orthonormal.) To conclude, we see that the same vector $\boldsymbol{x}$ can be equivalently represented by different bases:

$$\boldsymbol{x} = 1\boldsymbol{e}_1 + 2\boldsymbol{e}_2 = 2\boldsymbol{a}_1 + 1\boldsymbol{a}_2 = 2.47\boldsymbol{b}_1 + 0.35\boldsymbol{b}_2 \tag{2.205}$$

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 all basis vectors are necessary to represent any given vector in a vector space, it is still possible to approximate the vector in a subspace if error is allowed. Also, a continuous function may not be accurately representable in a finite dimensional space, but it may still be desirable to approximate the function in such a space. The issue in such approximation is how to minimize the error.

Let V be an N-D Hilbert space, and $U \subset V$ be an M-D subspace spanned by a set of M N-D basis vectors $\{\boldsymbol{a}_1, \cdots, \boldsymbol{a}_M\}$ (not necessarily orthogonal), and assume a given vector $\boldsymbol{x} \in V$ is approximated by a vector $\hat{\boldsymbol{x}} \in U$:

$$\hat{\boldsymbol{x}} = \sum_{n=1}^{M} c_n \boldsymbol{a}_n \tag{2.206}$$

An error vector is defined as

$$\tilde{\boldsymbol{x}} = \boldsymbol{x} - \hat{\boldsymbol{x}} = \boldsymbol{x} - \sum_{n=1}^{M} c_n \boldsymbol{a}_n \tag{2.207}$$

The least squares error of the approximation is defined as:

$$\varepsilon = ||\tilde{\boldsymbol{x}}||^2 = < \tilde{\boldsymbol{x}}, \tilde{\boldsymbol{x}} > \tag{2.208}$$

The goal is to find a set of coefficients $\{c_1, \cdots, c_M\}$ so that the error ε is minimized. The following *projection theorem* will address this issue.

Theorem 2.7. *(The projection theorem) The least squares error $\varepsilon = ||\tilde{\boldsymbol{x}}||^2$ of the approximation by equation 2.206 is minimized if and only if the error vector $\tilde{\boldsymbol{x}} = \boldsymbol{x} - \hat{\boldsymbol{x}}$ is orthogonal to the subspace U:*

$$\tilde{\boldsymbol{x}} \perp U, \quad \textit{i.e.,} \quad \tilde{\boldsymbol{x}} \perp \boldsymbol{a}_n, \ (n = 1, \cdots, M) \tag{2.209}$$

Proof: Let $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{x}}'$ be two vectors both in the subspace U, where $\hat{\boldsymbol{x}}'$ is arbitrary but $\hat{\boldsymbol{x}}$ is the projection of $\boldsymbol{x}$ onto U, i.e., $(\boldsymbol{x} - \hat{\boldsymbol{x}}) \perp U$. As $\hat{\boldsymbol{x}} - \hat{\boldsymbol{x}}'$ is also a vector in U, we have $(\boldsymbol{x} - \hat{\boldsymbol{x}}) \perp (\hat{\boldsymbol{x}} - \hat{\boldsymbol{x}}')$, i.e., $< \boldsymbol{x} - \hat{\boldsymbol{x}}, \ \hat{\boldsymbol{x}} - \hat{\boldsymbol{x}}' >= 0$. Now consider the approximation error associated with $\hat{\boldsymbol{x}}'$:

$$\begin{aligned} &||\boldsymbol{x} - \hat{\boldsymbol{x}}'||^2 = ||\boldsymbol{x} - \hat{\boldsymbol{x}} + \hat{\boldsymbol{x}} - \hat{\boldsymbol{x}}'||^2 \\ &= ||\boldsymbol{x} - \hat{\boldsymbol{x}}||^2 + < \boldsymbol{x} - \hat{\boldsymbol{x}}, \ \hat{\boldsymbol{x}} - \hat{\boldsymbol{x}}' > + < \hat{\boldsymbol{x}} - \hat{\boldsymbol{x}}', \ \boldsymbol{x} - \hat{\boldsymbol{x}} > + ||\hat{\boldsymbol{x}} - \hat{\boldsymbol{x}}'||^2 \\ &= ||\boldsymbol{x} - \hat{\boldsymbol{x}}||^2 + ||\hat{\boldsymbol{x}} - \hat{\boldsymbol{x}}'||^2 \end{aligned} \tag{2.210}$$

We see that the error $||\boldsymbol{x} - \hat{\boldsymbol{x}}||^2$ associated with $\hat{\boldsymbol{x}}'$ is always greater than the error $||\boldsymbol{x} - \hat{\boldsymbol{x}}||^2$ associated with $\hat{\boldsymbol{x}}$, unless $\hat{\boldsymbol{x}}' = \hat{\boldsymbol{x}}$. In other words, the error is

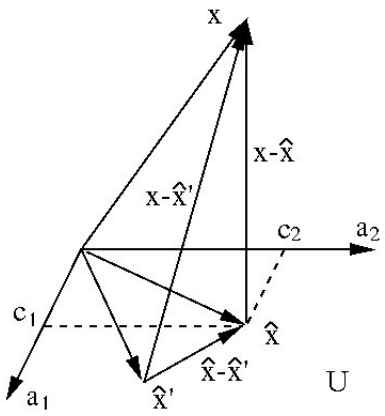

Figure 2.10 Projection theorem

minimized if and only if the approximation is $\hat{\boldsymbol{x}}$, the projection of $\boldsymbol{x}$ onto the subspace U. This completes the proof.

This theorem can be understood intuitively as shown in Fig.2.10, where a vector $\boldsymbol{x}$ in a 3-D space is approximated by a vector $\hat{\boldsymbol{x}}$ in a 2-D subspace $\hat{\boldsymbol{x}} = c_1\boldsymbol{a}_1 + c_2\boldsymbol{a}_2$. The error vector $\varepsilon = ||\boldsymbol{x} - \hat{\boldsymbol{x}}||^2$ is indeed minimum if $\boldsymbol{x} - \hat{\boldsymbol{x}}$ is orthogonal to the 2-D plane spanned by the basis vectors $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$, as any other vector $\hat{\boldsymbol{x}}'$ in this plane would be associated with a larger error. In other words, the optimal approximation $\hat{\boldsymbol{x}}$ is the *projection* of $\boldsymbol{x}$ onto the subspace U. This theorem can be generalized to any inner product space.

The coefficients c_n corresponding to the optimal approximation can be found based on the projection theorem, which states that the minimum error vector $\tilde{\boldsymbol{x}}$ has to be orthogonal to each of the basis vectors that span the subspace U:

$$\begin{aligned} &< \tilde{\boldsymbol{x}}, \boldsymbol{a}_m > = < \boldsymbol{x} - \sum_{n=1}^{M} c_n \boldsymbol{a}_n,\ \boldsymbol{a}_m > \\ &= < \boldsymbol{x}, \boldsymbol{a}_m > - \sum_{n=1}^{M} c_n < \boldsymbol{a}_n, \boldsymbol{a}_m > = 0, \qquad (m = 1, \cdots, M) \end{aligned} \tag{2.211}$$

i.e.

$$< \boldsymbol{x}, \boldsymbol{a}_m > = \sum_{n=1}^{M} c_n < \boldsymbol{a}_n, \boldsymbol{a}_m >, \qquad (m = 1, \cdots, M) \tag{2.212}$$

These equations can be rewritten in matrix form:

$$\begin{bmatrix} < \boldsymbol{x}, \boldsymbol{a}_1 > \\ \vdots \\ < \boldsymbol{x}, \boldsymbol{a}_M > \end{bmatrix}_{M\times 1} = \begin{bmatrix} < \boldsymbol{a}_1, \boldsymbol{a}_1 > & \cdots & < \boldsymbol{a}_M, \boldsymbol{a}_1 > \\ \vdots & \ddots & \vdots \\ < \boldsymbol{a}_1, \boldsymbol{a}_M > & \cdots & < \boldsymbol{a}_M, \boldsymbol{a}_M > \end{bmatrix}_{M\times M} \begin{bmatrix} c_1 \\ \vdots \\ c_m \end{bmatrix}_{M\times 1} \tag{2.213}$$

Solving this system of M equations and M unknowns, we get the optimal coefficients. The computational complexity for solving this linear system is $O(M^3)$.

In particular, if the basis vectors are orthogonal $< \boldsymbol{a}_m, \boldsymbol{a}_n >= 0$ for all $m \neq n$, then all off-diagonal components of the M by M matrix in the equation above are zero, and each of the coefficients can be obtained independently:

$$c_n = \frac{< \boldsymbol{x}, \boldsymbol{a}_n >}{< \boldsymbol{a}_n, \boldsymbol{a}_n >} = \frac{< \boldsymbol{x}, \boldsymbol{a}_n >}{||\boldsymbol{a}_n||^2}, \qquad (n = 1, \cdots, M) \tag{2.214}$$

Now the complexity for finding the m coefficients is $O(M^2)$, and the vector $\boldsymbol{x}$ can approximated as:

$$\hat{\boldsymbol{x}} = \sum_{n=1}^{M} c_n \boldsymbol{a}_n = \sum_{n=1}^{M} \frac{< \boldsymbol{x}, \boldsymbol{a}_n >}{||\boldsymbol{a}_n||^2} \boldsymbol{a}_n = \sum_{n=1}^{M} \boldsymbol{p}_{\boldsymbol{a}_n}(\boldsymbol{x}) \tag{2.215}$$

We see that $\hat{\boldsymbol{x}}$ is the vector sum of the projections of $\boldsymbol{x}$ onto each of the basis vectors $\boldsymbol{a}_n$ $(n = 1, \cdots, M)$ of the subspace U. Moreover, if all basis vectors are are orthonormal $< \boldsymbol{a}_m, \boldsymbol{a}_n >= \delta[m - n]$, i.e., $||\boldsymbol{a}_n||^2 = 1$, then the coefficients become:

$$c_n =< \boldsymbol{x}, \boldsymbol{a}_n >, \qquad (n = 1, \cdots, M) \tag{2.216}$$

and the approximation becomes:

$$\hat{\boldsymbol{x}} = \sum_{n=1}^{M} c_n \boldsymbol{a}_n = \sum_{n=1}^{M} < \boldsymbol{x}, \boldsymbol{a}_n > \boldsymbol{a}_n \tag{2.217}$$

The results above for a finite dimensional space can be generalized to an infinite dimensional Hilbert space H, where a vector $\boldsymbol{x} \in H$ can be approximated in a finite M-D subspace:

$$\hat{\boldsymbol{x}}_M = \sum_{n=1}^{M} c_n \boldsymbol{b}_n$$

where $\boldsymbol{b}_n$ is a set of orthogonal basis vectors. We want to find the coefficients c_n corresponding to the minimum error. According to the projection theorem, the approximation error is minimized if the error vector is orthogonal to the M-D subspace spanned by $\boldsymbol{b}_m$ $(m = 1, \cdots, M)$:

$$(\boldsymbol{x} - \hat{\boldsymbol{x}}) \perp b_m, \qquad (m = 1, \cdots, M)$$

i.e.,

$$< (\boldsymbol{x} - \hat{\boldsymbol{x}}), \boldsymbol{b}_m >=< (\boldsymbol{x} - \sum_{n=1}^{M} c_n \boldsymbol{b}_n), \boldsymbol{b}_m >=< \boldsymbol{x}, \boldsymbol{b}_m > - \sum_{n=1}^{M} c_n < \boldsymbol{b}_n, \boldsymbol{b}_m >$$

$$= < \boldsymbol{x}, \boldsymbol{b}_m > - \sum_{n=1}^{M} c_n \delta[m - n] =< \boldsymbol{x}, \boldsymbol{b}_m > - c_m = 0$$

We see that the coefficients c_n corresponding to minimum error can be obtained as the projection of $\boldsymbol{x}$ onto the basis vectors $\boldsymbol{b}_n$:

$$c_n =< \boldsymbol{x}, \boldsymbol{b}_n >, \qquad (n = 1, \cdots, M) \tag{2.218}$$

Now the approximation error becomes:

$$\begin{aligned} ||\boldsymbol{x} - \hat{\boldsymbol{x}}_M||^2 &=< \boldsymbol{x}, \boldsymbol{x} > - < \boldsymbol{x}, \hat{\boldsymbol{x}}_M > - < \hat{\boldsymbol{x}}_M, \boldsymbol{x} > + < \hat{\boldsymbol{x}}_M, \hat{\boldsymbol{x}}_M > \\ &= ||\boldsymbol{x}||^2 - \sum_{n=1}^{M} < \boldsymbol{x}, \boldsymbol{b}_n > \overline{c}_n - \sum_{n=1}^{M} c_n < \boldsymbol{b}_n, \boldsymbol{x} > + \sum_{n=1}^{M} |c_n|^2 \\ &= ||\boldsymbol{x}||^2 - \sum_{n=1}^{M} |c_n|^2 \geq 0 \end{aligned}$$

In a Hilbert space, the sequence $\boldsymbol{x}_M$ converges when $M \to \infty$:

$$\lim_{M\to\infty} \hat{\boldsymbol{x}}_M = \lim_{M\to\infty} \sum_{n=1}^{M} c_n \boldsymbol{b}_n = \sum_{n=1}^{\infty} c_n \boldsymbol{b}_n = \boldsymbol{x} \tag{2.219}$$

i.e.,

$$\lim_{m\to\infty} ||\boldsymbol{x} - \hat{\boldsymbol{x}}_M||^2 = ||\boldsymbol{x} - \sum_{n=1}^{\infty} c_n \boldsymbol{b}_n||^2 = 0 \tag{2.220}$$

This is Parseval's equality:

$$||\boldsymbol{x}||^2 = \sum_{n=1}^{\infty} |c_n|^2 \tag{2.221}$$

Consider specifically a unitary space $\mathbb{C}^N$ spanned by a basis $\{\boldsymbol{a}_1, \cdots, \boldsymbol{a}_N\}$ (not necessarily orthogonal). We want to express a given vector $\boldsymbol{x} = [x_1, \cdots, x_N]^T$ in an M-D subspace spanned by the first M basis vectors $\boldsymbol{a}_n$, $n = 1, \cdots, M$ as:

$$\boldsymbol{x} = \sum_{n=1}^{M} c_n \boldsymbol{a}_n = [\boldsymbol{a}_1, \cdots, \boldsymbol{a}_M]_{N\times M} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_M \end{bmatrix}_{M\times 1} = \boldsymbol{A}\boldsymbol{c} \tag{2.222}$$

This equation system is over-determined with only M unknowns $\{c_1, \cdots, c_M\}$ but N equations, i.e., $\boldsymbol{A}$ is an N by M non-square matrix and is non-invertible, therefore the system has no solution in general, indicating the impossibility of representing an N-D vector in an M-D subspace. However, based on the projection theorem, we can still find an approximate solution by solving Eq.2.213. In this unitary space, the inner products in the equation become dot products $< \boldsymbol{x}, \boldsymbol{a}_n >= \boldsymbol{a}_n^* \boldsymbol{x}$ and $< \boldsymbol{a}_m, \boldsymbol{a}_n >= \boldsymbol{a}_n^* \boldsymbol{a}_m$, and Eq. 2.213 can be written as:

$$\boldsymbol{A}^* \boldsymbol{x} = \boldsymbol{A}^* \boldsymbol{A} \boldsymbol{c} \tag{2.223}$$

where $\boldsymbol{A}^* \boldsymbol{A}$ is an M by M square matrix and therefore invertible. Premultiplying its inverse $(\boldsymbol{A}^T \boldsymbol{A})^{-1}$ on both sides, we can solve the over-determined equation

system to obtain the optimal coefficient corresponding to the minimum least square error:

$$\boldsymbol{c} = (\boldsymbol{A}^*\boldsymbol{A})^{-1}\boldsymbol{A}^*\boldsymbol{x} = \boldsymbol{A}^-\boldsymbol{x} \tag{2.224}$$

where

$$\boldsymbol{A}^- = (\boldsymbol{A}^*\boldsymbol{A})^{-1}\boldsymbol{A}^* \tag{2.225}$$

is an M by N matrix, known as the *generalized inverse or pseudo-inverse* of the N by M matrix $\boldsymbol{A}$, and we have: $\boldsymbol{A}^-\boldsymbol{A} = \boldsymbol{I}$. The pseudo-inverse in Eq.2.225 is for the case where $\boldsymbol{A}$ has more columns than rows ($M < N$ in this case). If $\boldsymbol{A}$ has more rows than columns ($M > N$ in this case), the pseudo-inverse becomes:

$$\boldsymbol{A}^- = \boldsymbol{A}^*(\boldsymbol{A}\boldsymbol{A}^*)^{-1} \tag{2.226}$$

If all N dimensions basis vectors can be used, then $\boldsymbol{A}$ becomes an N by N square matrix and the pseudo-inverse becomes the regular inverse:

$$\boldsymbol{A}^- = \boldsymbol{A}^{-1}(\boldsymbol{A}^*)^{-1}\boldsymbol{A}^* = \boldsymbol{A}^{-1} \tag{2.227}$$

and the coefficients can be found simply by:

$$\boldsymbol{c} = \boldsymbol{A}^{-1}\boldsymbol{x} \tag{2.228}$$

Example 2.8: Consider a 3-D Euclidean space $\mathbb{R}^3$ spanned by a set of three linearly independent vectors:

$$\boldsymbol{a}_1 = \begin{bmatrix}1\\0\\0\end{bmatrix}, \quad \boldsymbol{a}_2 = \begin{bmatrix}1\\1\\0\end{bmatrix}, \quad \boldsymbol{a}_3 = \begin{bmatrix}1\\1\\1\end{bmatrix}$$

We want to find two coefficients c_1 and c_2 so that a given vector $\boldsymbol{x} = [1, 2, 3]^T$ can be optimally approximated as $\hat{\boldsymbol{x}} = c_1\boldsymbol{a}_1 + c_2\boldsymbol{a}_2$ in the 2-D subspace spanned by $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$. First we construct a matrix composed of $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$:

$$\boldsymbol{A} = [\boldsymbol{a}_1, \boldsymbol{a}_2] = \begin{bmatrix}1 & 1\\0 & 1\\0 & 0\end{bmatrix}$$

Next we find the pseudo inverse of $\boldsymbol{A}$:

$$\boldsymbol{A}^- = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T = \begin{bmatrix}1 & -1 & 0\\0 & 1 & 0\end{bmatrix}$$

The two coefficients can then be obtained as:

$$\boldsymbol{c} = \begin{bmatrix}c_1\\c_2\end{bmatrix} = \boldsymbol{A}^-\boldsymbol{x} = \begin{bmatrix}1 & -1 & 0\\0 & 1 & 0\end{bmatrix}\begin{bmatrix}1\\2\\3\end{bmatrix} = \begin{bmatrix}-1\\2\end{bmatrix}$$

The optimal approximation is therefore

$$\hat{\boldsymbol{x}} = c_1\boldsymbol{a}_1 + c_2\boldsymbol{a}_2 = -1\begin{bmatrix}1\\0\\0\end{bmatrix} + 2\begin{bmatrix}1\\1\\0\end{bmatrix} = \begin{bmatrix}1\\2\\0\end{bmatrix}$$

which is indeed the projection of $\boldsymbol{x} = [1, 2, 3]^T$ onto the 2-D subspace spanned by $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$.

Alternatively if we want to approximate $\boldsymbol{x}$ by $\boldsymbol{a}_2$ and $\boldsymbol{a}_3$ as $\hat{\boldsymbol{x}} = c_2\boldsymbol{a}_2 + c_3\boldsymbol{a}_3$, we have:

$$\boldsymbol{A} = [\boldsymbol{a}_2, \boldsymbol{a}_3] = \begin{bmatrix}1&1\\1&1\\0&1\end{bmatrix}, \qquad \boldsymbol{A}^- = \frac{1}{2}\begin{bmatrix}1&1&-2\\0&0&2\end{bmatrix}$$

and

$$\boldsymbol{c} = \boldsymbol{A}^-\boldsymbol{x} = \begin{bmatrix}-1.5\\3\end{bmatrix}, \qquad \hat{\boldsymbol{x}} = c_2\boldsymbol{a}_2 + c_3\boldsymbol{a}_3 = -1.5\begin{bmatrix}1\\1\\0\end{bmatrix} + 3\begin{bmatrix}1\\1\\1\end{bmatrix} = \begin{bmatrix}1.5\\1.5\\3\end{bmatrix}$$

If all three basis vectors can be used, then the coefficients can be found as:

$$\boldsymbol{c} = \boldsymbol{A}^{-1}\boldsymbol{x} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \boldsymbol{a}_3]^{-1}\boldsymbol{x} = \begin{bmatrix}1&-1&0\\0&1&-1\\0&0&1\end{bmatrix}\begin{bmatrix}1\\2\\3\end{bmatrix} = \begin{bmatrix}-1\\-1\\3\end{bmatrix}$$

and $\boldsymbol{x}$ can be precisely represented as:

$$\boldsymbol{x} = c_1\boldsymbol{a}_1 + c_2\boldsymbol{a}_2 + c_3\boldsymbol{a}_3 = \boldsymbol{A}\boldsymbol{c} = \begin{bmatrix}1\\2\\3\end{bmatrix}$$

2.3.2 Signal Approximation

As shown before, a signal, either continuous or discrete, can be considered as a vector in an inner product space, and represented by a set of coefficients with respect to a specific basis that spans the space. As the space can also be spanned by different bases, the signal can be equivalently represented by different sets of coefficients each for one particular basis. Moreover, in an N-D vector space, any two orthonormal bases are related by a rotation, and correspondingly the two sets of coefficients for the same signal vector are related by an unitary transformation representing the rotation between the two bases.

Although different bases are equivalent in terms of representing the entire signal, they may differ drastically in term of what aspect of the signal each of the coefficients represents. 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 2.9: Given a signal $x(t) = t$ defined over $0 \le t < 2$ (undefined outside the range), we want to optimally approximate it in a subspace spanned by two basis functions $e_1(t)$ and $e_2(t)$:

$$\hat{x}(t) = c_1 e_1(t) + c_2 e_2(t)$$

where $e_1(t)$ and $e_2(t)$ are defined as:

$$e_1(t) = \begin{cases} 1, 0 \le t < 1 \\ 0, 1 \le t < 2 \end{cases}, \qquad e_2(t) = \begin{cases} 0, 0 \le t < 1 \\ 1, 1 \le t < 2 \end{cases}$$

These two basis functions are obviously orthonormal:

$$< e_1(t), e_2(t) >= \int_0^2 e_i(t) e_j(t) dt = \delta[i-j]$$

Following the projection theorem, the coefficients c_1 and c_2 can be found by solving these two simultaneous equations:

$$c_1 \int_0^2 e_1(t) e_1(t) dt + c_2 \int_0^2 e_2(t) e_1(t) dt = \int_0^2 x(t) e_1(t) dt$$
$$c_1 \int_0^2 e_1(t) e_2(t) dt + c_2 \int_0^2 e_2(t) e_2(t) dt = \int_0^2 x(t) e_2(t) dt$$

As $e_1(t)$ and $e_2(t)$ are orthonormal, the above becomes

$$c_1 = \int_0^2 x(t) e_1(t) dt = \int_0^1 t \, dt = 0.5, \quad c_2 = \int_0^2 x(t) e_2(t) dt = \int_1^2 t \, dt = 1.5 \tag{2.229}$$

i.e., the two coefficients c_1 and c_2 can be obtained independently as the projections of $x(t)$ onto each of the basis functions. Now the signal can be approximated as:

$$\hat{x}(t) = 0.5 e_1(t) + 1.5 e_2(t) = \begin{cases} 0.5, 0 \le t < 1 \\ 1.5, 1 \le t < 2 \end{cases} \tag{2.230}$$

Next, consider approximating this signal $x(t)$ by two different basis functions $u_1(t)$ and $u_2(t)$ that span the same subspace:

$$\hat{x}(t) = d_1 u_1(t) + d_2 u_2(t) \tag{2.231}$$

where

$$u_1(t) = \frac{1}{\sqrt{2}} = \frac{1}{\sqrt{2}}[e_1(t) + e_2(t)]$$
$$u_2(t) = \begin{cases} 1/\sqrt{2}, 0 \le t < 1 \\ -1/\sqrt{2}, 1 \le t < 2 \end{cases} = \frac{1}{\sqrt{2}}[e_1(t) - e_2(t)]$$

As these two basis functions are also orthonormal:

$$< u_i(t), u_j(t) >= \int_0^2 u_i(t) u_j(t) dt = \delta[i-j] \qquad (i = 1, 2) \tag{2.232}$$

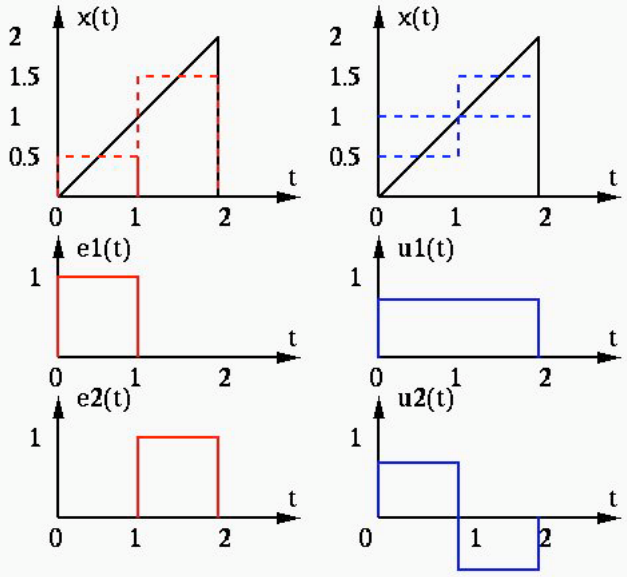

Figure 2.11 Approximation of a signal by different basis functions

the two coefficients d_1 and d_2 can be obtained independently as:

$$d_1 = \int_0^2 x(t)u_1(t)dt = \sqrt{2} \tag{2.233}$$

$$d_2 = \int_0^2 x(t)u_2(t)dt = -\frac{1}{\sqrt{2}} \tag{2.234}$$

The approximation is:

$$\hat{x}(t) = \sqrt{2}u_1(t) - \frac{1}{\sqrt{2}}u_2(t) = \begin{cases} 0.5, 0 \le t < 1 \\ 1.5, 1 \le t < 2 \end{cases} \tag{2.235}$$

The two approximations happen to be identical as shown in Fig.2.11.

We can make the following observations:

- The first basis $\{e_1(t), e_2(t)\}$ is the standard basis that represents the signal $x(t)$ in time domain, the two coefficients c_1 and c_2 are simply two time samples of $x(t)$.
- 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.)

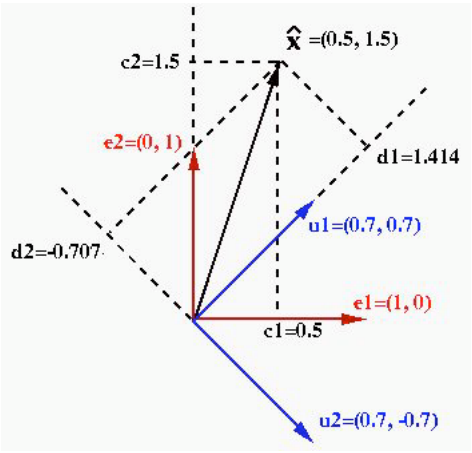

Figure 2.12 Representation of a signal vector under two different bases

- The second basis $\{u_1(t), u_2(t)\}$ is a rotated version of the first basis $\{e_1(t), e_2(t)\}$, 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 $\theta = -45°$:

$$\begin{bmatrix} d_2 \\ d_1 \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} c_2 \\ c_1 \end{bmatrix} = \begin{bmatrix} \sqrt{2}/2 & -\sqrt{2}/2 \\ \sqrt{2}/2 & \sqrt{2}/2 \end{bmatrix} \begin{bmatrix} 1/2 \\ 3/2 \end{bmatrix} = \begin{bmatrix} -1/\sqrt{2} \\ \sqrt{2} \end{bmatrix} \tag{2.236}$$

Example 2.10: The temperature is measured every 3 hours to obtain 8 samples for one particular day 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 $\boldsymbol{x} = [x_1, \cdots, x_8]^T = [65, 60, 65, 70, 75, 80, 75, 70]^T$ in an 8-D vector space, where the nth element x_n is the coefficient for the nth standard basis vector $\boldsymbol{e}_n = [0, \cdots, 0, 1, 0, \cdots, 0]^T$ (all elements are zero except the nth one), i.e.,

$$\boldsymbol{x} = \sum_{n=1}^{8} x_n \boldsymbol{e}_n$$

- This 8-D signal vector $\boldsymbol{x}$ can be approximated as $\hat{\boldsymbol{x}} = c_1 \boldsymbol{b}_1$ in a 1-D subspace spanned by $\boldsymbol{b}_1 = [1, 1, 1, 1, 1, 1, 1, 1]^T$. Here the coefficient can be obtained as:

$$c_1 = \frac{< \boldsymbol{x}, \boldsymbol{b}_1 >}{< \boldsymbol{b}_1, \boldsymbol{b}_1 >} = \frac{560}{8} = 70$$

which represents the average or DC component of the daily temperature. The approximation is:

$$\hat{\boldsymbol{x}} = c_1\boldsymbol{b}_1 = [70, 70, 70, 70, 70, 70, 70, 70]^T$$

The error vector is $\tilde{\boldsymbol{x}} = \boldsymbol{x} - \hat{\boldsymbol{x}} = [-5, -10, -5, 0, 5, 10, 5, 0]^T$ and the error is $||\tilde{\boldsymbol{x}}||^2 = 300$.

- The signal can be better approximated in a 2-D subspace spanned by $\boldsymbol{b}_1$ and $\boldsymbol{b}_2 = [1, 1, 1, 1, -1, -1, -1, -1]^T$. As $\boldsymbol{b}_2$ is orthogonal to $\boldsymbol{b}_1$, its coefficients c_2 can be found independently as

$$c_2 = \frac{< \boldsymbol{x}, \boldsymbol{b}_2 >}{< \boldsymbol{b}_2, \boldsymbol{b}_2 >} = \frac{-40}{8} = -5$$

which represents the temperature difference between morning and afternoon. The approximation is:

$$\hat{\boldsymbol{x}} = c_1\boldsymbol{b}_1 + c_2\boldsymbol{b}_2 = [65.65, 65, 65, 75, 75, 75, 75]^T$$

The error vector is $\tilde{\boldsymbol{x}} = \boldsymbol{x} - \hat{\boldsymbol{x}} = [0, -5, 0, 5, 0, 5, 0, -5]^T$ and the error is $||\tilde{\boldsymbol{x}}||^2 = 100$.

- The approximation can be further improved if a third basis vector $\boldsymbol{b}_3 = [1, 1, -1, -1, -1, -1, 1, 1]^T$ is added. Note that all three basis vectors are orthogonal to each other. The coefficient c_3 can also be independently obtained as

$$c_3 = \frac{< \boldsymbol{x}, \boldsymbol{b}_3 >}{< \boldsymbol{b}_3, \boldsymbol{b}_3 >} = \frac{-20}{8} = -2.5$$

which represents the temperature difference between day-time and night-time. The approximation can be expressed as:

$$\hat{\boldsymbol{x}} = c_1\boldsymbol{b}_1 + c_2\boldsymbol{b}_2 + c_3\boldsymbol{b}_3 = [62.5, 62, 5, 67.5, 67.5, 77.5, 77.5, 72.5, 72.5]^T$$

The error vector is $\tilde{\boldsymbol{x}} = \boldsymbol{x} - \hat{\boldsymbol{x}} = [2.5, -2.5, -2.5, 2.5, -2.5, 2.5, 2.5, -2.5]^T$ and the error is $||\tilde{\boldsymbol{x}}||^2 = 50$.

We can make the following observations:

- The original 8-D signal vector $\boldsymbol{x}$ can be approximated by k basis vectors spanning a k-D subspace $1 \leq k \leq 8$. 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 contain more energy compared to the latter. In order to reduce error when approximating the signal, low-frequency basis functions should be considered 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 be smaller compared to those for the slow-varying basis functions, they are more likely to be affected by noise such as some random fluctuation,

therefore they are less significant and could be neglected (filtered out) without losing much essential information.

- In fact, the above three basis vectors $\boldsymbol{b}_1$, $\boldsymbol{b}_2$ and $\boldsymbol{b}_3$ are the first three basis vectors of the sequency-ordered Hadamard transform to be discussed in a later chapter.

2.4 Frames and Biorthogonal Bases

2.4.1 Frames

Previously we considered the representation of a signal vector $\boldsymbol{x} \in H$ as some linear combination of an orthogonal basis $\{\boldsymbol{u}_n\}$ that spans the space:

$$\boldsymbol{x} = \sum_n c_n \boldsymbol{u}_n = \sum_n < \boldsymbol{x}, \boldsymbol{u}_n > \boldsymbol{u}_n \tag{2.237}$$

and Parseval's identity $||\boldsymbol{x}||^2 = ||\boldsymbol{c}||^2$ indicates that $\boldsymbol{x}$ is equivalently represented by the coefficients $\boldsymbol{c}$ without any redundancy. However, sometimes it may not be easy or even possible to identify a set of linearly independent basis vectors in the space. In such cases we could still consider representing a signal vector $\boldsymbol{x}$ by a set of vectors $\{\boldsymbol{f}_n\}$ which are not linearly independent and therefore do not form a basis of the space. A main issue though is the redundancy that exists among such a set of vectors. For example, as it is now possible to find a set of coefficients d_n so that $\sum_n d_n \boldsymbol{f}_n = 0$, an immediate consequence is that the representation is not unique:

$$\boldsymbol{x} = \sum_n c_n \boldsymbol{f}_n = \sum_n c_n \boldsymbol{f}_n + \sum_n d_n \boldsymbol{f}_n = \sum_n (c_n + d_n) \boldsymbol{f}_n \tag{2.238}$$

The consequence of the redundancy issue is that Parseval's identify no longer holds. The energy contained in the coefficients $||\boldsymbol{c}||^2$ may be either higher or lower then the actual energy $||\boldsymbol{x}||^2$ in the signal. We obviously need to develop some theory to address this issue when using non-independent vectors for signal representation.

First, in order for the expansion $\boldsymbol{x} = \sum_n c_n \boldsymbol{f}_n$ to be a precise representation of the signal vector $\boldsymbol{x}$ in terms of a set of coefficients $c_n =< \boldsymbol{x}, \boldsymbol{f}_n >$, we need to guarantee that for any vectors $\boldsymbol{x}, \boldsymbol{y} \in H$, the following always holds:

$$< \boldsymbol{x}, \boldsymbol{f}_n > =< \boldsymbol{y}, \boldsymbol{f}_n > \qquad \text{iff} \qquad \boldsymbol{x} = \boldsymbol{y} \tag{2.239}$$

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:

$$\text{if} \quad ||\boldsymbol{x}-\boldsymbol{y}||^2 \to 0, \quad \text{then} \quad \sum_n |<\boldsymbol{x},\boldsymbol{f}_n> - <\boldsymbol{y},\boldsymbol{f}_n>|^2 \to 0$$

i.e.,

$$\sum_n |<\boldsymbol{x},\boldsymbol{f}_n> - <\boldsymbol{y},\boldsymbol{f}_n>|^2 \leq B||\boldsymbol{x}-\boldsymbol{y}||^2$$

where $0 < B < \infty$ is a positive real constant which could be either greater or smaller than 1. In particular if $\boldsymbol{y} = \boldsymbol{0}$ and therefore $<\boldsymbol{y},\boldsymbol{f}_n>= 0$, we have:

$$\sum_n |<\boldsymbol{x},\boldsymbol{f}_n>|^2 \leq B||\boldsymbol{x}||^2 \tag{2.240}$$

- **Stable reconstruction:**
  If the difference between two sets of coefficients is small, the difference between the reconstructed vectors should also be small:

$$\text{if} \quad \sum_n |<\boldsymbol{x},\boldsymbol{f}_n> - <\boldsymbol{y},\boldsymbol{f}_n>|^2 \to 0, \quad \text{then} \quad ||\boldsymbol{x}-\boldsymbol{y}||^2 \to 0$$

  i.e.,

$$A||\boldsymbol{x}-\boldsymbol{y}||^2 \leq \sum_n |<\boldsymbol{x},\boldsymbol{f}_n> - <\boldsymbol{y},\boldsymbol{f}_n>|^2$$

  where $0 < A < \infty$ is also a positive real constant, either greater or smaller than 1. Again if $\boldsymbol{y} = \boldsymbol{0}$ and $<\boldsymbol{y},\boldsymbol{f}_n>= 0$, we have:

$$A||\boldsymbol{x}||^2 \leq \sum_n |<\boldsymbol{x},\boldsymbol{f}_n>|^2 \tag{2.241}$$

Combining Eqs.2.240 and 2.241, we have the following definition:

Definition 2.22. *A family of finite or infinite vectors $\{\boldsymbol{f}_n\}$ 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 $\boldsymbol{x} \in H$, the following holds:*

$$A||\boldsymbol{x}||^2 \leq \sum_n |<\boldsymbol{x},\boldsymbol{f}_n>|^2 \leq B||\boldsymbol{x}||^2 \tag{2.242}$$

In particular, if $A = B$, i.e.,

$$A||\boldsymbol{x}||^2 = \sum_n |<\boldsymbol{x},\boldsymbol{f}_n>|^2 \tag{2.243}$$

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 $\boldsymbol{x} \in H$ as a linear combination $\boldsymbol{x} = \sum_n c_n \boldsymbol{f}_n$ of a set of frame vectors $\{\boldsymbol{f}_n\}$. The process of finding the

coefficients c_n needed in the combination can be considered as a *frame transformation*, denoted by F^*, that maps the given $\boldsymbol{x}$ to a coefficient vector $\boldsymbol{c}$:

$$F^*\boldsymbol{x} = \boldsymbol{c} = [\cdots, c_n, \cdots]^T = [\cdots, <\boldsymbol{x}, \boldsymbol{f}_n>, \cdots]^T \tag{2.244}$$

where we defined $c_n =< \boldsymbol{x}, \boldsymbol{f}_n >$ following the unitary transformation in Eq.2.83, and F^* is the adjoint of another transformation F, which can be found from the following inner product (Eq.2.123):

$$< \boldsymbol{c}, F^*\boldsymbol{x} >= \sum_n c_n \overline{< \boldsymbol{x}, \boldsymbol{f}_n >} = \sum_n c_n \ < \boldsymbol{f}_n, \boldsymbol{x} >=< \sum_n c_n \boldsymbol{f}_n, \boldsymbol{x} >=< F\boldsymbol{c}, \boldsymbol{x} > \tag{2.245}$$

where F is an transformation that generates a vector as a linear combination of frame $\{\boldsymbol{f}_n\}$ based on a given set of coefficients:

$$F\boldsymbol{c} = \sum_n c_n \boldsymbol{f}_n \tag{2.246}$$

Note that in general $F\boldsymbol{c} \neq \boldsymbol{x}$. We further define an operator FF^*:

$$FF^*\boldsymbol{x} = F(F^*\boldsymbol{x}) = F\boldsymbol{c} = \sum_n < \boldsymbol{x}, \boldsymbol{f}_n > \boldsymbol{f}_n \tag{2.247}$$

Applying its inverse $(FF^*)^{-1}$ to both sides of the equation, we get:

$$\begin{aligned} \boldsymbol{x} &= (FF^*)^{-1}[\sum_n < \boldsymbol{x}, \boldsymbol{f}_n > \boldsymbol{f}_n] = \sum_n < \boldsymbol{x}, \boldsymbol{f}_n > (FF^*)^{-1}\boldsymbol{f}_n \\ &= \sum_n < \boldsymbol{x}, \boldsymbol{f}_n > \tilde{\boldsymbol{f}}_n = \sum_n c_n \tilde{\boldsymbol{f}}_n \end{aligned} \tag{2.248}$$

where $\tilde{\boldsymbol{f}}_n$, called the *dual vector* of $\boldsymbol{f}_n$, is defined as:

$$\tilde{\boldsymbol{f}}_n = (FF^*)^{-1}\boldsymbol{f}_n, \quad \text{i.e.} \quad \boldsymbol{f}_n = (FF^*)\tilde{\boldsymbol{f}}_n \tag{2.249}$$

We recognize that $(FF^*)^{-1}F = (F^*)^-$ is actually the pseudo-inverse of F^*, and define it as another operator $\tilde{F} = (F^*)^-$. Now Eq.2.248 can also be written as:

$$\boldsymbol{x} = \tilde{F}\boldsymbol{c} = \tilde{F}[\cdots, c_n, \cdots]^T = \sum_n c_n \tilde{\boldsymbol{f}}_n = \sum_n < \boldsymbol{x}, \boldsymbol{f}_n > \tilde{\boldsymbol{f}}_n \tag{2.250}$$

This is the reconstruction of vector $\boldsymbol{x}$ based on the coefficients $\boldsymbol{c}$ obtained in Eq.2.244.

The adjoint of $\tilde{F}$ can be found from the following inner product (reversal of the steps in Eq.2.245):

$$< \tilde{F}\boldsymbol{c}, \boldsymbol{x} >=< \sum_n c_n \tilde{\boldsymbol{f}}_n, \boldsymbol{x} >= \sum_n c_n \ < \tilde{\boldsymbol{f}}_n, \boldsymbol{x} >= \sum_n c_n \overline{< \boldsymbol{x}, \tilde{\boldsymbol{f}}_n >} =< \boldsymbol{c}, \tilde{F}^*\boldsymbol{x} > \tag{2.251}$$

where $\tilde{F}^*$ is the adjoint of $\tilde{F}$:

$$\tilde{F}^*\boldsymbol{x} = [\cdots, <\boldsymbol{x}, \tilde{\boldsymbol{f}}_n >, \cdots]^T = [\cdots, d_n, \cdots]^T = \boldsymbol{d} \tag{2.252}$$

and we have defined $d_n =< \boldsymbol{x}, \tilde{\boldsymbol{f}}_n >$.

Theorem 2.8. *A vector $\boldsymbol{x} \in H$ can be equivalently represented by either of the two dual frames $\{\boldsymbol{f}_n\}$ or $\{\tilde{\boldsymbol{f}}_n\}$:*

$$\boldsymbol{x} = \sum_n < \boldsymbol{x}, \tilde{\boldsymbol{f}}_n > \boldsymbol{f}_n = \sum_n < \boldsymbol{x}, \boldsymbol{f}_n > \tilde{\boldsymbol{f}}_n \tag{2.253}$$

Proof: Consider the inner product $< \boldsymbol{x}, \boldsymbol{x} >$, with the first $\boldsymbol{x}$ replaced by the expression in Eq.2.248:

$$\begin{aligned} < \boldsymbol{x}, \boldsymbol{x} > &= < \sum_n < \boldsymbol{x}, \boldsymbol{f}_n > \tilde{\boldsymbol{f}}_n, \boldsymbol{x} > = \sum_n < \boldsymbol{x}, \boldsymbol{f}_n >< \tilde{\boldsymbol{f}}_n, \boldsymbol{x} > \\ &= < \boldsymbol{x}, \sum_n \overline{< \tilde{\boldsymbol{f}}_n, \boldsymbol{x} >} \boldsymbol{f}_n > = < \boldsymbol{x}, \sum_n < \boldsymbol{x}, \tilde{\boldsymbol{f}}_n > \boldsymbol{f}_n > \end{aligned} \tag{2.254}$$

Comparing the two sides of the equation, we get:

$$\boldsymbol{x} = \sum_n < \boldsymbol{x}, \tilde{\boldsymbol{f}}_n > \boldsymbol{f}_n \tag{2.255}$$

Combining this result with Eq.2.248, we get Eq.2.253. This completes the proof.

Note that Eq.2.255 can also be written as the following due to Eq.2.246:

$$\boldsymbol{x} = \sum_n < \boldsymbol{x}, \tilde{\boldsymbol{f}}_n > \boldsymbol{f}_n = \sum_n d_n \boldsymbol{f}_n = F\boldsymbol{d} \tag{2.256}$$

We can now combine Eqs.2.244 and 2.252 together with Eq.2.253 to form two alternative frame transformation pairs based on either frame $\{\boldsymbol{f}_n\}$ or its dual $\{\tilde{\boldsymbol{f}}_n\}$:

$$\begin{cases} c_n = < \boldsymbol{x}, \boldsymbol{f}_n > \\ \boldsymbol{x} = \sum_n c_n \tilde{\boldsymbol{f}}_n = \sum_n < \boldsymbol{x}, \boldsymbol{f}_n > \tilde{\boldsymbol{f}}_n \end{cases} \qquad \begin{cases} d_n = < \boldsymbol{x}, \tilde{\boldsymbol{f}}_n > \\ \boldsymbol{x} = \sum_n d_n \boldsymbol{f}_n = \sum_n < \boldsymbol{x}, \tilde{\boldsymbol{f}}_n > \boldsymbol{f}_n \end{cases} \tag{2.257}$$

These equations are respectively the forward and inverse frame transformation of $\boldsymbol{x}$ based on frame and its dual, which can also be expressed (due to Eqs.2.250 and 2.256) more concisely as:

$$\begin{cases} \boldsymbol{c} = F^* \boldsymbol{x} \\ \boldsymbol{x} = \tilde{F}\boldsymbol{c} \end{cases}, \qquad \begin{cases} \boldsymbol{d} = \tilde{F}^* \boldsymbol{x} \\ \boldsymbol{x} = F\boldsymbol{d} \end{cases} \tag{2.258}$$

The frame transformation pairs in Eqs.2.258 and 2.257 can be considered as the generalization of the unitary transformation given in Eq.2.179. From Eq.2.258 we also see that

$$\tilde{F}F^* \boldsymbol{x} = F\tilde{F}^* \boldsymbol{x} = \boldsymbol{x} \tag{2.259}$$

i.e., $\tilde{F}F^* = F\tilde{F}^* = I$ is an identity operator, same as $UU^* = UU^{-1} = I$ in the unitary transformation.

In frame transformation, the signal energy is related to the coefficients by:

$$\begin{aligned} ||\boldsymbol{x}||^2 &= < \boldsymbol{x}, \boldsymbol{x} > = < \tilde{F}\boldsymbol{c}, \boldsymbol{x} > = < \boldsymbol{c}, \tilde{F}^* \boldsymbol{x} > = < \boldsymbol{c}, \boldsymbol{d} > \\ &= < F\boldsymbol{d}, \boldsymbol{x} > = < \boldsymbol{d}, \tilde{F}^* \boldsymbol{x} > = < \boldsymbol{d}, \boldsymbol{c} > \end{aligned} \tag{2.260}$$

However, we see that Parseval's identity is no longer valid:

$$||\boldsymbol{c}||^2 =< \boldsymbol{c}, \boldsymbol{c} >=< F^*\boldsymbol{x}, F^*\boldsymbol{x} >=< FF^*\boldsymbol{x}, \boldsymbol{x} >\neq< \boldsymbol{x}, \boldsymbol{x} >= ||\boldsymbol{x}||^2$$
$$||\boldsymbol{d}||^2 < \boldsymbol{d}, \boldsymbol{d} >=< \tilde{F}^*\boldsymbol{x}, \tilde{F}^*\boldsymbol{x} >=< \tilde{F}\tilde{F}^*\boldsymbol{x}, \boldsymbol{x} >\neq< \boldsymbol{x}, \boldsymbol{x} >= ||\boldsymbol{x}||^2 \quad (2.261)$$

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.2.247 and a vector $\boldsymbol{y}$:

$$< FF^*\boldsymbol{x}, \boldsymbol{y} >= \sum_n < \boldsymbol{x}, \boldsymbol{f}_n >< \boldsymbol{f}_n, \boldsymbol{y} >=< \boldsymbol{x}, \sum_n \overline{< \boldsymbol{f}_n, \boldsymbol{y} >}\boldsymbol{f}_n >$$
$$= < \boldsymbol{x}, \sum_n < \boldsymbol{y}, \boldsymbol{f}_n > \boldsymbol{f}_n >=< \boldsymbol{x}, FF^*\boldsymbol{y} > \quad (2.262)$$

which indicates that FF^* is a self-adjoint operator. If we let $\{\lambda_n\}$ and $\{\boldsymbol{\phi}_n\}$ be the eigenvalues and eigenvectors of FF^*, i.e.,

$$FF^*\boldsymbol{\phi}_n = \lambda_n\boldsymbol{\phi}_n, \qquad \text{(for all } n\text{)} \quad (2.263)$$

then all $\{\lambda_n\}$ are real, and all $\{\boldsymbol{\phi}_n\}$ are orthogonal $< \boldsymbol{\phi}_m, \boldsymbol{\phi}_n >= \delta[m-n]$ and they form a complete orthogonal system (Theorem 2.4). Now $\boldsymbol{x}$ can also be expanded in terms of these eigenvectors as:

$$\boldsymbol{x} = \sum_n < \boldsymbol{x}, \boldsymbol{\phi}_n > \boldsymbol{\phi}_n \quad (2.264)$$

and the energy contained in $\boldsymbol{x}$ is:

$$||\boldsymbol{x}||^2 =< \boldsymbol{x}, \boldsymbol{x} >=< \sum_m < \boldsymbol{x}, \boldsymbol{\phi}_m > \boldsymbol{\phi}_m, \sum_n < \boldsymbol{x}, \boldsymbol{\phi}_n > \boldsymbol{\phi}_n >$$
$$= \sum_m \sum_n < \boldsymbol{x}, \boldsymbol{\phi}_m > \overline{< \boldsymbol{x}, \boldsymbol{\phi}_n >} < \boldsymbol{\phi}_m, \boldsymbol{\phi}_n >= \sum_n | < \boldsymbol{x}, \boldsymbol{\phi}_n > |^2 \quad (2.265)$$

Correspondingly for the dual frame transformation $\tilde{F}$, we have:

$$\tilde{F}\tilde{F}^* = [(FF^*)^{-1}F]\,[(FF^*)^{-1}F]^* = (FF^*)^{-1}FF^*(FF^*)^{-1} = (FF^*)^{-1} \quad (2.266)$$

whose eigenvalues and eigenvectors are respectively $\{1/\lambda_n\}$ and $\boldsymbol{\phi}_n$, i.e.,:

$$\tilde{F}\tilde{F}^*\boldsymbol{\phi}_n = (FF^*)^{-1}\boldsymbol{\phi}_n = \frac{1}{\lambda_n}\boldsymbol{\phi}_n, \qquad \text{(for all } n\text{)} \quad (2.267)$$

Theorem 2.9. *The frame transformation coefficients* $c_n =< \boldsymbol{x}, \boldsymbol{f}_n >$ *and* $d_n =< \boldsymbol{x}, \tilde{\boldsymbol{f}}_n >$ *satisfy respectively the following inequalities:*

$$\lambda_{min}||\boldsymbol{x}||^2 \leq \sum_n | < \boldsymbol{x}, \boldsymbol{f}_n > |^2 = ||\boldsymbol{c}||^2 = ||F^*\boldsymbol{x}||^2 \leq \lambda_{max}||\boldsymbol{x}||^2 \quad (2.268)$$

$$\frac{1}{\lambda_{max}}||\boldsymbol{x}||^2 \leq \sum_n | < \boldsymbol{x}, \tilde{\boldsymbol{f}}_n > |^2 = ||\boldsymbol{d}||^2 = ||\tilde{F}^*\boldsymbol{x}||^2 \leq \frac{1}{\lambda_{min}}||\boldsymbol{x}||^2 \quad (2.269)$$

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 the frame is tight:*

$$\sum_n |< \boldsymbol{x}, \boldsymbol{f}_n >|^2 = \lambda ||\boldsymbol{x}||^2, \qquad \sum_n |< \boldsymbol{x}, \tilde{\boldsymbol{f}}_n >|^2 = \frac{1}{\lambda}||\boldsymbol{x}||^2 \tag{2.270}$$

Proof: Applying $(FF^*)^{-1}$ to both sides of Eq.2.256 we get:

$$(FF^*)^{-1}\boldsymbol{x} = \sum_n < \boldsymbol{x}, \tilde{\boldsymbol{f}}_n > (FF^*)^{-1}\boldsymbol{f}_n = \sum_n < \boldsymbol{x}, \tilde{\boldsymbol{f}}_n > \tilde{\boldsymbol{f}}_n \tag{2.271}$$

This result and Eq.2.247 form a symmetric pair:

$$(FF^*)\boldsymbol{x} = \sum_n < \boldsymbol{x}, \boldsymbol{f}_n > \boldsymbol{f}_n \tag{2.272}$$

$$(FF^*)^{-1}\boldsymbol{x} = \sum_n < \boldsymbol{x}, \tilde{\boldsymbol{f}}_n > \tilde{\boldsymbol{f}}_n \tag{2.273}$$

Taking the inner product of each of these equations with $\boldsymbol{x}$, we get:

$$< (FF^*)\boldsymbol{x}, \boldsymbol{x} > = \sum_n < \boldsymbol{x}, \boldsymbol{f}_n >< \boldsymbol{f}_n, \boldsymbol{x} >= \sum_n |< \boldsymbol{x}, \boldsymbol{f}_n >|^2 \tag{2.274}$$

$$< (FF^*)^{-1}\boldsymbol{x}, \boldsymbol{x} > = \sum_n < \boldsymbol{x}, \tilde{\boldsymbol{f}}_n >< \tilde{\boldsymbol{f}}_n, \boldsymbol{x} >= \sum_n |< \boldsymbol{x}, \tilde{\boldsymbol{f}}_n >|^2 \tag{2.275}$$

These two expressions represent the energy contained in each of the two sets of coefficients $< \boldsymbol{x}, \boldsymbol{f}_n >$ and $< \boldsymbol{x}, \tilde{\boldsymbol{f}}_n >$. On the other hand, applying operator FF^* to $\boldsymbol{x}$ in Eq.2.264, we get:

$$\begin{aligned} FF^*\boldsymbol{x} &= FF^*(\sum_n < \boldsymbol{x}, \boldsymbol{\phi}_n > \boldsymbol{\phi}_n) = \sum_n < \boldsymbol{x}, \boldsymbol{\phi}_n > FF^*\boldsymbol{\phi}_n \\ &= \sum_n < \boldsymbol{x}, \boldsymbol{\phi}_n > \lambda_n \boldsymbol{\phi}_n \end{aligned} \tag{2.276}$$

and

$$\begin{aligned} < FF^*\boldsymbol{x}, \boldsymbol{x} > &= < \sum_n < \boldsymbol{x}, \boldsymbol{\phi}_n > \lambda_n \boldsymbol{\phi}_n, \boldsymbol{x} >= \sum_n < \boldsymbol{x}, \boldsymbol{\phi}_n > \lambda_n < \boldsymbol{\phi}_n, \boldsymbol{x} > \\ &= \sum_n \lambda_n |< \boldsymbol{x}, \boldsymbol{\phi}_n >|^2 \begin{cases} \leq \lambda_{max}||\boldsymbol{x}||^2 \\ \geq \lambda_{min}||\boldsymbol{x}||^2 \end{cases} \end{aligned} \tag{2.277}$$

The last step is due to Eq.2.265. Now replacing the left hand side by Eq.2.274, we get:

$$\lambda_{min}||\boldsymbol{x}||^2 \leq \sum_n |< \boldsymbol{x}, \boldsymbol{f}_n >|^2 \leq \lambda_{max}||\boldsymbol{x}||^2 \tag{2.278}$$

Similarly, we can also get:

$$\sum_n |< \boldsymbol{x}, \tilde{\boldsymbol{f}}_n >|^2 = \sum_n \frac{1}{\lambda_n} |< \boldsymbol{x}, \tilde{\boldsymbol{\phi}}_n >|^2 \tag{2.279}$$

and

$$\frac{1}{\lambda_{max}}||\boldsymbol{x}||^2 \leq \sum_n |<\boldsymbol{x}, \tilde{\boldsymbol{f}}_n>|^2 \leq \frac{1}{\lambda_{min}}||\boldsymbol{x}||^2 \tag{2.280}$$

The proof is complete.

Comparing these results with Parseval's identity $||U\boldsymbol{x}||^2 = ||\boldsymbol{x}||^2$ (Eq.2.177) for a unitary transformation, we see that the frame transformation does not conserve signal energy, due obviously to the redundancy of the non-independent frame vectors. However, as shown in Eq.2.260, the energy is conserved in the following fashion when both sets of coefficients are involved:

$$||\boldsymbol{x}||^2 =<\boldsymbol{c}, \boldsymbol{d}>= \sum_n <\boldsymbol{x}, \boldsymbol{f}_n> \overline{<\boldsymbol{x}, \tilde{\boldsymbol{f}}_n>} \tag{2.281}$$

Theorem 2.10. *Let λ_k and $\boldsymbol{\phi}_k$ be the kth eigenvalue and the corresponding eigenvector of operator FF^*:*

$$FF^*\boldsymbol{\phi}_k = \lambda_k\boldsymbol{\phi}_k, \qquad \textit{(for all k)} \tag{2.282}$$

Then

$$\sum_k \lambda_k = \sum_n ||\boldsymbol{f}_n||^2 \tag{2.283}$$

Proof: As noted before, FF^* is self-adjoint and λ_k's are real and $<\boldsymbol{\phi}_k, \boldsymbol{\phi}_l>= \delta[k-l]$. We have:

$$\begin{aligned}\sum_k \lambda_k &= \sum_k \lambda_k <\boldsymbol{\phi}_k, \boldsymbol{\phi}_k>= \sum_k <FF^*\boldsymbol{\phi}_k, \boldsymbol{\phi}_k> \\ &= \sum_k <\sum_n <\boldsymbol{\phi}_k, \boldsymbol{f}_n> \boldsymbol{f}_n, \boldsymbol{\phi}_k>= \sum_k \sum_n |<\boldsymbol{f}_n, \boldsymbol{\phi}_k>|^2\end{aligned} \tag{2.284}$$

On the other hand:

$$\begin{aligned}||\boldsymbol{f}_n||^2 &= <\boldsymbol{f}_n, \boldsymbol{f}_n>=<\sum_k <\boldsymbol{f}_n, \boldsymbol{\phi}_k> \boldsymbol{\phi}_k, \sum_k <\boldsymbol{f}_n, \boldsymbol{\phi}_l> \boldsymbol{\phi}_l> \\ &= \sum_k \sum_l <\boldsymbol{f}_n, \boldsymbol{\phi}_k> \overline{<\boldsymbol{f}_n, \boldsymbol{\phi}_l>} <\boldsymbol{\phi}_k, \boldsymbol{\phi}_l>= \sum_k |<\boldsymbol{f}_n, \boldsymbol{\phi}_k>|^2\end{aligned} \tag{2.285}$$

Therefore we get

$$\sum_n ||\boldsymbol{f}_n||^2 = \sum_n \sum_k |<\boldsymbol{f}_n, \boldsymbol{\phi}_k>|^2 = \sum_k \lambda_k \tag{2.286}$$

The proof is complete.

Definition 2.23. *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* $\{\boldsymbol{f}_n\}$ *and its dual* $\{\tilde{\boldsymbol{f}}_n\}$ *form a pair of biorthogonal bases satisfying*

$$< \boldsymbol{f}_m, \tilde{\boldsymbol{f}}_n >= \delta[m-n], \qquad m, n \in \mathbb{Z} \tag{2.287}$$

Proof: We let $\boldsymbol{x} = \boldsymbol{f}_m$ in Eq.2.253 and get:

$$\boldsymbol{f}_m = \sum_n < \boldsymbol{f}_m, \tilde{\boldsymbol{f}}_n > \boldsymbol{f}_n \tag{2.288}$$

Since these vectors are linearly independent, i.e., $\boldsymbol{f}_m$ cannot be expressed as a linear combination of the rest of the frame vectors, the equation above has only one interpretation: all coefficients $< \boldsymbol{f}_m, \tilde{\boldsymbol{f}}_n >$ for $m \neq n$ are zero except the mth one $< \boldsymbol{f}_m, \tilde{\boldsymbol{f}}_m >= 1$. In other words, these frame vectors are orthogonal to their dual vectors, i.e., Eq.2.288 holds. This completes the proof.

Summarizing the discussion above, we see that signal representation by a set of orthogonal and linearly independent basis vectors $\boldsymbol{x} = \sum_n c_n \boldsymbol{\phi}_n = \sum_n < \boldsymbol{x}, \boldsymbol{b}_n > \boldsymbol{x}_n$ (Eq.2.76) is much generalized to a set of frame vectors, which are in general neither linearly independent nor orthogonal. Now the signal can be represented in either of the two mutually dual frames, and the frame transformation and its inverse are pseudo-inverse of each other. Moreover, now 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.2.268, 2.269, and 2.281.

On the other hand, if specially the eigenvalues of operator FF^* are all the same, then $A = B = \lambda$ and the fame is tight. Moreover, if all eigenvalues are $\lambda = 1$, then $FF^* = I$ becomes an identity operator, i.e., F becomes a unitary operator satisfying $F^* = F^{-1}$. Now the pseudo-inverse $F^- = (FF^*)^{-1}F^* = F^* = F^{-1}$ becomes a regular inverse, the frame and its dual become identical (Eq.2.249). Moreover, the biorthogonality in Eq.2.287 becomes regular orthogonality, and Eqs.2.281 become Parseval's identity.

2.4.3 Frames in Finite-Dimensional Space

Here we consider the frame transformation in an N-D unitary space $\mathbb{C}^N$. Let $\boldsymbol{F} = [\boldsymbol{f}_1, \cdots, \boldsymbol{f}_M]$ be an N by M 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. Now a vector $\boldsymbol{x} \in \mathbb{C}^N$ can be represented by either the frame $\boldsymbol{F}$ or its dual $\tilde{\boldsymbol{F}} = [\tilde{\boldsymbol{f}}_1, \cdots, \tilde{\boldsymbol{f}}_M]$. To obtain the M coefficients, we apply the frame operator F^* to $\boldsymbol{x}$ and get:

$$\boldsymbol{c} = \begin{bmatrix} < \boldsymbol{x}, \boldsymbol{f}_1 > \\ \vdots \\ < \boldsymbol{x}, \boldsymbol{f}_M > \end{bmatrix} = \begin{bmatrix} \boldsymbol{f}_1^* \boldsymbol{x} \\ \vdots \\ \boldsymbol{f}_M^* \boldsymbol{x} \end{bmatrix}_{M\times 1} = \begin{bmatrix} \boldsymbol{f}_1^* \\ \vdots \\ \boldsymbol{f}_M^* \end{bmatrix}_{M\times N} \boldsymbol{x}_{N\times 1} = \boldsymbol{F}^* \boldsymbol{x} \tag{2.289}$$

This is the forward frame transformation $F^*\boldsymbol{x} = \boldsymbol{c}$, which is actually a multiplication of $\boldsymbol{x}$ by matrix $\boldsymbol{F}^*$, the adjoint or conjugate transpose of $\boldsymbol{F}$, which can be obtained as the pseudo-inverse $(\boldsymbol{F}^*)^-$ of $\boldsymbol{f}^*$ (Eq.2.249):

$$\tilde{\boldsymbol{F}} = [\tilde{\boldsymbol{f}}_1, \cdots, \tilde{\boldsymbol{f}}_M] = (\boldsymbol{F}\boldsymbol{F}^*)^{-1}\boldsymbol{F} = (\boldsymbol{F}^*)^- \tag{2.290}$$

As the M by N matrix $\boldsymbol{F}^*$ is not invertible, the inverse transformation for the reconstruction of $\boldsymbol{x}$ from $\boldsymbol{c}$ is a multiplication by the pseudo-inverse $(\boldsymbol{F}^*)^{-1}$:

$$\boldsymbol{x} = (\boldsymbol{F}^*)^-\boldsymbol{c} = \tilde{\boldsymbol{F}}^*\boldsymbol{c} = [\tilde{\boldsymbol{f}}_1, \cdots, \tilde{\boldsymbol{f}}_M] \begin{bmatrix} < \boldsymbol{x}, \boldsymbol{f}_1 > \\ \vdots \\ < \boldsymbol{x}, \boldsymbol{f}_M > \end{bmatrix} = \sum_{n=1}^{M} < \boldsymbol{x}, \boldsymbol{f}_n > \tilde{\boldsymbol{f}}_n \tag{2.291}$$

Alternatively, $\boldsymbol{x}$ can also be represented by the dual frame $\tilde{\boldsymbol{F}}$ with coefficients:

$$\boldsymbol{d} = \tilde{\boldsymbol{F}}^*\boldsymbol{x} = \begin{bmatrix} \tilde{\boldsymbol{f}}_1^* \\ \vdots \\ \tilde{\boldsymbol{f}}_M^* \end{bmatrix} \boldsymbol{x} = \begin{bmatrix} < \boldsymbol{x}, \tilde{\boldsymbol{f}}_1 > \\ \vdots \\ < \boldsymbol{x}, \tilde{\boldsymbol{f}}_M > \end{bmatrix} \tag{2.292}$$

and the signal is represented as:

$$\boldsymbol{x} = \boldsymbol{F}\boldsymbol{d} = [\boldsymbol{f}_1, \cdots, \boldsymbol{f}_M] \begin{bmatrix} < \boldsymbol{x}, \tilde{\boldsymbol{f}}_1 > \\ \vdots \\ < \boldsymbol{x}, \tilde{\boldsymbol{f}}_M > \end{bmatrix} = \sum_{n=1}^{M} < \boldsymbol{x}, \tilde{\boldsymbol{f}}_n > \boldsymbol{f}_n \tag{2.293}$$

Theorem 2.12. *If a frame $\boldsymbol{F} = [\boldsymbol{f}_1, \cdots, \boldsymbol{f}_M]$ in $\mathbb{C}^N$ is tight, i.e., all eigenvalues $A = B = \lambda$ of $\boldsymbol{F}\boldsymbol{F}^*$ are the same, and all frame vectors are normalized $||\boldsymbol{f}_n|| = 1$, then the frame bound is M/N.*

Proof: As $\boldsymbol{F}\boldsymbol{F}^*$ is an N by N matrix, it has N eigenvalues $\lambda_k = \lambda$ for all $k = 1, \cdots, N$. Then Theorem 2.10 becomes:

$$\sum_{k=1}^{N} \lambda_k = N\lambda = \sum_{n=1}^{M} ||\boldsymbol{f}_n||^2 = M \tag{2.294}$$

i.e., $\lambda = M/N$. The proof is complete.

In particular, if $M = N$ linearly independent frame vectors are used, then they form a Riesz basis in $\mathbb{C}^N$, and $\boldsymbol{F} = [\boldsymbol{f}_1, \cdots, \boldsymbol{f}_N]$ becomes an N by N invertible matrix, and its pseudo-inverse is just a regular inverse, and Eq.2.290 becomes:

$$\tilde{\boldsymbol{F}} = [\tilde{\boldsymbol{f}}_1, \cdots, \tilde{\boldsymbol{f}}_N] = (\boldsymbol{F}^*)^- = (\boldsymbol{F}^*)^{-1} \tag{2.295}$$

We now have:

$$\begin{bmatrix} \boldsymbol{f}_1^* \\ \vdots \\ \boldsymbol{f}_N^* \end{bmatrix} [\tilde{\boldsymbol{f}}_1, \cdots, \tilde{\boldsymbol{f}}_N] = \boldsymbol{F}^*\tilde{\boldsymbol{F}} = \boldsymbol{F}^*(\boldsymbol{F}^*)^{-1} = \boldsymbol{I} \tag{2.296}$$

which indicates that these Riesz vectors are indeed biorthogonal:

$$< \boldsymbol{f}_m, \tilde{\boldsymbol{f}}_n >= \delta[m-n], \qquad (m,n=1,\cdots,N) \tag{2.297}$$

Moreover, if these N vectors are also orthogonal, i.e., $< \boldsymbol{f}_m, \boldsymbol{f}_n >= \delta[m-n]$, then $\boldsymbol{F}=[\boldsymbol{f}_1,\cdots,\boldsymbol{f}_N]$ becomes a unitary matrix $\boldsymbol{U}=[\boldsymbol{u}_1,\cdots,\boldsymbol{u}_N]$ satisfying $\boldsymbol{U}^*=\boldsymbol{U}^{-1}$, and $\tilde{\boldsymbol{U}}=(\boldsymbol{U}^*)^{-1}=\boldsymbol{U}$, 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 $\boldsymbol{U}^*\boldsymbol{x}=\boldsymbol{c}$ and the inverse is simply $\boldsymbol{U}\boldsymbol{c}=\boldsymbol{x}$. Also the eigenvalues of $\boldsymbol{U}\boldsymbol{U}^*=\boldsymbol{I}$ are all $\lambda_n=1$, and $||\boldsymbol{u}_n||^2=1$, Theorem 2.10 holds trivially.

Example 2.11: Three normalized vectors in $\mathbb{R}^2$ form a frame:

$$\boldsymbol{F}=[\boldsymbol{f}_1,\boldsymbol{f}_2,\boldsymbol{f}_3]=\begin{bmatrix}-1 & 1/2 & 1/2\\ 0 & \sqrt{3}/2 & -\sqrt{3}/2\end{bmatrix}$$

and we have:

$$\boldsymbol{F}\boldsymbol{F}^T=\frac{3}{2}\begin{bmatrix}1&0\\0&1\end{bmatrix}, \qquad (\boldsymbol{F}\boldsymbol{F}^T)^{-1}=\frac{2}{3}\begin{bmatrix}1&0\\0&1\end{bmatrix}$$

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 $\tilde{\boldsymbol{F}}$ can be found as the pseudo-inverse of $\boldsymbol{F}^T$:

$$\tilde{\boldsymbol{F}}=[\tilde{\boldsymbol{f}}_1,\tilde{\boldsymbol{f}}_2,\tilde{\boldsymbol{f}}_3]=(\boldsymbol{F}\boldsymbol{F}^T)^{-1}\boldsymbol{F}=\frac{2}{3}\boldsymbol{F}=\begin{bmatrix}-2/3 & 1/3 & 1/3\\ 0 & \sqrt{3}/3 & -\sqrt{3}/3\end{bmatrix}$$

Any $\boldsymbol{x}=[x_1,x_2]^T$ can be expanded in terms of either of the two frames:

$$\boldsymbol{x}=\sum_{n=1}^{3}c_n\boldsymbol{f}_n=\sum_{n=1}^{3}<\boldsymbol{x},\boldsymbol{f}_n>\tilde{\boldsymbol{f}}_n=\sum_{n=1}^{3}d_n\boldsymbol{f}_n=\sum_{n=1}^{3}<\boldsymbol{x},\tilde{\boldsymbol{f}}_n>\boldsymbol{f}_n$$

where

$$c_1=x_1, \quad c_2=\frac{1}{2}[x_1+\sqrt{3}x_2], \quad c_3=\frac{1}{2}[x_1-\sqrt{3}x_2]$$

and

$$d_1=x_1, \quad d_2=\frac{1}{3}[x_1+\sqrt{3}x_2], \quad d_3=\frac{1}{3}[x_1-\sqrt{3}x_2]$$

The energy contained in the coefficients $\boldsymbol{c}$ and $\boldsymbol{d}$ is respectively:

$$||\boldsymbol{c}||^2=\sum_{n=1}^{3}|<\boldsymbol{x},\boldsymbol{f}_n>|^2=\frac{3}{2}||\boldsymbol{x}||^2=\lambda||\boldsymbol{x}||^2$$

and

$$||\boldsymbol{d}||^2=\sum_{n=1}^{3}|<\boldsymbol{x},\tilde{\boldsymbol{f}}_n>|^2=\frac{2}{3}||\boldsymbol{x}||^2=\frac{1}{\lambda}||\boldsymbol{x}||^2$$

Specifically if we let $\boldsymbol{x} = [1, 2]^T$, then

$$\boldsymbol{c} = \boldsymbol{F}^T\boldsymbol{x} = \begin{bmatrix} \boldsymbol{f}_1^T \\ \boldsymbol{f}_2^T \\ \boldsymbol{f}_3^T \end{bmatrix} \boldsymbol{x} = \begin{bmatrix} <\boldsymbol{x}, \boldsymbol{f}_1> \\ <\boldsymbol{x}, \boldsymbol{f}_2> \\ <\boldsymbol{x}, \boldsymbol{f}_3> \end{bmatrix} = \begin{bmatrix} -1 \\ 1+\sqrt{3} \\ 1-\sqrt{3} \end{bmatrix}$$

and

$$\boldsymbol{d} = \tilde{\boldsymbol{F}}^T\boldsymbol{x} = \begin{bmatrix} \tilde{\boldsymbol{f}}_1^T \\ \tilde{\boldsymbol{f}}_2^T \\ \tilde{\boldsymbol{f}}_3^T \end{bmatrix} \boldsymbol{x} = \begin{bmatrix} <\boldsymbol{x}, \tilde{\boldsymbol{f}}_1> \\ <\boldsymbol{x}, \tilde{\boldsymbol{f}}_2> \\ <\boldsymbol{x}, \tilde{\boldsymbol{f}}_3> \end{bmatrix} = \frac{2}{3}\begin{bmatrix} -1 \\ 1+\sqrt{3} \\ 1-\sqrt{3} \end{bmatrix}$$

Example 2.12: Consider a frame in $\mathbb{R}^2$ containing three vectors that form a frame matrix:

$$\boldsymbol{F} = [\boldsymbol{f}_1, \boldsymbol{f}_2, \boldsymbol{f}_3] = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

and we have:

$$\boldsymbol{F}\boldsymbol{F}^T = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$$

with eigenvalues $\lambda_1 = 1$, and $\lambda_2 = 3$, The dual frame is the pseudo-inverse of $\boldsymbol{F}^T$:

$$\tilde{\boldsymbol{F}} = (\boldsymbol{F}^T)^- = (\boldsymbol{F}\boldsymbol{F}^T)^{-1}\boldsymbol{F} = \frac{1}{3}\begin{bmatrix} 2 & -1 & 1 \\ 1 & 1 & 2 \end{bmatrix}$$

For a given vector $\boldsymbol{x} = [1, 2]^T$, we can find the coefficient vectors:

$$\boldsymbol{c} = \boldsymbol{F}^T\boldsymbol{x} = [1, 1, 2], \quad \boldsymbol{d} = \tilde{\boldsymbol{F}}^T\boldsymbol{x} = \frac{1}{3}[4, 1, 5]^T$$

We can verify that indeed $\boldsymbol{x}$ can be reconstructed from these coefficients:

$$\boldsymbol{x} = \sum_{n=1}^{3} c_n\tilde{\boldsymbol{f}}_n = \sum_{n=1}^{3} d_n\boldsymbol{f}_n = [1, 2]^T$$

The signal energy is $||\boldsymbol{x}||^2 = 5$, and the energy contained in the coefficients is $||\boldsymbol{c}||^2 = 6$ and $||\boldsymbol{d}||^2 = 14/3$, respectively, bounded by:

$$\lambda_{min}||\boldsymbol{x}||^2 = 5 < ||\boldsymbol{c}||^2 = 6 < \lambda_{max}||\boldsymbol{x}||^2 = 15$$

and

$$\frac{1}{\lambda_{max}}||\boldsymbol{x}||^2 = \frac{5}{3} < ||\boldsymbol{d}||^2 = \frac{14}{3} < \frac{1}{\lambda_{min}}||\boldsymbol{x}||^2 = \frac{15}{3}$$

Example 2.13: Consider a frame in $\mathbb{R}^2$ containing two vectors that form a frame matrix:

$$\boldsymbol{F} = [\boldsymbol{f}_1, \boldsymbol{f}_2] = \begin{bmatrix} 2 & -1 \\ 1 & -2 \end{bmatrix}$$

As $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$ are linearly independent, they form a Riesz basis. We have:

$$\boldsymbol{F}\boldsymbol{F}^T = \begin{bmatrix} 5 & 4 \\ 4 & 5 \end{bmatrix}$$

with eigenvalues $\lambda_1 = 1$, and $\lambda_2 = 9$, The dual frame is the pseudo-inverse of $\boldsymbol{F}^T$:

$$\tilde{\boldsymbol{F}} = (\boldsymbol{F}^T)^- = (\boldsymbol{F}\boldsymbol{F}^T)^{-1}\boldsymbol{F} = \frac{1}{3}\begin{bmatrix} 2 & 1 \\ -1 & -2 \end{bmatrix}$$

For a given vector $\boldsymbol{x} = [2, 3]^T$, we can find the coefficient vectors:

$$\boldsymbol{c} = \boldsymbol{F}^T\boldsymbol{x} = [7, -8]^{\cdot} \qquad \boldsymbol{d} = \tilde{\boldsymbol{F}}^T\boldsymbol{x} = \frac{1}{3}[1, -4]^T$$

We can verify that indeed $\boldsymbol{x}$ can be reconstructed from these coefficients:

$$\boldsymbol{x} = \sum_{n=1}^{3} c_n \tilde{\boldsymbol{f}}_n = \sum_{n=1}^{3} d_n \boldsymbol{f}_n = [2, 3]^T$$

The signal energy is $||\boldsymbol{x}||^2 = 13$, and the energy contained in the coefficients is $||\boldsymbol{c}||^2 = 113$ and $||\boldsymbol{d}||^2 = 17/9$, respectively, bounded by:

$$\lambda_{min}||\boldsymbol{x}||^2 = 13 < ||\boldsymbol{c}||^2 = 113 < \lambda_{max}||\boldsymbol{x}||^2 = 117$$

and

$$\frac{1}{\lambda_{max}}||\boldsymbol{x}||^2 = \frac{13}{9} < ||\boldsymbol{d}||^2 = \frac{17}{9} < \frac{1}{\lambda_{min}}||\boldsymbol{x}||^2 = \frac{117}{9}$$

In this case, we also have:

$$\boldsymbol{F}^*\tilde{\boldsymbol{F}} = \boldsymbol{I}$$

i.e., the two sets of mutually dual frame vectors are biorthogonal.

Example 2.14: Vectors $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$ form a basis that spans the 2-D space:

$$\boldsymbol{f}_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad \boldsymbol{f}_2 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \qquad \boldsymbol{f} = [\boldsymbol{f}_1, \boldsymbol{f}_2] = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$$

$$\boldsymbol{f}\boldsymbol{f}^T = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}, \qquad (\boldsymbol{f}\boldsymbol{f}^T)^{-1} = \begin{bmatrix} 1 & -1 \\ -1 & 2 \end{bmatrix}$$

The dual frame can be found to be:

$$\tilde{\boldsymbol{f}} = (\boldsymbol{f}\boldsymbol{f}^T)^{-1}\boldsymbol{f} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix} \quad \text{i.e.} \quad \tilde{\boldsymbol{f}}_1 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}, \quad \tilde{\boldsymbol{f}}_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Obviously the biorthogonality condition in Eq.2.287 is satisfied by these two sets of bases. Next, to represent a vector $\boldsymbol{x} = [0, 2]^T$ by each of the two bases, we find the coefficients as:

$$c_1 =< \boldsymbol{x}, \tilde{\boldsymbol{f}}_1 >= 2; \qquad c_1 =< \boldsymbol{x}, \tilde{\boldsymbol{f}}_2 >= -2$$
$$d_1 =< \boldsymbol{x}, \boldsymbol{f}_1 >= 0; \qquad d_2 =< \boldsymbol{x}, \boldsymbol{f}_2 >= -2$$

Now we have:

$$\boldsymbol{x} = c_1\boldsymbol{f}_1 + c_2\boldsymbol{f}_2 = 2\begin{bmatrix} 1 \\ 0 \end{bmatrix} - 2\begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -2 \end{bmatrix} \quad \textit{or} \quad \boldsymbol{x} = d_1\tilde{\boldsymbol{f}}_1 + d_2\tilde{\boldsymbol{f}}_2 = -2\begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -2 \end{bmatrix}$$

Example 2.15: Given a basis in $\mathbb{R}^3$:

$$\boldsymbol{f}_1 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \boldsymbol{f}_2 = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \quad \boldsymbol{f}_3 = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

Find its biorthogonal dual $\tilde{\boldsymbol{f}}_1$, $\tilde{\boldsymbol{f}}_2$, $\tilde{\boldsymbol{f}}_3$, and two sets of coefficients c_k and d_k $(k = 1, 2, 3)$ to represent a vector $\boldsymbol{x} = [1, 2, 3]^T$.

Solution: We need to find $\tilde{\boldsymbol{f}}_i$ that is orthogonal to all $\boldsymbol{f}_j$ except $i = j$ $(i, j = 1, 2, 3)$.

$$\tilde{\boldsymbol{f}} = (\boldsymbol{f}\boldsymbol{f}^T)^{-1}\boldsymbol{f} = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix}$$

$$\tilde{\boldsymbol{f}}_1 = \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}, \quad \tilde{\boldsymbol{f}}_2 = \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix}, \quad \tilde{\boldsymbol{f}}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Find coefficients c_i for $\boldsymbol{f}_i$ by $\boldsymbol{c} = \tilde{\boldsymbol{f}}^T\boldsymbol{x}$, i.e.,

$$c_1 =< \boldsymbol{x}, \tilde{\boldsymbol{f}}_1 >= -1, \quad c_2 =< \boldsymbol{x}, \tilde{\boldsymbol{f}}_1 >= -1, \quad c_3 =< \boldsymbol{x}, \tilde{\boldsymbol{f}}_1 >= 3$$

Find coefficients d_i for $\tilde{\boldsymbol{f}}_i$ by $\boldsymbol{d} = \boldsymbol{f}^T\boldsymbol{x}$, i.e.,

$$d_1 =< \boldsymbol{x}, \boldsymbol{f}_1 >= 1, \quad d_2 =< \boldsymbol{x}, \boldsymbol{f}_1 >= 3, \quad d_3 =< \boldsymbol{x}, \boldsymbol{f}_1 >= 6$$

Now $\boldsymbol{x}$ can be expressed as:

$$\boldsymbol{x} = \sum_{k=1}^{3} c_k\boldsymbol{f}_k = -\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} + 3\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

$$\boldsymbol{x} = \sum_{k=1}^{3} d_k \tilde{\boldsymbol{f}}_k = \begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix} + 3 \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix} + 6 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$$

We can further verify that

$$\sum_{k=1}^{3} < \boldsymbol{x}, \tilde{\boldsymbol{f}}_k >< \boldsymbol{x}, \boldsymbol{f}_k >= ||\boldsymbol{x}||^2 = 14$$

2.5 Kernel Function and Mercer's Theorem

Definition 2.24. *A kernel is a function K that maps two continuous variable t, τ to a complex value $K(t, \tau)$. If the two variables are sampled to become discrete t_m, t_n, then the kernel is $K(t_m, t_n) = K_{mn}$.*

Definition 2.25. *If $K(t, \tau) = \overline{K}(\tau, t)$ or $K_{mn} = \overline{K}_{nm}$, the kernel is Hermitian.*

Definition 2.26. *A kernel is positive definite if the following holds for any function $x(t)$ defined over $[a, b]$:*

$$\int \int_a^b x(t) K(t, \tau) x(\tau) d\tau \; dt > 0 \tag{2.298}$$

A kernel K_{mn} is positive definite if the following holds for any vector $\boldsymbol{x} = [x_1, \cdots, x_N]$:

$$\sum_{m=1}^{N} \sum_{n=1}^{N} x_m K_{mn} x_n > 0 \tag{2.299}$$

Definition 2.27. *Associated with a continuous kernel $K(t, \tau)$ an operator T_K can be defined as:*

$$T_K x(t) = \int_a^b K(t, \tau) x(\tau) d\tau = y(t) \tag{2.300}$$

Associated with a discrete kernel K_{mn} an operator T_K can be defined as a matrix:

$$\boldsymbol{T} = \begin{bmatrix} K_{11} & K_{12} & \cdots & K_{1N} \\ K_{21} & K_{22} & \cdots & K_{2N} \\ \vdots & \cdots & \ddots & \vdots \\ K_{N1} & K_{N2} & \cdots & K_{NN} \end{bmatrix} \tag{2.301}$$

which can be applied to a vector $\boldsymbol{x}$ *to generate:*

$$T_K\boldsymbol{x} = \boldsymbol{T}\boldsymbol{x} = \boldsymbol{y}, \quad \textit{or in component form:} \quad \sum_{m=1}^{N} K_{mn}x_m = y_n, \quad (n = 1, \cdots, N) \tag{2.302}$$

Theorem 2.13. *The operator* T_K *associated with a Hermitian kernel is self-adjoint.*

Proof: For operator T_K associated with a continuous kernel, we have:

$$< T_K x(t), y(t) >= \int_a^b T_K x(t)\ \overline{y}(t)dt = \int_a^b [\int_a^b K(t,\tau)x(\tau)d\tau]\ \overline{y}(t)dt$$
$$= \int_a^b [\int_a^b \overline{K}(\tau,t)\overline{y}(t)dt]\ x(\tau)d\tau = \int_a^b x(\tau)\ \overline{T_K y(\tau)}d\tau =< x(t), T_K y(t) > \tag{2.303}$$

For operator $T_K = \boldsymbol{T}$ associated with a discrete kernel, we have:

$$< \boldsymbol{T}\boldsymbol{x}, \boldsymbol{y} >= \sum_{n=1}^{N}\left[\sum_{m=1}^{N} K_{mn}x_m\right]\ \overline{y}_n = \sum_{m=1}^{N} x_m\left[\sum_{n=1}^{N}\overline{K}_{mn}\overline{y}_n\right] =< \boldsymbol{x}, \boldsymbol{T}\boldsymbol{y} > \tag{2.304}$$

As a self-adjoint operator T_K associated with a Hermitian kernel has all the properties of a self-adjoint operator stated in Theorem 2.4. Specifically, let λ_n be the nth eigenvalue of a self-adjoint operator T_K and $\phi_n(t)$ or $\boldsymbol{\phi}_n$ be the corresponding eigenfunction or eigenvector:

$$\int_a^b K(t,\tau)\phi_n(\tau)d\tau = \lambda_n\phi_n(t), \qquad \text{or} \qquad T_K\boldsymbol{\phi}_n = \boldsymbol{T}\boldsymbol{\phi}_n = \lambda_m\boldsymbol{\phi}_n \tag{2.305}$$

then the following statements are true:

1. All eigenvalues λ_n are real;
2. All eigenfunctions/eigenvectors are mutually orthogonal:
$$< \phi_m(t), \phi_n(t) >=< \boldsymbol{\phi}_m, \boldsymbol{\phi}_n >= \delta[m-n] \tag{2.306}$$
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* $K(t,\tau)$ *be a positive definite Hermitian kernel, and* λ_n *and* $\phi_n(t)$ $(n = 1, 2, \cdots)$ *be the nth eigenvalue and the corresponding eigenfunction of the associated operator* T_K*. Then the kernel can be expanded to become:*

$$K(t,\tau) = \sum_{n=1}^{\infty}\lambda_n\phi_n(t)\overline{\phi}_n(\tau) \tag{2.307}$$

The general proof of this theorem in Hilbert space is beyond the scope of this book and therefore omitted. However, we can prove the special case of the theorem in the N-D unitary space:

Theorem 2.15. *Let $K[m,n]$ be a positive definite Hermitian kernel, and λ_k and $\boldsymbol{\phi}_k$ $(k=1,2,\cdots)$ be the kth eigenvalue and the corresponding eigenvector of the associated operator $\boldsymbol{T}$. Then the kernel can be expanded to become:*

$$\boldsymbol{T}=\sum_{k=1}^{N}\lambda_k\boldsymbol{\phi}_k\overline{\boldsymbol{\phi}}_k \tag{2.308}$$

Proof: Since the discrete kernel is Hermitian: $K[m,n]=\overline{K}[n,m]$ for all $m,n=1,\cdots,N$, they form Hermitian matrix matrix $\boldsymbol{T}=\boldsymbol{T}^*$, which is a self-adjoint operator in $\mathbb{C}^N$:

$$<\boldsymbol{A}x,\boldsymbol{y}>=<\boldsymbol{x},\boldsymbol{A}y>$$

as previously shown in Eq.2.139. Also, we have shown in Eq.2.146 that this self-adjoint operator $\boldsymbol{T}$ can be expanded:

$$\boldsymbol{T}=\sum_{k=1}^{N}\lambda_k\boldsymbol{\phi}_k\boldsymbol{\phi}_k^* \tag{2.309}$$

i.e., the element in mth row and nth column of $\boldsymbol{T}$ is:

$$K[m,n]=\sum_{k=1}^{N}\lambda_k\phi_{mk}\overline{\phi}_{nk}\qquad(m,n=1,\cdots,N) \tag{2.310}$$

where ϕ_{mk} is the mth element of the mth eigenvector $\boldsymbol{\phi}_k$. This is the discrete and finite version of Mercer's theorem in an N-D unitary space. This completes the proof.

We see that the result in Eq.2.305 can be easily derived from Eq.2.307:

$$\begin{aligned}\int_a^b K(t,\tau)\phi_m(\tau)d\tau&=\int_a^b\left[\sum_{n=1}^{\infty}\lambda_n\phi_n(t)\overline{\phi}_n(\tau)\right]\ \phi_m(\tau)d\tau\\&=\sum_{n=1}^{\infty}\lambda_n\phi_n(t)\int_a^b\overline{\phi}_n(\tau)\phi_m(\tau)d\tau=\sum_{n=1}^{\infty}\lambda_n\phi_n(t)\delta[m-n]=\lambda_m\phi_m(t)\end{aligned} \tag{2.311}$$

Mercer's theorem will be used later in Chapter 8, and it also finds important applications in machine learning as the foundation of a type of methods called *kernel trick*.

As an example, consider a centered stochastic process $x(t)$ with $\mu_x(t)=0$ for all t. The covariance function is:

$$\sigma_x^2(t,\tau)=E[(x(t)-\mu_x(t))\ (\overline{x}(\tau)-\overline{\mu}_x(\tau))]=E[x(t)\overline{x}(\tau)]=\overline{E[x(\tau)\overline{x}(t)]}=\overline{\sigma}_x^2(\tau,t) \tag{2.312}$$

As this covariance function maps two variables t and τ to a real value, it can be considered as a kernel $K(t,\tau)=\sigma_x^2(t,\tau)$, which is symmetric and also positive definite, i.e., for any deterministic function $f(t)$, we have:

$$\int_a^b\int_a^b f(t)\sigma_x^2\overline{f}(\tau)\,dt\,d\tau=\int_a^b\int_a^b E[f(t)x(t)\,\overline{f}(\tau)\overline{x}(\tau)]\,dt\,d\tau$$
$$=E\int_a^b f(t)x(t)\,dt\int_a^b\overline{f}(\tau)\overline{x}(\tau)\,d\tau=E|\int_a^b f(t)x(t)\,dt|^2>0 \qquad (2.313)$$

According to theorems above, the integral operator T_K associated with this Hermitian kernel $K(t,\tau)$ is self-adjoint, its eigenequation can be written as:

$$T_k\phi_k(t)=\int_a^b\sigma_x^2(t,\tau)\phi_k(t)dt=\lambda_k\phi_k(t),\qquad k=1,2,\cdots \qquad (2.314)$$

where all eigenvalues $\lambda_k>0$ are real and positive, and the eigenfunctions $\phi_k(t)$ are orthogonal:

$$<\phi_m(t),\phi_n(t)>=\int_a^b\phi_m(t)\overline{\phi}_n(t)dt=\delta[m-n] \qquad (2.315)$$

If the stochastic process $x(t)$ is truncated and sampled, it become a random vector $\boldsymbol{x}=[x_1,\cdots,x_N]^T$. The covariance between any two components x_m and x_n is

$$\sigma_{mn}^2=E(x_m\overline{x}_n)=\overline{E(x_n\overline{x}_m)}=\overline{\sigma}_{nm}^2,\qquad (m,n=1,\cdots,N) \qquad (2.316)$$

which is a discrete Hermitian kernel, and the associated operator is the N by N covariance matrix of $\boldsymbol{x}$:

$$\boldsymbol{\Sigma}_x=E(\boldsymbol{x}\boldsymbol{x}^*)=\begin{bmatrix}\sigma_{11}^2 & \sigma_{12}^2 & \cdots & \sigma_{1N}^2\\ \sigma_{21}^2 & \sigma_{22}^2 & \cdots & \sigma_{2N}^2\\ \vdots & \cdots & \ddots & \vdots\\ \sigma_{N1}^2 & \sigma_{N2}^2 & \cdots & \sigma_{NN}^2\end{bmatrix} \qquad (2.317)$$

The eigenequation of this operator is

$$\boldsymbol{\Sigma}_x\boldsymbol{\phi}_n=\lambda_n\boldsymbol{\phi}_n,\qquad (n=1,\cdots,N) \qquad (2.318)$$

As $\boldsymbol{\Sigma}^*=\boldsymbol{\Sigma}$ is Hermitian (symmetric if $\boldsymbol{x}$ is real) and positive definite, its eigenvalues λ_k are all real positive, and the eigenvectors are orthogonal:

$$<\boldsymbol{\phi}_m,\boldsymbol{\phi}_n>=\boldsymbol{\phi}_m^T\overline{\boldsymbol{\phi}}_n=\delta[m-n],\qquad (m,n=1,\cdots,N) \qquad (2.319)$$

and they form a unitary matrix $\boldsymbol{\Phi}=[\boldsymbol{\phi}_1,\cdots,\boldsymbol{\phi}_N]$ satisfying $\boldsymbol{\Phi}^{-1}=\boldsymbol{\Phi}^*$ i.e., $\boldsymbol{\Phi}^*\boldsymbol{\Phi}=\boldsymbol{I}$. Eq.2.318 can also be written in the following forms:

$$\boldsymbol{\Sigma}_x\boldsymbol{\Phi}=\boldsymbol{\Phi}\boldsymbol{\Lambda},\qquad \boldsymbol{\Phi}^*\boldsymbol{\Sigma}_x\boldsymbol{\Phi}=\boldsymbol{\Lambda},\qquad \boldsymbol{\Sigma}_x=\boldsymbol{\Phi}\boldsymbol{\Lambda}\boldsymbol{\Phi}^*=\sum_{n=1}^N\lambda_n\boldsymbol{\phi}_n\boldsymbol{\phi}_n^* \qquad (2.320)$$

Theorem 2.16. *(Karhunen-Loeve Theorem) Let $x(t)$ be a centered stochastic process $x(t)$ with $\mu_x(t) = E[x(t)] = 0$, and λ_k and $\phi_k(t)$ be respectively the kth eigenvalue and the corresponding eigenfunction of the integral operator associated with the covariance $\sigma_x^2(t,\tau)$:*

$$T_K\phi_k(t) = \int_a^b \sigma_x^2(t,\tau)\phi_k(t)dt = \lambda_k\phi_k(t), \quad \textit{for all } k \tag{2.321}$$

Then $x(t)$ can be series expanded:

$$x(t) = \sum_{n=1}^{\infty} c_n\phi_n(t) \tag{2.322}$$

where the c_n is a random coefficient given by

$$c_n = \int_a^b x(t)\overline{\phi}_n(t)dt, \quad n = 1, 2, \cdots \tag{2.323}$$

which are centered (zero mean)

$$E(c_n) = 0 \tag{2.324}$$

and uncorrelated:

$$\sigma_{mn}^2 = Cov(c_m, c_n) = \lambda_m\ \delta[m-n], \quad \textit{i.e.,} \quad \sigma_n^2 = Var(c_n) = \lambda_n \tag{2.325}$$

Proof: We first show that Eq.2.323 can be obtained by taking an inner product with $\phi_m(t)$ on both sides of Eq.2.322:

$$< x(t), \phi_m(t) >= \int_a^b x(t)\overline{\phi}_n(t)dt = \sum_{n=1}^{\infty} c_n < \phi_n(t), \phi_m(t) >= \sum_{n=1}^{\infty} c_n\delta[m-n] = c_m \tag{2.326}$$

The expectation of this equation is indeed zero:

$$E(c_m) = E[\int_a^b x(t)\overline{\phi}_m(t)dt] = \int_a^b E[x(t)]\ \overline{\phi}_m(t)dt = 0 \tag{2.327}$$

Next we show that Eq.2.325 holds:

$$\begin{aligned}\sigma_{mn}^2 &= Cov(c_m, c_n) = E(c_m\overline{c}_n) = E[\int_a^b x(t)\overline{\phi}_m(t)dt \int_a^b \overline{x}(\tau)\phi_n(\tau)d\tau] \\ &= \int_a^b [\int_a^b \phi_n(\tau)E[x(t)\overline{x}(\tau)]d\tau]\overline{\phi}_m(t)dt = \int_a^b [\int_a^b \phi_n(\tau)\sigma_x^2(t,\tau)d\tau]\ \overline{\phi}_m(t)dt \\ &= \int_a^b \lambda_n\phi_n(t)\overline{\phi}_m(t)dt = \lambda_n\int_a^b \phi_l(t)\overline{\phi}_m(t)dt = \lambda_m\ \delta[m-n]\end{aligned} \tag{2.328}$$

This completes the proof.

When the centered stochastic process $x(t)$ is truncated and sampled to become a finite random vector $\boldsymbol{x} = [x_1, \cdots, x_N]^T$ with $E(\boldsymbol{x}) = \boldsymbol{\mu}_x = 0$, the Karhunen-Loeve theorem takes a discrete form. Given Eq.2.318, $\boldsymbol{x}$ can be series expanded

to become a linear combination of the eigenvectors $\boldsymbol{\phi}_n$ of its covariance matrix $\boldsymbol{\Sigma}_x$:

$$\boldsymbol{x} = \sum_{n=1}^{N} c_n \boldsymbol{\phi}_n = \boldsymbol{\Phi}\boldsymbol{c} \tag{2.329}$$

where $\boldsymbol{c} = [c_1, \cdots, c_N]^T$ is a random vector formed by the N coefficients. To obtain these coefficients, we pre-multiply both sides by $\boldsymbol{\Phi}^{-1} = \boldsymbol{\Phi}^*$ to get:

$$\boldsymbol{\Phi}^* \boldsymbol{x} = \boldsymbol{c} \tag{2.330}$$

i.e.,

$$c_n = < \boldsymbol{x}, \boldsymbol{\phi}_n > = \boldsymbol{\phi}_n^* \boldsymbol{x}, \quad (n = 1, \cdots, N) \tag{2.331}$$

The mean vector of $\boldsymbol{c}$ is zero:

$$\boldsymbol{\mu}_c = E(\boldsymbol{c}) = E(\boldsymbol{\Phi}^* \boldsymbol{x}) = \boldsymbol{\Phi}^* E(\boldsymbol{x}) = \boldsymbol{0} \tag{2.332}$$

and the covariance matrix of $\boldsymbol{c}$ is:

$$\begin{aligned} \boldsymbol{\Sigma}_c &= E(\boldsymbol{c}\boldsymbol{c}^*) = E[(\boldsymbol{\Phi}^* \boldsymbol{x})(\boldsymbol{\Phi}^* \boldsymbol{x})^*] = E[\boldsymbol{\Phi}^* \boldsymbol{x}\boldsymbol{x}^* \boldsymbol{\Phi}] \\ &= \boldsymbol{\Phi}^* E(\boldsymbol{x}\boldsymbol{x}^*)\boldsymbol{\Phi} = \boldsymbol{\Phi}^* \boldsymbol{\Sigma}_x \boldsymbol{\Phi} = \boldsymbol{\Lambda} \end{aligned} \tag{2.333}$$

The last equal sign is due to Eq.2.320. The covariance matrix $\boldsymbol{\Sigma}_c = \boldsymbol{\Lambda}$ is diagonalized:

$$\sigma_{mn}^2 = \lambda_n \delta[m - n], \quad (m, n = 1, \cdots, N) \tag{2.334}$$

We see that the variance σ_n^2 of the nth coefficient c_n is the nth eigenvalue λ_n corresponding to the nth eigenvector $\boldsymbol{\phi}_n$, and the random signal $\boldsymbol{x}$ is decorrelated by the transformation $\boldsymbol{c} = \boldsymbol{\Phi}^* \boldsymbol{x}$ in Eq.2.330, as the components c_m and c_n of the resulting random signal $\boldsymbol{c}$ are no longer correlated ($\sigma_{mn}^2 = 0$).

Comparing the generalized Fourier expansion in Eqs.2.114 and 2.115 with this Karhunen-Loeve series expansion in Eqs.2.322 and 2.323, 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 $\phi_n(t)$, while the latter is for a stochastic signal, and the basis functions $\phi_n(t)$, as the eigenfunctions of the integral operator associated with the covariance function of the stochastic process, are completely dependent on the specific signal being considered. Eqs.2.329 and 2.330 are simply the discrete version of Eqs.2.322 and 2.323, which correspond to the discrete version of the Fourier transform, as we will see later.

2.6 Summary

Let us summarize the most essential points discussed so far. These will appear repeatedly in the following chapters during the specific discussion of various orthogonal transform methods.

- A time signal can be considered as a vector $\boldsymbol{x} \in H$ in a Hilbert space. Specifically, a continuous signal $x(t)$ over time interval $a \leq t \leq b$ is a vector $\boldsymbol{x} = x(t)$ in a function space; and its discrete samples $\cdots, x[n-1], x[n], x[n+1], \cdots$ is a vector $\boldsymbol{x} = [\cdots, x[n], \cdots]^T$ in an N-D unitary space $\mathbb{C}^N$.
- Under the basis $\{\boldsymbol{b}_n\}$ that span the space H, a signal vector can be represented by the following expansion:

$$\boldsymbol{x} = \sum_n c_n \boldsymbol{b}_n = \sum_n < \boldsymbol{x}, \boldsymbol{b}_n > \boldsymbol{b}_n \tag{2.335}$$

  where $c_n =< \boldsymbol{x}, \boldsymbol{b}_n >$ is the decomposition or analysis of the signal by which the signal is decomposed into a set of components $c_n \boldsymbol{b}_n$, and $\boldsymbol{x} = \sum_n c_n \boldsymbol{b}_n$ is the reconstruction or synthesis of the signal by which the signal is reconstructed by its components.
- The representation of the signal vector depends on the specific basis used. A signal typically given in its original form $x(t)$ or $\boldsymbol{x} = [\cdots, x[n], \cdots]^T$ is represented as a sequence of weighted and shifted time impulses (Eqs.1.3 and 1.6). Specifically, a continuous signal is expressed as:

$$x(t) = \int x(\tau)\delta(t-\tau)d\tau, \quad \text{(for all } t\text{)} \tag{2.336}$$

  while a discrete signal is expressed as:

$$\boldsymbol{x} = \sum_n x[n]\boldsymbol{e}_n, \quad \text{or} \quad x[m] = \sum_n x[n]e_{mn} = \sum_n x[n]\delta[m-n], \quad \text{(for all } m\text{)} \tag{2.337}$$

  where $\delta(t-\tau)$ or $e_{mn} = \delta[m-n]$ can be considered as the standard basis which is always implicitly used to represent a time signal. In other words, a signal $x(t)$ or $x[n]$ is always given as a set of coefficients, or weights, for the standard basis.
- Alternatively, the same signal can also be represented under a different orthonormal basis obtained by some unitary transformation or rotation of the standard basis. For a continuous signal $x(t)$, we have:

$$\begin{aligned} x(t) &= \int c(f)\phi_f(t)\, df, \quad \text{(for all } t\text{)} \\ c(f) &= < x(t), \phi_f(t) >= \int x(t)\overline{\phi}_f(t)\, dt, \quad \text{(for all } f\text{)} \end{aligned} \tag{2.338}$$

  The first equation expresses the signal function $x(t)$ 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 $\phi_f(t)$, which is also called the *kernel function* of the transform.

Similarly, for a discrete signal $\boldsymbol{x} = [\cdots, x[n], \cdots]^T$, we have

$$\boldsymbol{x} = \sum_m c_m \boldsymbol{b}_m, \quad \text{or} \quad x[n] = \sum_m c_m b_{nm}, \quad \text{(for all n)}$$
$$c_m = <\boldsymbol{x}, \boldsymbol{b}_m> = \sum_n x[n]\overline{b}_{nm}, \quad \text{(for all } m\text{)} \tag{2.339}$$

where $x[n]$ is the nth element of $\boldsymbol{x}$, and b_{nm} is the nth element of the mth basis vector $\boldsymbol{b}_m$. The first equation expresses the signal vector as a linear combination of a set of countable basis vectors $\boldsymbol{b}_m$ (or in component form b_{nm}) for all m. The second equation gives the mth coefficient c_m as the projection of the signal $\boldsymbol{x}$ onto the corresponding basis vector $\boldsymbol{b}_m$.

Both of the two pairs of equations above are unitary (orthogonal if real) transformations, which could be either continuous or discrete. 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 with respect to 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.

- A signal vector in its vector space can be represented in many different ways, each corresponding to one of the infinitely many possible orthogonal bases all spanning the same vector space. All these representations of the signal 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, which preserves the norm of the vector according to the Parseval's equality.
- The unitary transformation, a rotation, of a basis will result in another basis, which needs a different set of coefficients for the representation of a given signal. In particular, the signal originally given in the implicit standard basis also corresponds to a special unitary transformation, the identity transform. In an alternative but equivalent view, the signal vector itself, originally given under the standard basis, can be rotated differently according to the transform method used. The central theme of the book is to study the various orthogonal transform methods, each corresponding to a different representation of the same signal. Also an issue of interest is how to find the "optimal" transform among all possible transforms according to some criteria.
- The topics of interest in the future discussion include: why such an unitary transformation is desirable to start with, and how to find a particular basis most suitable for a specific task, so that the signal is represented in such a way that it can be most effectively and conveniently processed, analyzed, com-

pressed for transmission and storage, and the information of interest extracted. These issues will be specifically discussed in the rest of the book.

- We will discuss mostly orthogonal transforms based on orthogonal basis vector or functions. The inner product of any two such basis vectors or functions is zero, indicating they each carry some independent information. However, sometimes certain non-orthogonal basis functions will also be considered, such as in the discussion of various wavelet transforms. In such cases, the inner product of two basis functions may not be zero, i.e., one is correlated with the other. In other words, there exists some redundancy in terms of the information they each carry. Although such redundancy is obviously a drawback in terms of data compression, it is not always necessarily bad in terms of reconstruction of signals when noise is present.

2.7 Problems

1. Approximate a given 3-D vector $\boldsymbol{x} = [1, 2, 3]^T$ in an 2-D subspace spanned by the two standard basis vectors $\boldsymbol{e}_1 = [1, 0, 0]^T$ and $\boldsymbol{e}_2 = [0, 1, 0]^T$. Of course this approximation is trivial as the vector is originally given in terms of the three standard basis vectors $\boldsymbol{e}_1$, $\boldsymbol{e}_2$ and $\boldsymbol{e}_3$. Now the two coefficients are simply the projections of the vector $\boldsymbol{x}$ onto each of the two standard basis vectors of the 2-D subspace:

$$c_1 =< \boldsymbol{x}, \boldsymbol{e}_1 >= 1, \qquad c_1 =< \boldsymbol{x}, \boldsymbol{e}_2 >= 2$$

The approximation is simply

$$\hat{\boldsymbol{x}} = c_1\boldsymbol{e}_1 + c_2\boldsymbol{e}_2 = [1, 2, 0]^T$$

and the error vector is

$$\tilde{\boldsymbol{x}} = \boldsymbol{x} - \hat{\boldsymbol{x}} = [1, 2, 3]^T - [1, 2, 0]^T = [0, 0, 3]^T$$

which is of course orthogonal to both $\boldsymbol{e}_1 = [1, 0, 0]^T$ and $\boldsymbol{e}_1 = [0, 1, 0]^T$.
Next, we use two different basis vectors to span a 2-D subspace:

$$\boldsymbol{a}_1 = [1, 0, -1]^T, \qquad \boldsymbol{a}_2 = [-1, 2, 0]^T$$

We need to find a vector in this 2-D subspace

$$\hat{\boldsymbol{x}} = c_1\boldsymbol{a}_1 + c_2\boldsymbol{a}_2$$

so that the error $||\boldsymbol{x} - \hat{\boldsymbol{x}}||$ is minimized.
According to the projection theorem, to reach minimum error, the error vector $\boldsymbol{x} - \hat{\boldsymbol{x}}$ has to be orthogonal to the basis functions $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ of the 2-D subspace:

$$< \boldsymbol{x} - \hat{\boldsymbol{x}}, \boldsymbol{a}_k >=< \boldsymbol{x} - c_1\boldsymbol{a}_1 - c_2\boldsymbol{a}_2, \boldsymbol{a}_k >= 0, \qquad (k = 1, 2)$$

i.e.,

$$\begin{cases} c_1 < \boldsymbol{a}_1, \boldsymbol{a}_1 > + c_2 < \boldsymbol{a}_1, \boldsymbol{a}_2 > = < \boldsymbol{x}, \boldsymbol{a}_1 > \\ c_1 < \boldsymbol{a}_1, \boldsymbol{a}_2 > + c_2 < \boldsymbol{a}_2, \boldsymbol{a}_1 > = < \boldsymbol{x}, \boldsymbol{a}_2 > \end{cases}$$

This equation system can be expressed in matrix form:

$$\begin{bmatrix} < \boldsymbol{a}_1, \boldsymbol{a}_1 > & < \boldsymbol{a}_2, \boldsymbol{a}_1 > \\ < \boldsymbol{a}_1, \boldsymbol{a}_2 > & < \boldsymbol{a}_2, \boldsymbol{a}_2 > \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 2 & -1 \\ -1 & 5 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} < \boldsymbol{x}, \boldsymbol{a}_1 > \\ < \boldsymbol{x}, \boldsymbol{a}_2 > \end{bmatrix} = \begin{bmatrix} -2 \\ 3 \end{bmatrix}$$

Solving this equation system we get $c_1 = -7/9$ and $c_2 = 4/9$. Of course we could also directly use the pseudo inverse method to get the same results:

$$\begin{aligned} \boldsymbol{c} &= (\boldsymbol{a}^T \boldsymbol{a})^{-1} \boldsymbol{a}^T \boldsymbol{x} \\ &= \left(\begin{bmatrix} 1 & 0 & -1 \\ -1 & 2 & 0 \end{bmatrix} \begin{bmatrix} 1 & -1 \\ 0 & 2 \\ -1 & 0 \end{bmatrix} \right)^{-1} \begin{bmatrix} 1 & 0 & -1 \\ -1 & 2 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \frac{1}{9} \begin{bmatrix} -7 \\ 4 \end{bmatrix} \end{aligned}$$

Having found c_1 and c_2, we further get

$$\hat{\boldsymbol{x}} = c_1 \boldsymbol{a}_1 + c_2 \boldsymbol{a}_2 = -\frac{7}{9} \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} + \frac{4}{9} \begin{bmatrix} -1 \\ 2 \\ 0 \end{bmatrix} = \frac{1}{9} \begin{bmatrix} -11 \\ 8 \\ 7 \end{bmatrix}$$

The error vector is:

$$\tilde{\boldsymbol{x}} = \boldsymbol{x} - \hat{\boldsymbol{x}} = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} - \frac{1}{9} \begin{bmatrix} -11 \\ 8 \\ 7 \end{bmatrix} = \frac{1}{9} \begin{bmatrix} 20 \\ 10 \\ 20 \end{bmatrix}$$

which is orthogonal to the 2-D plane spanned by $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$:

$$< \tilde{\boldsymbol{x}}, \boldsymbol{a}_1 > = < \tilde{\boldsymbol{x}}, \boldsymbol{a}_2 > = 0$$

indicating that $\hat{\boldsymbol{x}}$ is indeed the optimal approximation of $\boldsymbol{x}$ in the 2-D subspace.

In particular, if two orthonormal basis vectors are used to span the 2-D subspace, then the off-diagonal elements of the 2 by 2 matrix above are zero, and all elements on the main diagonal are one, and consequently the coefficients c_1 and c_2 can be much more conveniently obtained the as length of the projections of $\boldsymbol{x}$ onto the two basis vectors, as can be illustrated in the following example.

2. Use the Gram-Schmidt orthogonalization process to construct two new orthonormal basis vectors $\boldsymbol{b}_1$ and $\boldsymbol{b}_2$ from the two old vectors $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ used in the previous example, so that they span the same 2-D space, and then re-approximate the vector $\boldsymbol{x} = [1, 2, 3]^T$ above. Now the coefficients c_1 and c_2 can be easily found without solving a linear equation system. This problem is left for the reader as an exercise.

First, let $\boldsymbol{o}_1 = \boldsymbol{a}_1$. Second, let

$$\boldsymbol{o}_2 = \boldsymbol{a}_2 - P_{\boldsymbol{o}_1}(\boldsymbol{a}_2) = \boldsymbol{a}_2 - \frac{<\boldsymbol{a}_1, \boldsymbol{a}_2>}{<\boldsymbol{a}_1, \boldsymbol{a}_1>}\boldsymbol{a}_1 = \begin{bmatrix} -1 \\ 2 \\ 0 \end{bmatrix} + \begin{bmatrix} 0.5 \\ 0 \\ -0.5 \end{bmatrix} = \begin{bmatrix} -0.5 \\ 2 \\ -0.5 \end{bmatrix}$$

The new basis vectors are indeed orthogonal: $<\boldsymbol{o}_1, \boldsymbol{o}_2>= 0$ and the coefficients can be found to be:

$$c_1 = \frac{<\boldsymbol{x}, \boldsymbol{o}_1>}{<\boldsymbol{o}_1, \boldsymbol{o}_1>} = -1$$
$$c_2 = \frac{<\boldsymbol{x}, \boldsymbol{o}_2>}{<\boldsymbol{o}_2, \boldsymbol{o}_2>} = 4/9$$

and the approximation is

$$\hat{\boldsymbol{x}} = c_1\boldsymbol{a}_1 + c_2\boldsymbol{a}_2 = -\begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} + \frac{4}{9}\begin{bmatrix} -0.5 \\ 2 \\ -0.5 \end{bmatrix} = \frac{1}{9}\begin{bmatrix} -11 \\ 8 \\ 7 \end{bmatrix}$$

which is the same as what we got using $\boldsymbol{a}_1$ and $\boldsymbol{a}_2$ before.

3. 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 $a_1(t)$:

$$a_1(t) = 1, \quad a_2(t) = \begin{cases} 0 \ (0 \le t < 1/2) \\ 1 \ (1/2 \le t < 1) \end{cases}$$

As $< a_1(t), a_2(t) >= \int a_1(t)a_2(t)dt \neq 0$, the basis functions are not orthogonal and we have to solve a linear equation system to find the coefficients c_1 and c_2:

$$\begin{bmatrix} < a_1(t), a_1(t) > & < a_2(t), a_1(t) > \\ < a_1(t), a_2(t) > & < a_2(t), a_2(t) > \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} < x(t), a_1(t) > \\ < x(t), a_2(t) > \end{bmatrix}$$

All six inner products can be easily found by carrying out the following six integrals:

$$\int_0^1 a_1^2(t)dt = 1, \quad \int_0^1 a_1(t)a_2(t)dt = \int_0^1 a_2^2(t)dt = 1/2,$$
$$\int_0^1 x(t)a_1(t)dt = \frac{1}{3}, \quad \int_0^1 x(t)a_2(t)dt = \frac{7}{24}$$

Now we have

$$\begin{bmatrix} 1 & 1/2 \\ 1/2 & 1/2 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 1/3 \\ 7/24 \end{bmatrix}$$

Solving this we get $c_1 = 1/12$ and $c_2 = 1/2$, and

$$\hat{x}(t) = c_1a_1(t) + c_2a_2(t) = \frac{1}{12}a_1(t) + \frac{1}{2}a_2(t) = \begin{cases} 1/12 \ 0 \le t < 1/5 \\ 7/12 \ 1/5 \le 1 \end{cases}$$

This result is shown in the plot on the left in Fig. 2.13.

4. If we redefine the second basis function $a_2(t)$ as

$$a_2(t) = \begin{cases} -1 \ (0 \le t < 1/2) \\ 1 \ (1/2 \le t < 1) \end{cases}$$

it becomes orthogonal to the first function $a_1(t)$:

$$< a_1(t), a_2(t) >= \int_0^1 a_1(t)a_2(t)dt = 0$$

and they are actually the first two basis function of an orthogonal Walsh-Hadamard transform (WHT) to be discussed in details later. we can find c_1' and c_2' from this equation system

$$\begin{bmatrix} < a_1(t), a_1(t) > & 0 \\ 0 & < a_2(t), a_2(t) > \end{bmatrix} \begin{bmatrix} c_1' \\ c_2' \end{bmatrix} = \begin{bmatrix} < x(t), a_1(t) > \\ < x(t), a_2(t) > \end{bmatrix}$$

Note that all off-diagonal elements of the matrix equal to zero due to the fact that the two basis functions are orthogonal. Moreover, as $a_2(t)$ is also normalized as well as $a_1(t)$:

$$\int_0^1 a_2(t)a_2(t)dt = \int_0^1 a_1(t)a_1(t)dt = 1$$

all elements along the mail diagonal are 1, i.e., the two coefficients can be directly obtained as:

$$c_1' = < x(t), a_1(t) >= \int_0^1 x(t)a_1(t)dt = \frac{1}{3}$$
$$c_2' = < x(t), a_2(t) >= \int_0^1 x(t)a_2(t)dt = \frac{1}{4}$$

and the estimated signal is exactly the same as before:

$$\hat{x}(t) = \frac{1}{3}a_1(t) + \frac{1}{4}a_2(t) = \begin{cases} 1/12 \ 0 \le t < 1/5 \\ 7/12 \ 1/5 \le 1 \end{cases}$$

5. Based on the previous example, we add one more basis function defined as:

$$a_3(t) = \begin{cases} 1 \quad (0 \le t < 1/4) \\ -1 \ (1/4 \le t < 3/4) \\ 1 \quad (3/4 \le t < 1) \end{cases}$$

so that the 2-D space is expanded to a 3-D space spanned by $a_1(t)$, $a_2(t)$ and $a_3(t)$, which are actually the first three basis functions of the Walsh-Hadamard transform. In general, when adding a new basis function, the coefficients for the previous basis function may need to be recalculated. However, in this case the three basis function are orthonormal:

$$< a_m(t), a_n(t) >= \int_0^1 a_m(t)a_n(t)dt = \delta[m-n]$$

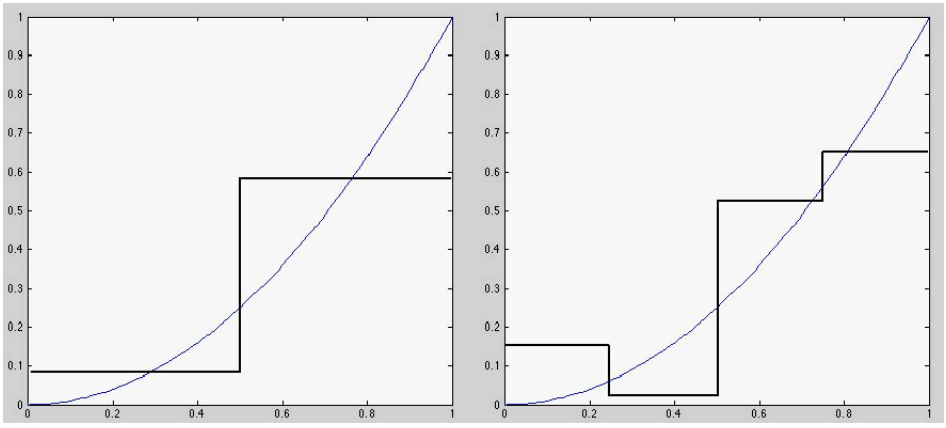

Figure 2.13 Approximation of $x(t) = t^2$ in 2-D (left) and 3-D (right)

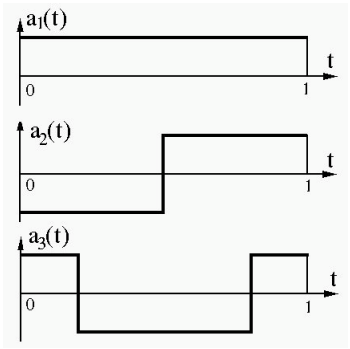

Figure 2.14 First three basis functions of WHT

the coefficients $c_1 = 1/3$ and $c_2 = 1/4$ obtained previously are still valid, and the computation for the coefficient c_3 can be carried out independently:

$$c_3 = \int x(t)a_3(t)dt = \int_0^{1/4} t^2 dt - \int_{1/4}^{3/4} t^2(t)dt + \int_{3/4}^{1} t^2(t)dt = \frac{1}{16}$$

and the optimal approximation becomes:

$$\hat{x}(t) = \frac{1}{3}a_1(t) + \frac{1}{4}a_2(t) + \frac{1}{16}a_3(t)$$

This result is shown in the plot on the right in Fig. 2.13. Also, the first three basis functions of the Walsh-Hadamard transform used here are shown in Fig. 2.14.

6. Approximate the same function $x(t) = t^2$ above in a 3-D space spanned by three basis functions , defined over the same time period $(0 \leq t < 1)$:

$$a_1(t) = \cos(0\pi t) = 1; \quad a_2(t) = \sqrt{2}\cos(\pi t); \quad a_2(t) = \sqrt{2}\sin(2\pi t)$$

We leave his problem to the reader as an exercise.

7. Approximate a function $x(t) = t^2$ defined over an interval $[0, 1]$ in a 2-D subspace spanned by two basis functions $a_0(t)$ and $a_1(t)$:

$$\hat{x}(t) = c_0 a_0(t) + c_1 a_1(t)$$

a. Find the coefficients c_0 and c_1 for these basis functions:

$$a_0(t) = 1, \quad a_1(t) = \begin{cases} 0 \ (0 \le t < 1/2) \\ 1 \ (1/2 \le t < 1) \end{cases}$$

Solution:
To find the coefficients c_0 and c_1, we need to solve a linear equation system:

$$\begin{bmatrix} < x(t), a_0(t) > \\ < x(t), a_1(t) > \end{bmatrix} = \begin{bmatrix} < a_0(t), a_0(t) > & < a_1(t), a_0(t) > \\ < a_0(t), a_1(t) > & < a_1(t), a_1(t) > \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \end{bmatrix}$$

All six inner products can be easily found to be:

$$< x(t), a_0(t) > = \int_0^1 x(t) a_0(t) dt = \frac{1}{3}$$
$$< x(t), a_1(t) > = \int_0^1 x(t) a_1(t) dt = \frac{7}{24}$$
$$< a_0(t), a_0(t) > = \int_0^1 a_1^2(t) dt = 1$$
$$< a_0(t), a_1(t) > = < a_1(t), a_0(t) > = < a_1(t), a_1(t) > = \frac{1}{2}$$

Note that $< a_0(t), a_1(t) > \neq 0$, i.e., they are not orthogonal. Now we have

$$\begin{bmatrix} 1/3 \\ 7/24 \end{bmatrix} = \begin{bmatrix} 1 & 1/2 \\ 1/2 & 1/2 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \end{bmatrix}$$

Solving this we get $c_0 = 1/12$ and $c_1 = 1/2$, and

$$\hat{x}(t) = \frac{1}{12} a_0(t) + \frac{1}{2} a_1(t) = \begin{cases} 1/12 \ 0 \le t < 1/2 \\ 7/12 \ 1/2 \le t < 1 \end{cases}$$

b. Find the coefficients c_0 and c_1 for these basis functions:

$$a_0(t) = 1, \quad a_1(t) = \begin{cases} 1 \ (0 \le t < 1/2) \\ -1 \ (1/2 \le t < 1) \end{cases}$$

Compare the results with the first part.
Solution:
Note that they are orthogonal

$$< a_0(t), a_1(t) > = \int_0^1 a_0(t) a_1(t) dt = 0$$

The given function can now be approximated as

$$\hat{x}(t) = c_0 a_0(t) + c_1 a_1(t)$$

and c_0 and c_1 can be obtained from this equation system

$$\begin{bmatrix} < x(t), a_0(t) > \\ < x(t), a_1(t) > \end{bmatrix} = \begin{bmatrix} < a_0(t), a_0(t) > & 0 \\ 0 & < a_1(t), a_1(t) > \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \end{bmatrix}$$

With all the off-diagonal elements of the matrix equal to zero, this equation system can be easily solved to get:

$$c_0 = \frac{< x(t), a_0(t) >}{< a_0(t), a_0(t) >} = \frac{\int_0^1 x(t)_1(t)dt}{\int_0^1 a_1(t)a_1(t)dt} = \frac{1}{3}$$

$$c_1 = \frac{< x(t), a_1(t) >}{< a_1(t), a_1(t) >} = \frac{\int_0^1 x(t)a_1(t)dt}{\int_0^1 a_1(t)a_1(t)dt} = -\frac{1}{4}$$

and

$$\hat{x}(t) = \frac{1}{3}a_0(t) - \frac{1}{4}a_1(t) = \begin{cases} 1/12 & 0 \le t < 1/2 \\ 7/12 & 1/2 \le t < 1 \end{cases}$$

In particular, the coefficient $c_0 = 1/3$ for the first basis function $a_0(t) = 1$ is the average or the DC component of the signal $x(t) = t^2$, i.e.,

$$c_0 = \frac{1}{T}\int_T x(t)dt$$

c. In order to better approximate the given function $x(t) = t^2$, a third basis function $a_2(t)$ is included in addition to the two basis functions $a_0(t)$ and $a_1(t)$ used in the previous part:

$$a_2(t) = \begin{cases} 0 & (0 \le t < 1/2) \\ 1 & (1/2 \le t < 3/4) \\ -1 & (3/4 \le t < 1) \end{cases}$$

These three basis functions are three of the first four Haar transform basis functions. Find the coefficient c_0, c_1 and c_2 to optimally approximate $x(t)$.
Solution: As $a_2(t)$ is orthogonal to both $a_0(t)$ and $a_1(t)$, we can find its coefficient c_2 independent of c_0 and c_1 obtained previously:

$$c_2 = \frac{< x(t), a_2(t) >}{< a_2(t), a_2(t) >} = \frac{\int_0^1 x(t)a_2(t)dt}{\int_0^1 a_2(t)a_2(t)dt} = -\frac{3}{16}$$

Now we have:

$$\hat{x}(t) = \frac{1}{3}a_0(t) - \frac{1}{4}a_1(t) - \frac{3}{16}a_2(t) = \begin{cases} 1/12 & 0 \le t < 1/2 \\ 19/48 & 1/2 \le t < 3/4 \\ 37/48 & 3/4 \le t < 1 \end{cases}$$

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:

$$\int x^2 \cos(ax)dx = \frac{2x\cos(ax)}{a^2} + \frac{a^2x^2-2}{a^3}\sin(ax) + C$$

Solution:
First realize that these basis functions are orthonormal $< a_i(t), a_j(t) >= \delta[i-j]$, therefore we simply have

$$c_1 = \int x(t)a_1(t)dt = \int_0^1 t^2 dt = 1/3$$
$$c_2 = \int x(t)a_2(t)dt = \sqrt{2}\int_0^1 t^2\cos(\pi t)dt = -\sqrt{8}/\pi^2 \approx -0.29$$
$$c_3 = \int x(t)a_3(t)dt = \sqrt{2}\int_0^1 t^2\cos(2\pi t)dt = 1/2\pi^2 = 0.05$$

9. In Example 2.10 we approximated the temperature signal, a 8-D vector $\boldsymbol{x} = [65, 60, 65, 70, 75, 80, 75, 70]^T$, 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):

$$\boldsymbol{H}_w = \frac{1}{\sqrt{8}}\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix}$$

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 $k = 1, 2, \cdots, 8$, 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.

10. The same temperature signal in Example 2.10 $\boldsymbol{x} = [65, 60, 65, 70, 75, 80, 75, 70]^T$ can also be approximated using a set of different basis vectors obtained by sampling the following cosine functions:

$$a_0(t) = 1, \quad a_1(t) = \sqrt{2}\cos(\pi t), \quad a_2(t) = \sqrt{2}\cos(2\pi t)$$

at 8 equally points $n_k = 1/16 + n/8 = 0.0625 + n \times 0.125$, $(n = 1, 2, \cdots, 8)$. 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 ($k = 1, 2, \cdots, 8$), and plot the original signal together with the approximation for each case. Use a software tool such as Matlab.
Solution:

$$\begin{aligned}
\boldsymbol{a}_1 &= [1,1,1,1,1,1,1,1]^T/8, \\
\boldsymbol{a}_2 &= [1.387, 1.176, 0.786, 0.276, -0.276, -0.786, -1.176, -1.287]^T/8, \\
\boldsymbol{a}_3 &= [1.307, 0.541, -0.541, -1.307, -1.307, -0.541, 0.541, 1.307]^T/8
\end{aligned}$$

$$c_1 =< \boldsymbol{x}, \boldsymbol{b}_1 >= 70; \quad c_1 =< \boldsymbol{x}, \boldsymbol{b}_2 >= -4.72; \quad c_1 =< \boldsymbol{x}, \boldsymbol{b}_3 >= -2.31$$

c_1 is the average temperature, $c_2 = -4.72$ indicates morning temperature is 4.71 degrees lower than afternoon, and c_3 indicates night temperature is 2.31 degrees lower than day time temperature.

3 Continuous-Time Fourier Transform

3.1 The Fourier Series Expansion of Periodic Signals

3.1.1 Formulation of The Fourier Expansion

As we have already seen in the previous chapter, the second-order differential operator D^2 over the interval $[0, T]$ is a self-adjoint operator, and its eigenfunctions $\phi_k(t) = e^{j2k\pi f_0 t}/\sqrt{T}$ $(k = 0, \pm 1, \pm 2, \cdots)$ are orthonormal (Eq.1.27):

$$< \phi_m(t), \phi_n(t) >= \frac{1}{T}\int_T e^{j2m\pi f_0 t} e^{-j2n\pi f_0 t} dt = \frac{1}{T}\int_T e^{j2(m-n)\pi f_0 t} dt = \delta[m-n] \tag{3.1}$$

where $\omega_0 = 2\pi f_0 = 2\pi/T$. These eigenfunctions form a complete orthogonal system that spans a function space over interval $[0, T]$, and any periodic signal $x_T(t) = x_T(t+T)$ in the space can be expressed as a linear combination of these basis functions:

$$x_T(t) = \sum_{k=-\infty}^{\infty} X[k]\phi_k(t) = \frac{1}{\sqrt{T}}\sum_{k=-\infty}^{\infty} X[k] e^{j2k\pi f_0 t} \tag{3.2}$$

Note that at $t = 0$ 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. 2.149 is guaranteed. Consequently at the end points $t = 0$ and $t = T$ the reconstructed signal 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 nth coefficient $X[n]$ can be found by taking an inner product with $\phi_n(t) = e^{j2n\pi f_0 t}/\sqrt{T}$ on both sides of the equation above:

$$\begin{aligned} < x_T(t), \phi_n(t) > &= < x_T(t), e^{j2n\pi f_0 t}/\sqrt{T} >= \frac{1}{T}\sum_{k=0}^{\infty} X[k] < e^{j2k\pi f_0 t}, e^{j2n\pi f_0 t} > \\ &= \sum_{k=-\infty}^{\infty} X[k]\delta[k-n] = X[n] \end{aligned} \tag{3.3}$$

i.e., the nth coefficient $X[n]$ is the projection of function $x_T(t)$ onto the nth basis function $\phi_n(t)$:

$$X[n] =< x_T(t), \phi_n(t) >= \frac{1}{\sqrt{T}}\int_T x_T(t) e^{-j2n\pi f_0 t} dt \tag{3.4}$$

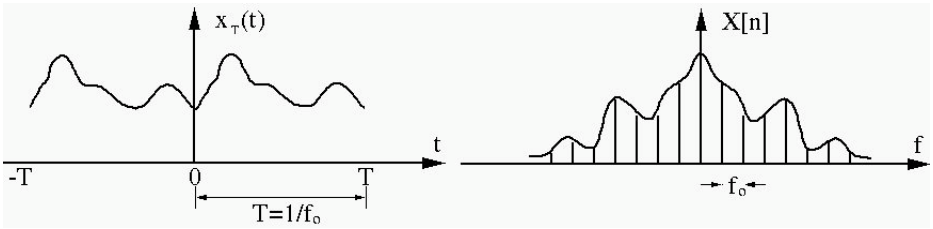

Figure 3.1 Fourier series expansion of periodic signals

Equations 3.2 and 3.4 form a pair of the Fourier series expansion:

$$x_T(t) = \frac{1}{\sqrt{T}} \sum_{k=-\infty}^{\infty} X[k] e^{j2k\pi f_0 t}$$
$$X[k] = \frac{1}{\sqrt{T}} \int_T x_T(t) e^{-j2k\pi f_0 t} dt \tag{3.5}$$

As the signal and the basis functions are both periodic, the integral above can be over any interval of T, such as from 0 to T, or from $-T/2$ to $T/2$.

In practice, the constant scaling factor $1/\sqrt{T}$ in the equations above has little significance, and the Fourier series expansion pair could be expressed in some alternative forms such as:

$$x_T(t) = \sum_{k=-\infty}^{\infty} X[k] e^{j2k\pi f_0 t} = \sum_{k=-\infty}^{\infty} X[k] e^{jk\omega_0 t}$$
$$X[k] = \frac{1}{T} \int_T x_T(t) e^{-j2k\pi f_0 t} dt = \frac{1}{T} \int_T x_T(t) e^{-jk\omega_0 t} dt \tag{3.6}$$

Now $X[0] = \int_T x_T(t)dt/T$ has a clear interpretation, the average or the DC component of the signal.

In some literatures, angular frequency $\omega_0 = 2\pi f_0 = 2\pi/T$ is preferred to use. But we will use either f_0 or $\omega_0 = 2\pi f_0 = 2\pi/T$ interchangeably.

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 $[\cdots, X[-1], X[0], X[1], \cdots]^T$ in another vector space. Moreover, the inner product of any two functions $x_T(t)$ and $y_T(t)$ remains the same before and after the

transformation:

$$\begin{aligned}
&< x_T(t), y_T(t) >= \int_T x_T(t)\overline{y}_T(t)dt \\
&= \frac{1}{T}\int_T \sum_{k=-\infty}^{\infty} X[k]e^{j2k\pi f_0 t} \sum_{l=-\infty}^{\infty} \overline{Y}[l]e^{-j2n\pi f_0 t}dt \\
&= \frac{1}{T}\sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} X[k]\overline{Y}[l]\int_T e^{j2(k-l)\pi f_0 t}dt \\
&= \sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} X[k]\overline{Y}[l]\delta[k-l] = \sum_{k=-\infty}^{\infty} X[k]\overline{Y}[k] =< \boldsymbol{X}, \boldsymbol{Y} >
\end{aligned} \tag{3.7}$$

In particular, if $y_T(t) = x_T(t)$, the above becomes Parseval's identity

$$||x_T(t)||^2 =< x_T(t), x_T(t) >=< \boldsymbol{X}, \boldsymbol{X} >= ||\boldsymbol{X}||^2 \tag{3.8}$$

indicating that the total energy or information contained in the signal is preserved 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:

$$\begin{aligned}
x_T(t) &= \sum_{k=-\infty}^{\infty} X[k]e^{jk\omega_0 t} = X[0] + \sum_{k=1}^{\infty}[X[-k]e^{-jk\omega_0 t} + X[k]e^{jk\omega_0 t}] \\
&= X[0] + \sum_{k=1}^{\infty}[X[-k](\cos k\omega_0 t - j\sin k\omega_0 t) + X[k](\cos k\omega_0 t + j\sin k\omega_0 t)] \\
&= X[0] + \sum_{k=1}^{\infty}[(X[k] + X[-k])\cos k\omega_0 t + j(X[k] - X[-k])\sin k\omega_0 t] \\
&= X[0] + 2\sum_{k=1}^{\infty}(a_k \cos k\omega_0 t + b_k \sin k\omega_0 t)
\end{aligned} \tag{3.9}$$

where

$$\begin{aligned}
a_k &= \frac{X[k] + X[-k]}{2} = \frac{1}{2T}\int_T x_T(t)[e^{-jk\omega_0 t} + e^{jk\omega_0 t}]dt = \frac{1}{T}\int_T x_T(t)\cos k\omega_0 t dt \\
b_k &= \frac{j(X[k] - X[-k])}{2} = \frac{j}{2T}\int_T x_T(t)[e^{-jk\omega_0 t} - e^{jk\omega_0 t}]dt = \frac{1}{T}\int_T x_T(t)\sin k\omega_0 t dt \\
&\qquad (k = 1, 2, \cdots)
\end{aligned} \tag{3.10}$$

This is an alternative form of the Fourier series expansion of $x_T(t)$. Here we have used the Euler's formula:

$$\cos k\omega_0 = \frac{e^{jk\omega_0} + e^{-jk\omega_0}}{2}, \qquad \sin k\omega_0 = \frac{e^{jk\omega_0} - e^{-jk\omega_0}}{2j} \tag{3.11}$$

In particular, if $x_T(t)$ is real as all physical signals in reality, we have

$$X[-k] = \frac{1}{\sqrt{T}} \int_T x_T(t) e^{j2k\pi f_0 t} dt = \overline{X}[k] \tag{3.12}$$

i.e.,

$$Re[X[-k]] = Re[X[k]], \qquad Im[X[-k]] = -Im[X[k]] \tag{3.13}$$

i.e., the real part of $X[k]$ is even and the imaginary part is odd. Now we have:

$$\begin{aligned} a_k &= \frac{X[k] + X[-k]}{2} = \frac{X[k] + \overline{X}[k]}{2} = Re[X[k]] \\ b_k &= \frac{j(X[k] - X[-k])}{2} = \frac{j(X[k] - \overline{X}[k])}{2} = -Im[X[k]] \end{aligned} \tag{3.14}$$

and

$$\begin{cases} |X[k]| = \sqrt{a_k^2 + b_k^2} \\ \angle X[k] = -\tan^{-1} b_k/a_k \end{cases} \qquad \begin{cases} a_k = |X[k]| \cos \angle X[k] \\ b_k = -|X[k]| \sin \angle X[k] \end{cases} \tag{3.15}$$

Now the Fourier series expansion of a real signal $x_T(t)$ (Eq. 3.9) can be rewritten as:

$$\begin{aligned} x_T(t) &= X[0] + 2\sum_{k=1}^{\infty} (a_k \cos k\omega_0 t + b_k \sin k\omega_0 t) \\ &= X[0] + 2\sum_{k=1}^{\infty} |X[k]| (\cos \angle X[k] \cos k\omega_0 t - \sin \angle X[k] \sin k\omega_0 t) \\ &= X[0] + 2\sum_{k=1}^{\infty} |X[k]| \cos(k\omega_0 t + \angle X[k]) \end{aligned} \tag{3.16}$$

In other words, 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,

- when $k = 0$, the coefficient $X[0] = \int_T x_T(t)dt/T$ is the average or DC component (offset) of the signal $x_T(t)$;
- when $k = 1$, the sinusoid $\cos(\omega_0 t + \phi_1) = \cos(2\pi t/T + \angle X[1])$ has the same period T as the signal $x_T(t)$ and is therefore called the *fundamental frequency* of the signal;
- when $k > 1$, 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 kth *harmonic* of the signal.

Example 3.1: In this example we show the Fourier series expansion of three periodic signals. A family of sinusoids of different frequencies are shown on the left of Fig.3.2 as the basis functions that span the function space. These basis

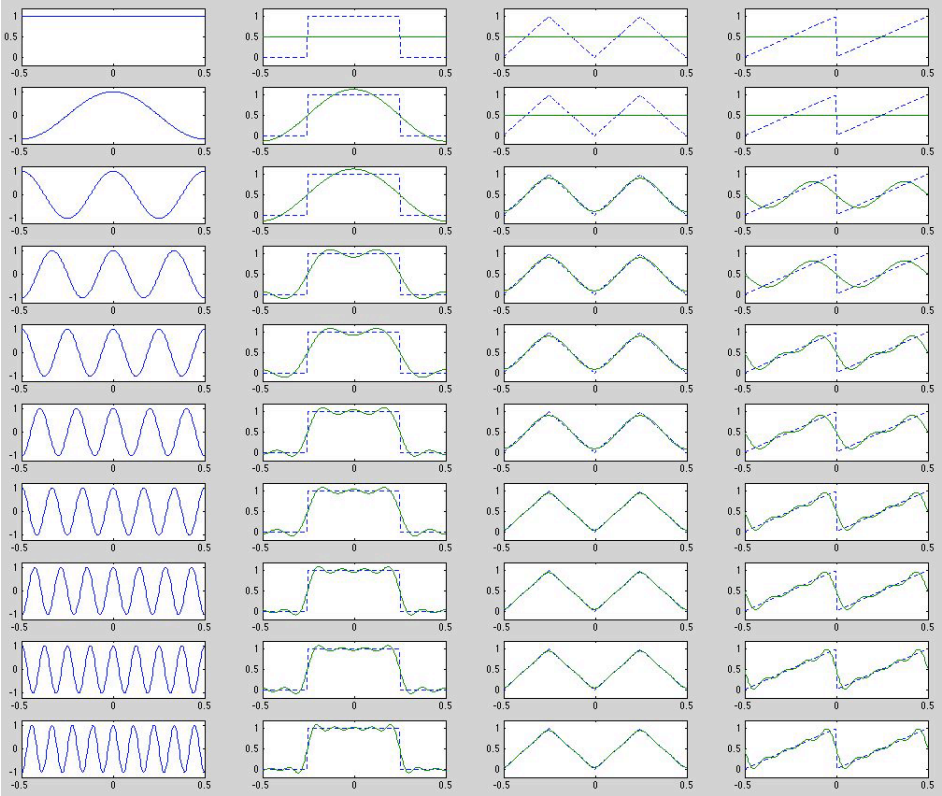

Figure 3.2 Fourier expansion of square, triangle, and sawtooth waves

functions can be linearly combined with different weights, the Fourier coefficients, to represent various functions in the space, such as the square wave, triangle wave, and sawtooth wave shown in the figure. As can be seen, the accuracy of the approximation of a signal is improved continuously as progressively more basis functions of higher frequencies, the higher harmonics, are included so that finer details (corresponding to rapid changes in time) can be better represented. The actual Fourier coefficients used in these expansions will be derived later and shown in Fig.3.3.

3.1.3 Properties of The Fourier Series Expansion

Here we discuss only a few of the properties of the Fourier series expansion. Let $x_T(t)$ be a periodic signal with period T and $X[k]$ be its Fourier series expansion coefficients.

- **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 $a/T =$

af_0. If $a > 1$, the signal is compressed by a factor a and the frequencies of its fundamental and harmonics are a times higher; if $a < 1$, the signal is expanded and the frequencies of its fundamental and harmonics are a times lower. But in either case, the coefficients remain the same:

$$x(at) = \sum_{k=-\infty}^{\infty} X[k]e^{jka\omega_0 t} \tag{3.17}$$

- **Time shifting:** A time signal $x(t)$ shifted in time by t_0 becomes $y(t) = x(t - t_0)$. Defining $t' = t - t_0$ we can get its Fourier coefficient as:

$$Y[k] = \frac{1}{T}\int_T x(t - t_0)e^{-jk\omega_0 t}dt = \frac{1}{T}\int_T x(t')e^{-jk\omega_0(t'+t_0)}dt$$
$$= X[k]e^{-jk\omega_0 t_0} = X[k]e^{-j2k\pi f t_0} \tag{3.18}$$

- **Differentiation:** The time derivative of $x(t)$ is $y(t) = d\,x(t)/dt$ its Fourier coefficients can be found to be:

$$Y[k] = \frac{1}{T}\int_T \frac{d}{dt}x(t)e^{-jk\omega_0 t}dt = \frac{1}{T}\left[e^{-jk\omega_0 t}x(t)\Big|_0^T + jk\omega_0\int_T x(t)e^{-jk\omega_0 t}dt\right]$$
$$= jk\omega_0 X[k] = jk\frac{2\pi}{T}X[k] \tag{3.19}$$

- **Integration:** The time integration of $x(t)$ is

$$y(t) = \int_{-\infty}^{t} x(\tau)d\tau \tag{3.20}$$

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 $x(t) = y'(t)$, according to the differentiation property, we have

$$X[k] = jk\frac{2\pi}{T}Y[k], \quad \text{i.e.} \quad Y[k] = \frac{T}{j2k\pi}X[k] \tag{3.21}$$

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:

$$Y[0] = \frac{1}{T}\int_T y(t)dt \tag{3.22}$$

- **Parseval's theorem:**

$$\frac{1}{T}\int_T |x_T(t)|^2 dt = \sum_{k=-\infty}^{\infty} |X[k]|^2 \tag{3.23}$$

This can be easily proven:

$$\frac{1}{T}\int_T |x_T(t)|^2 dt = \frac{1}{T}\int_T x_T(t)\overline{x}_T(t)dt$$
$$= \frac{1}{T}\int_T \sum_{k=-\infty}^{\infty} X[k]e^{j2\pi k f_0 t} \sum_{l=-\infty}^{\infty} \overline{X}[l]e^{-j2\pi l f_0 t}dt = \sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} X[k]\overline{X}[l]\frac{1}{T}\int_T e^{j2\pi k f_0 t}e^{-j2\pi l f_0 t}dt$$
$$= \sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} X[k]\overline{X}[l]\delta[k-l] = \sum_{k=-\infty}^{\infty} |X[k]|^2$$

Obviously the left-hand side of Eq.3.23 represents the average power in $x_T(t)$, and similarly,

$$\frac{1}{T}\int_T |X[k]e^{j2\pi k f_0 t}|^2 dt = \frac{1}{T}\int_T |X[k]|^2 dt = |X[k]|^2$$

represents the average power in the kth frequency component. Therefore Eq.3.23 states that the average power in one period of the signal is the sum of the average powers in 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

- **Constant:**
  A constant $x(t) = c$ can be expressed as a complex exponential $x(t) = e^{j0t}$ with arbitrary period T, i.e., it is a zero-frequency or DC (direct current) component. The coefficient for this zero frequency is $X[0] = c$, while all other coefficients for nonzero frequencies are zero. Alternatively, following the definition, we get

$$X[k] = \frac{1}{T}\int_T ce^{-jk\omega_0 t}dt = \begin{cases} c & k = 0 \\ 0 & k \neq 0 \end{cases} \tag{3.24}$$

  The last equal sign is due to Eq.1.27
- **Complex exponential:**
  A complex exponential $x(t) = e^{j\omega_0 t}$ (with period $T = 2\pi/\omega_0$) with a coefficient $X[1] = 1$. We can also find $X[k]$ by definition:

$$c_k = \frac{1}{T}\int_T e^{j\omega_0 t}e^{-jk\omega_0 t}dt = \frac{1}{T}\int_T e^{j\omega_0(1-k)t}dt = \delta[k-1] = \begin{cases} 1 & k = 0 \\ 0 & k \neq 0 \end{cases}$$

- **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

$$\begin{aligned} X[k] &= \frac{1}{T}\int_T \cos(2\pi f_0 t) e^{-j2\pi k f_0 t} dt \\ &= \frac{1}{2}[\frac{1}{T}\int_T e^{-j2\pi(k-1)f_0 t} dt + \frac{1}{T}\int_T e^{-j2\pi(k+1)f_0 t} dt] \\ &= \frac{1}{2}(\delta[k-1] + \delta[k+1]) \end{aligned} \tag{3.25}$$

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:

$$\begin{aligned} X[k] &= \frac{1}{T}\int_T \sin(2\pi f_0 t) e^{-j2\pi k f_0 t} dt \\ &= \frac{1}{2j}[\frac{1}{T}\int_T e^{-j2\pi(k-1)f_0 t} dt - \frac{1}{T}\int_T e^{-j2\pi(k+1)f_0 t} dt] \\ &= \frac{1}{2j}(\delta[k-1] - \delta[k+1]) \end{aligned} \tag{3.26}$$

- **Square wave:**
  Let $x(t)$ be an odd square wave:

$$x(t) = \begin{cases} 1 & 0 < t < \tau \\ 0 & \tau < t < T \end{cases} \tag{3.27}$$

The Fourier coefficients of this function are

$$X[k] = \frac{1}{T}\int_0^T x(t) e^{-jk\omega_0 t} dt = \frac{1}{T}\int_0^\tau e^{-jk\omega_0 t} dt = \frac{1}{j2k\pi}\,(1 - e^{-jk\omega_0 \tau}) \tag{3.28}$$

In particular, as the DC component, $X[0] = \tau/T$. A *sinc function* is commonly defined as:

$$sinc(x) = \frac{\sin(\pi x)}{\pi x}, \quad \text{and} \quad \lim_{x\to 0} sinc(x) = 1 \tag{3.29}$$

and the expression above for $X[k]$ can be written as:

$$\begin{aligned} X[k] &= \frac{e^{-jk\pi\tau/T}}{k\pi}\,\frac{1}{2j}(e^{jk\pi\tau/T} - e^{-jk\pi\tau/T}) \\ &= \frac{e^{-jk\pi\tau/T}}{k\pi}\,sin(k\pi\tau/T) = \frac{\tau}{T} sinc(k\tau/T) e^{-jk\pi\tau/T} \end{aligned} \tag{3.30}$$

In particular, if $\tau = T/2$, then $X[0] = 1/2$ and $X[k]$ above becomes:

$$X[k] = \frac{1}{j2k\pi}\,(1 - e^{-jk\pi}) \tag{3.31}$$

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:

$$X[\pm(2k-1)] = \pm 1/j\pi(2k-1), \quad (k = 1, 2, \cdots) \tag{3.32}$$

and the Fourier series expansion of the square wave becomes a linear combination of sinusoids:

$$\begin{aligned} x(t) &= \sum_{k=-\infty}^{\infty} X[k]e^{j2k\pi f_0 t} \\ &= X[0] + \sum_{k=1}^{\infty}[\frac{1}{j\pi(2k-1)}e^{j(2k-1)\omega_0 t} + \frac{1}{-j\pi(2k-1)}e^{-j(2k-1)\omega_0 t}] \\ &= \frac{1}{2} + \frac{2}{\pi}\sum_{k=1}^{\infty}\frac{\sin((2k-1)\omega_0 t)}{2k-1} \\ &= \frac{1}{2} + \frac{2}{\pi}[\frac{\sin(\omega_0 t)}{1} + \frac{\sin(3\omega_0 t)}{3} + \frac{\sin(5\omega_0 t)}{5} + \cdots] \end{aligned} \tag{3.33}$$

If we remove the DC component of $x(t)$ by letting $X[0] = 0$, the square wave become

$$x(t) = \begin{cases} 1/2 & 0 < t < T/2 \\ -1/2 & T/2 < t < T \end{cases} \tag{3.34}$$

and the square wave is an odd function composed of odd harmonics of sine functions (odd).

Homework problem:

If the square wave is shifted to the left by $T/4$, it becomes an even function:

$$x_T(t) = \begin{cases} 1 & |t| < T/4 \\ 0 & T/4 < |t| < T/2 \end{cases} \tag{3.35}$$

Show that its Fourier series expansion becomes

$$\begin{aligned} x(t) &= \sum_{k=-\infty}^{\infty} X[k]e^{jk\omega_0 t} \\ &= \frac{1}{2} + \frac{2}{\pi}[\frac{\cos(\omega_0 t)}{1} - \frac{\cos(3\omega_0 t)}{3} + \frac{\cos(5\omega_0 t)}{5} + \cdots] \end{aligned} \tag{3.36}$$

composed of odd harmonics of cosine functions (even).

- **Triangle wave:** A triangle wave can be defined as:

$$x(t) = 2|t|/T, \quad (|t| \le T/2) \tag{3.37}$$

This triangle wave can be obtained as an integral of the square wave defined in Eq. 3.27 with these modifications: (a) $\tau = T/2$, (b) DC offset $X[0]$ set to zero, and (c) scaled by $4/T$. Now according to the integration property, the Fourier coefficients can be easily obtained as

$$X[k] = \frac{4}{T}\,\frac{T}{j2k\pi}\,\frac{e^{-jk\pi/2}}{k\pi}\sin(k\pi/2) = \frac{2}{j}\frac{\sin(k\pi/2)}{(k\pi)^2}e^{-jk\pi/2} \tag{3.38}$$

The DC offset is $X[0] = 1/2$. According to the time shift property, the complex exponential $e^{-jk\pi/2}$ corresponds to a right-shifted signal $x(t - t_0)$ by $t_0 = T/4$.

If we shift the signal left by $T/4$, then the complex exponential term in the expression of the coefficients disappears.
The Fourier series expansion of such a triangle wave can be written as

$$\begin{aligned} x(t) &= \sum_{k=-\infty}^{\infty} X[k]e^{j2k\pi f_0 t} = \frac{1}{2} + \sum_{k=1}^{\infty}[X[k]e^{j2k\pi f_0 t} + X[-k]e^{-j2k\pi f_0 t}] \\ &= \frac{1}{2} + \sum_{k=1}^{\infty}(\frac{2}{j}\frac{\sin(k\pi/2)}{(k\pi)^2}e^{j2k\pi f_0 t} - \frac{2}{j}\frac{\sin(k\pi/2)}{(k\pi)^2}e^{-j2k\pi f_0 t}) \\ &= \frac{1}{2} + \frac{4}{\pi^2}\sum_{k=1}^{\infty}\frac{\sin(k\pi/2)}{k^2}\sin(2k\pi f_0 t) \\ &= \frac{1}{2} + \frac{4}{\pi^2}[\sin(2\pi f_0 t) - \frac{1}{9}\sin(6\pi f_0 t) + \frac{1}{25}\sin(10\pi f_0 t) - \cdots] \end{aligned} \quad (3.39)$$

- **Sawtooth:**
A sawtooth function is defined as

$$x(t) = t/T, \qquad (0 < t < T) \quad (3.40)$$

We first find $X[0]$, the average or DC component:

$$X[0] = \frac{1}{T}\int_T \frac{t}{T}e^{-j0\omega_0 t}dt = \frac{1}{2} \quad (3.41)$$

Next we find all remaining coefficients $X[k]$ $(k \neq 0)$:

$$X[k] = \frac{1}{T}\int_T \frac{t}{T}e^{-jk\omega_0 t}dt \quad (3.42)$$

In general, this type of integrals can be found using integration by parts:

$$\int te^{at}dt = \frac{1}{a^2}(at-1)e^{at} + C \quad (3.43)$$

Here $a = -jk\omega_0 = -j2k\pi/T \neq 0$ and we get

$$X[k] = \frac{1}{T^2(jk\omega_0)^2}[(-jk\omega_0 t - 1)e^{-jk\omega_0 t}]_0^T = \frac{j}{2k\pi} \quad (3.44)$$

The Fourier series expansion of the function is

$$x(t) = \frac{1}{2} + \sum_{k=1}^{\infty}[\frac{j}{2k\pi}e^{j\omega_0 t} - \frac{j}{2k\pi}e^{-j\omega_0 t}] = \frac{1}{2} - \frac{1}{\pi}\sum_{k=1}^{\infty}\frac{1}{k}\sin(k\omega_0 t) \quad (3.45)$$

Note that this sawtooth wave is an odd function and therefore it is composed of only odd sine functions.
Homework problem: Consider a different version of the sawtooth wave:

$$x(t) = t/T, \qquad (0 < |t| < T/2) \quad (3.46)$$

- **Impulse Train:**

An impulse train, also called a comb function or sampling function, is a sequence of infinite unit impulse separated by time interval T:

$$comb(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT) \tag{3.47}$$

As a periodic function with period T, an impulse train can be Fourier expanded:

$$comb(t) = \sum_{k=-\infty}^{\infty} Comb[k]\, e^{j2k\pi t/T} \tag{3.48}$$

with coefficients:

$$\begin{aligned} Comb[k] &= \frac{1}{T}\int_{-T/2}^{T/2} comb(t)e^{-j2k\pi t/T}dt = \frac{1}{T}\int_{-T/2}^{T/2}\sum_{n=-\infty}^{\infty}\delta(t-nT)e^{-j2k\pi t/T}dt \\ &= \frac{1}{T}\int_{-T/2}^{T/2}\delta(0)e^{-j2k\pi t/T}dt = \frac{1}{T}, \quad (k = 0, \pm 1, \pm 2, \cdots) \end{aligned} \tag{3.49}$$

The last equation is due to Eq. 1.6. Substituting $Comb[k] = 1/T$ back into the Fourier series expansion of $comb(t)$, we can also express the impulse train as:

$$comb(t) = \sum_{n=-\infty}^{\infty}\delta(t-nT) = \frac{1}{T}\sum_{k=-\infty}^{\infty} e^{j2k\pi t/T} \tag{3.50}$$

This is actually the same as Eq.1.28 shown before.

Fig.3.3 shows a set of periodic signals (on the left) and their corresponding Fourier coefficients (on the right).

3.2 The Fourier Transform of Non-Periodic Signals

3.2.1 Formulation

The Fourier series expansion is not applicable if the given signal $x(t)$ is non-periodic. In order to still be able to process and analyze the signal in frequency domain, the concept of the Fourier series expansion needs to be modified.

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:

$$\begin{aligned} x_T(t) &= \sum_{k=-\infty}^{\infty}\frac{1}{T}X[k]e^{jk\omega_0 t} = \sum_{k=-\infty}^{\infty}\frac{1}{T}X[k]e^{j2k\pi f_0 t} \\ X[k] &= \int_T x_T(t)e^{-jk\omega_0 t}dt = \int_T x_T(t)e^{-j2k\pi f_0 t}dt \end{aligned} \tag{3.51}$$

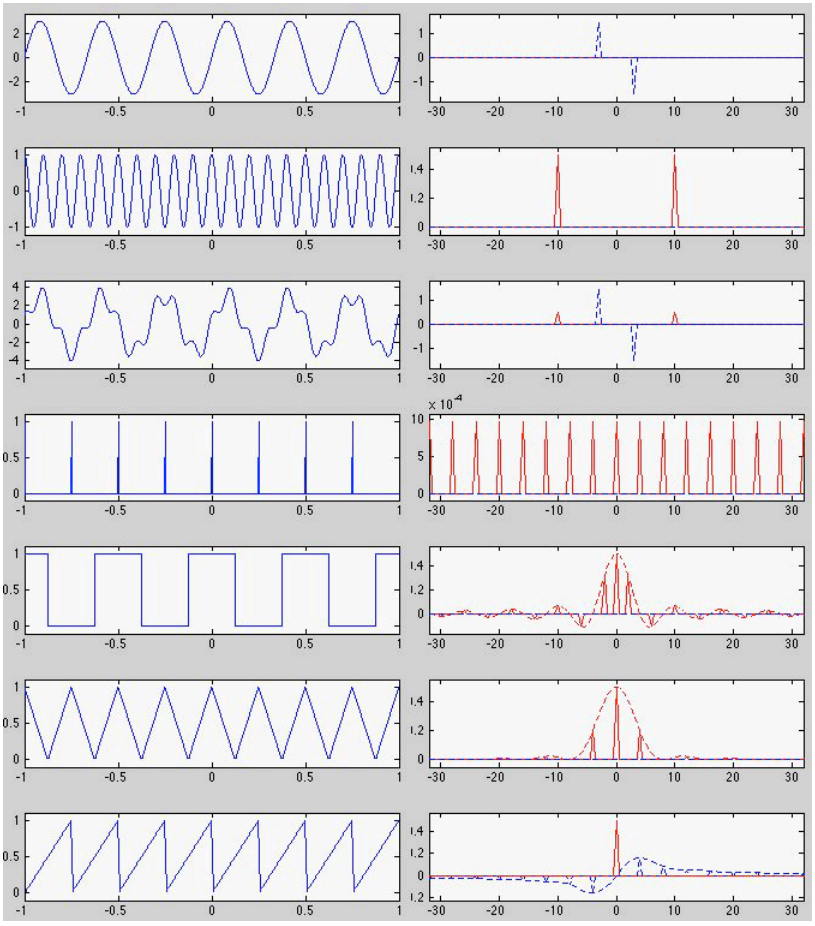

Figure 3.3 Examples of Fourier series expansions
A set of periodic signals are shown on the left and their Fourier expansion coefficients are shown on the right as a function of frequency f (real and imaginary parts are shown in solid and dashed lines, respectively). The first three rows show two sinusoids $x_1(t) = \sin(2\pi 3t)$ and $x_2(t) = \cos(2\pi 10t)$, and their sum $x_1(t) + x_2(t)$. The following four rows are for the impulse train, square wave, triangle wave, and sawtooth wave, respectively.

Here the coefficient $X[k]$ is redefined so that its value is scaled by T, and its dimensionality becomes that of the signal $x_T(t)$ multiplied by time, or divided by frequency (while 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 $T \to \infty$. At the limit the following changes take place:

- $\omega_0 = 2\pi f_0 = 2\pi/T \to 0$, and the discrete frequencies $k\omega_0 = 2k\pi f_0$ for all $k = -\infty, \cdots, -1, 0, 1, \cdots, \infty$ can be replaced by a continuous variable $-\infty < \omega = 2\pi f < \infty$.
- The discrete and periodic basis functions $\phi_k(t) = e^{j2k\pi t/T}$ for all k become uncountable and non-periodic $\phi_f(t) = e^{j2\pi ft}$ for all f, and they now span a function space over $(-\infty, \infty)$ containing all non-periodic functions $x(t)$.
- The coefficients $X[k]$ for the discrete frequency components $\phi_k(t) = e^{j2k\pi f_0 t}$ for all k is replaced by a continuous weight function $X(f)$ for the continuous and uncountable frequency component function $\phi_f(t) = e^{j2\pi ft}$ for all f.
- Define $\triangle f = 1/T$, then $\triangle f \to df$, and the summation in the first equation in Eq. 3.51 becomes an integral.

Due to the changes above, the two equations in Eq. 3.51 become

$$x(t) = \lim_{T\to\infty}[\frac{1}{T}\sum_{k=-\infty}^{\infty} X[k]e^{j2k\pi f_0 t}] = \int_{-\infty}^{\infty} X(f)^{j2\pi ft}df$$
$$X(f) = \lim_{T\to\infty}[\int_{T} x(t)e^{-j2k\pi f_0 t}dt] = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt \tag{3.52}$$

These two equations form the *continuous-time Fourier transform (CTFT)* pair:

$$X(f) = \mathcal{F}[x(t)] = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt$$
$$x(t) = \mathcal{F}^{-1}[X(f)] = \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df \tag{3.53}$$

The second equation, the inverse Fourier transform, represents a non-periodic signal $x(t)$ as a linear combination of an uncountable and infinite set of basis functions $\phi_f(t) = e^{j2\pi ft}$, weighted by a coefficient or weight function $X(f)$, called the *frequency spectrum* of $x(t)$, which can be obtained as the projection of the signal $x(t)$ onto a basis function $\phi_f(t)$ representing frequency f:

$$X(f) = \mathcal{F}[x(t)] =< x(t), \phi_f(t) >=< x(t), e^{j2\pi ft} >= \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt \tag{3.54}$$

This is the first equation, the forward Fourier transform. As the dimension of $X(f)$ is that of the signal $x(t)$ multiplied by time or divided by frequency, it is actually a *frequency density* function, representing the distribution of energy or information contained in the signal over frequency. This integral is also called an *integral transform*, and $\phi_f(t) = e^{j2\pi ft}$, a function of two variables t and f, is called the *kernel function* of the transform.

The Fourier transform pair in Eq.3.53 can also be equivalently represented in terms of the angular frequency $\omega = 2\pi f$:

$$X(\omega) = \mathcal{F}[x(t)] = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt$$
$$x(t) = \mathcal{F}^{-1}[X(\omega)] = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\omega)e^{j\omega t}d\omega \tag{3.55}$$

In some literatures, the spectrum $X(f)$ or $X(\omega)$ is also denoted by $X(j\omega)$, because it takes this form when considered as a special case of the Laplace transform, to be considered in a later chapter. However, we note that all these different forms of the spectrum are just some notational variations all representing essentially the same fact: the spectrum is simply a function of frequency f, or angular frequency $f = 2\pi f$. No confusion should be caused given the specific context in the discussion. Moreover, when the spectrum is denoted by $X(f)$, the Fourier transform pair in Eq.3.53 appears symmetric so that the time-frequency duality an be more clearly revealed. Therefore we will use $X(f)$, $X(\omega)$ and $(j\omega)$ interchangeably, whichever is more convenient and suitable in each specific case.

In order for the integral in Eq.3.53 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:

$$\int_{-\infty}^{\infty} |x(t)|dt < \infty \tag{3.56}$$

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. 2.33). As some obvious examples, signals such as $x(t) = t$ and $x(t) = t^2$ grow without bound as $|t| \to \infty$ 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 still exist. For example, some important signals such $x(t) = 1$, $x(t) = u(t)$, and $x(t) = \delta(t)$ are neither square integrable nor absolutely integrable, but their Fourier spectra can all be obtained, due to the use 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 (Theorem 2.6):

$$\begin{aligned} &< x(t), y(t) >= \int_{-\infty}^{\infty} x(t)\overline{y}(t)dt = \int_{-\infty}^{\infty} [\int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df] \, [\int_{-\infty}^{\infty} \overline{Y}(f')e^{-j2\pi f't}df'] \, dt \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} X(f)\overline{Y}(f')[\int_{-\infty}^{\infty} e^{j2\pi(f-f')t}dt] \, df \, df' = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} X(f)\overline{Y}(f')\delta(f-f')df \, df' \\ &= \int_{-\infty}^{\infty} X(f)\overline{Y}(f)df =< X(f), Y(f) > \end{aligned} \tag{3.57}$$

Here we have used the fact (Eq.1.26) that:

$$\int_{-\infty}^{\infty} e^{j2\pi(f-f')t}dt = \delta(f-f') \tag{3.58}$$

This equation has a major significance as it also illustrates that indeed the function family $\{\phi_f(t) = e^{j2\pi ft},\ (-\infty < f < \infty)\}$ forms an orthonormal basis that spans the function space, and any function in the space can be expressed as a linear combination of these basis functions. This is the very essence of the inverse Fourier transform given in Eq.3.53.

Replacing $y(y)$ by $x(t)$ in Eq.3.57 above, we get Parseval's equality:

$$||x(t)||^2 =< x(t), x(t) >=< X(f), X(f) >= ||X(f)||^2 \tag{3.59}$$

As a unitary transformation, the Fourier transform can be considered as a rotation of the basis functions of the function space. Before the Fourier 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:

$$x(t) = \int_{-\infty}^{\infty} x(\tau)\delta(t-\tau)d\tau \tag{3.60}$$

After the transformation, the function is represented as a linear combination of a different set of orthonormal basis functions $\phi_f(t) = e^{j2\pi ft}$ for all frequencies f, weighted by the coefficient function $X(f)$ for the amplitude of each frequency component:

$$x(t) = \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df \tag{3.61}$$

The representations of the signal in time domain by $x(t)$ and in frequency domain by $X(f)$ are equivalent, in the sense that the total amount of energy or information is preserved due to the Parseval's equality $||x(t)|| = ||X(f)||$, i.e.,

$$\int_{-\infty}^{\infty} |x(t)|^2\ dt = \int_{-\infty}^{\infty} |X(f)|^2\ df$$

Based on the discussion above, we conclude that the Fourier series expansion is actually a special case of the general Fourier transform, when the time signal is periodic and consequently the spectrum is discrete. When the period $T \to \infty$ approaches infinity, the gap $f_0 = 1/T \to 0$ approaches zero, i.e., the Fourier expansion becomes the Fourier transform.

Example 3.2: Consider the Fourier transform of a few special signals:

- The Dirac delta or unit impulse function $x(t) = \delta(t)$:

$$\mathcal{F}[\delta(t)] = \int_{-\infty}^{\infty} \delta(t)e^{-j2\pi ft} = e^{-j2\pi 0f} = 1 \tag{3.62}$$

- Constant function $x(t) = 1$:

$$\mathcal{F}[1] = \int_{-\infty}^{\infty} e^{-j2\pi ft} = \delta(f) \tag{3.63}$$

The second equal sign is due to Eq.1.26.

- The sign function $x(t) = sgn(t)$:

$$sgn(t) = \begin{cases} -1 & t < 0 \\ 0 & t = 0 \\ 1 & t > 0 \end{cases}$$

$$\mathcal{F}[sgn(t)] = -\int_{-\infty}^{0} e^{-j2\pi ft}dt + \int_{0}^{\infty} e^{-j2\pi ft}dt = \int_{0}^{\infty} e^{j2\pi ft}dt + \int_{0}^{\infty} e^{-j2\pi ft}dt$$

Consider the first integral as the following limit when $a > 0$ approaches zero:

$$\lim_{a\to 0}\int_{0}^{\infty} e^{-at}e^{j2\pi ft}dt = \lim_{a\to 0}\frac{-1}{a - j2\pi f}e^{-(a-j2\pi f)t}\Big|_{0}^{\infty} = \lim_{a\to 0}\frac{1}{a - j2\pi f} = \frac{1}{j2\pi f}$$

Similarly we get the same result for the second integral, therefore:

$$\mathcal{F}[sgn(t)] = \frac{1}{j\pi f} \tag{3.64}$$

- The unit step function $x(t) = u(t)$:

$$u(t) = \frac{1}{2}[1 + sgn(t)] = \begin{cases} 0 & t < 0 \\ 1/2 & t = 0 \\ 1 & t > 0 \end{cases}$$

Due to the linearity of the Fourier transform, we have:

$$U(f) = \mathcal{F}[u(t)] = \frac{1}{2}[\mathcal{F}[1] + \mathcal{F}[sgn(t)]] = \frac{1}{2}\delta(f) + \frac{1}{j2\pi f} \tag{3.65}$$

Alternatively, $U(f) = \mathcal{F}[u(t)]$ can also be obtained directly from the definition. However, note that as the unit step $u(t)$ is not square-integrable, its Fourier transform integral does not converge in the normal sense:

$$\mathcal{F}[u(t)] = \int_{-\infty}^{\infty} u(t)e^{-j2\pi ft}dt = \int_{0}^{\infty} e^{-j2\pi ft}dt$$

To overcome this difficulty, we consider $u(t)$ as a special case of the exponential decay function $e^{-at}u(t)$ when $a = 0$, so that $U(f) = \mathcal{F}[u(t)]$ can be obtained the same as in Eq.3.65:

$$\mathcal{F}[u(t)] = \lim_{a\to 0}\mathcal{F}[e^{-at}u(t)] = \lim_{a\to 0}\frac{a - j\omega}{a^2 + \omega^2} = \frac{1}{2}\delta(f) + \frac{1}{j2\pi f}$$

The proof of this result is left to the reader as a homework problem.

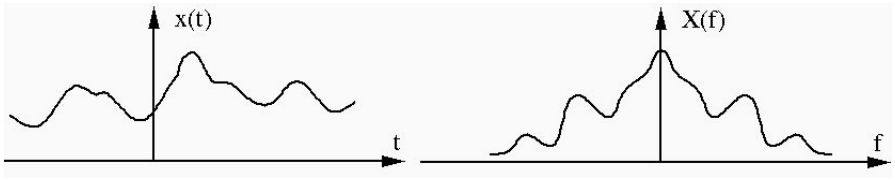

Figure 3.4 Fourier transform of non-periodic and continuous signals
When the time signal is no longer periodic, its discrete spectrum represented by the Fourier series coefficients becomes a continuous function.

3.2.2 Physical Interpretation

In general the spectrum $X(f)$ of a time signal $x(t)$ is complex and can be expressed in either Cartesian or polar form:

$$X(f) = X_r(f) + jX_j(f) = |X(f)|e^{j\angle X(f)} \tag{3.66}$$

where

$$\begin{cases} |X(f)| = \sqrt{X_r^2(f) + X_j^2(f)} \\ \angle X(f) = tan^{-1} X_j(f)/X_r(f) \end{cases}, \qquad \begin{cases} X_r(f) = |X(f)| \cos \angle X(f) \\ X_j(f) = |X(f)| \sin \angle X(f) \end{cases} \tag{3.67}$$

If the signal $x(t)$ is real, we have

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt = \int_{-\infty}^{\infty} x(t)(\cos 2\pi ft - j \sin 2\pi ft)dt = X_r(f) - jX_j(f) \tag{3.68}$$

where the real part $X_r(f)$ is even and the imaginary part $X_j(f)$ is odd:

$$\begin{aligned} X_r(f) &= \int_{-\infty}^{\infty} x(t) \cos(2\pi ft)dt = X_r(-f) \\ X_j(f) &= \int_{-\infty}^{\infty} x(t) \sin(2\pi ft)dt = -X_j(-f) \end{aligned} \tag{3.69}$$

therefore $|X(f)| = \sqrt{X_r^2(f) + X_j^2(f)}$ is even. Now the signal can be expressed as

$$\begin{aligned} x(t) &= \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df = \int_{-\infty}^{\infty} |X(f)|e^{j2\pi ft + \angle X(f)}df \\ &= \int_{-\infty}^{\infty} |X(f)| \cos(2\pi ft + \angle X(f))df + j\int_{-\infty}^{\infty} |X(f)| \sin(2\pi ft + \angle X(f))df \\ &= 2\int_{0}^{\infty} |X(f)| \cos(2\pi ft + \angle X(f))df \end{aligned} \tag{3.70}$$

The last equation is due to the fact that the integrand of the real term is even and that of the imaginary term is odd. 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 $\angle X(f)$.

3.2.3 Relation to The Fourier Expansion

We consider how the Fourier transform of a periodic function is related to its Fourier coefficients. The Fourier series expansion of a periodic function $x_T(t)$ is:

$$x_T(t) = \sum_{k=-\infty}^{\infty} X[k]e^{j2k\pi t/T} = \sum_{k=-\infty}^{\infty} X[k]e^{j2k\pi f_0 t} \tag{3.71}$$

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:

$$\begin{aligned} X(f) &= \int_{-\infty}^{\infty} x_T(t)e^{-j2\pi ft}dt = \int_{-\infty}^{\infty} [\sum_{k=-\infty}^{\infty} X[k]e^{j2k\pi f_0 t}]e^{-j2\pi ft}dt \\ &= \sum_{k=-\infty}^{\infty} X[k]\int_{-\infty}^{\infty} e^{-j2\pi(f-kf_0)t}dt = \sum_{k=-\infty}^{\infty} X[k]\delta(f-kf_0) \end{aligned} \tag{3.72}$$

Here we have used the result of Eq.1.31. It is clear that the spectrum of a periodic function is discrete, in the sense that it is none-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 dimensionality of the Fourier coefficient $X[k]$ is the same as that of the function $x_T(t)$, i.e., $[X[k]] = [x_T(t)]$, the dimensionality of the spectrum is

$$[X(f)] = [X[k]][t] = \frac{[X[k]]}{[f]} \tag{3.73}$$

i.e., $X(f)$ is a density function over frequency, only when integrated over frequency, will it become the coefficient:

$$\int_{-\infty}^{\infty} X[k]\delta(f - kf_0)df = X[k] \tag{3.74}$$

When $T \to \infty$, $x(t)$ becomes non-periodic and the gap $f_0 = 1/T$ between two consecutive frequency components in its spectrum becomes zero, i.e., the discrete spectrum becomes continuous.

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:

$$x'(t) = \sum_{n=-\infty}^{\infty} x(t + nT) = x'(t + T) \tag{3.75}$$

As $x'(t)$ is periodic, it can be Fourier expanded and the kth Fourier coefficient is:

$$\begin{aligned} X'[k] &= \frac{1}{T}\int_0^T x'(t)e^{-j2\pi kt/T} = \frac{1}{T}\int_0^T [\sum_{n=-\infty}^{\infty} x(t + nT)\,]\,e^{-j2\pi kt/T}dt \\ &= \frac{1}{T}\sum_{n=-\infty}^{\infty}\int_0^T x(t + nT)]e^{-j2\pi kt/T}dt \end{aligned} \tag{3.76}$$

If we define $\tau = t + nT$, i.e., $t = \tau - nT$, the above becomes:

$$X'[k] = \frac{1}{T}\sum_{n=-\infty}^{\infty}\int_{nT}^{(n+1)T} x(\tau)e^{-j2\pi k\tau/T}d\tau\ e^{-j2\pi nk}$$
$$= \frac{1}{T}\int_{-\infty}^{\infty} x(\tau)e^{-j2\pi k\tau/T}d\tau = \frac{1}{T}X(\frac{k}{T}) \tag{3.77}$$

(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:

$$x'(t) = \sum_{k=-\infty}^{\infty} X'[k]e^{j2\pi kt/T} = \sum_{k=-\infty}^{\infty}\frac{1}{T}X\left(\frac{k}{T}\right)e^{j2\pi kt/T} \tag{3.78}$$

This equation is called *Poisson summation formula*.

3.2.4 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 of the Fourier transform, they naturally share 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:**

$$\mathcal{F}[ax(t) + by(t)] = a\mathcal{F}[x(t)] + b\mathcal{F}[y(t)] \tag{3.79}$$

  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.3.54). Therefore due to the linearity of the inner product in the first variable, the Fourier transform is also linear.

- **Time-frequency duality:**

$$\text{if } \mathcal{F}[x(t)] = X(f), \quad \text{then} \quad \mathcal{F}[X(t)] = x(-f) \tag{3.80}$$

  Proof:

$$x(t) = \mathcal{F}^{-1}[X(f)] = \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df \tag{3.81}$$

  Defining $t' = -t$, we have

$$x(-t') = \mathcal{F}^{-1}[X(f)]\int_{-\infty}^{\infty} X(f)e^{-j2\pi ft'}df \tag{3.82}$$

  Interchanging variables t' and f, we get

$$x(-f) = \int_{-\infty}^{\infty} X(t')e^{-j2\pi ft'}dt' = \mathcal{F}[X(t)] \tag{3.83}$$

In particular, if $x(t) = x(-t)$ is even, we have

$$\text{if } \mathcal{F}[x(t)] = X(f), \quad \text{then} \quad \mathcal{F}[X(t)] = x(f) \tag{3.84}$$

This duality is simply the result of the definition of the forward and inverse transforms in Eq. 3.53, which are highly symmetric between time and frequency. Consequently, many of the properties and transforms of typical functions have strong duality between the time and frequency domains.

- **Multiplication (Plancherel) theorem:**

$$\int_{-\infty}^{\infty} x(t)\overline{y}(t)dt = \int_{-\infty}^{\infty} X(f)\overline{Y}(f)df \tag{3.85}$$

This is Eq. 3.57, 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:

$$\int_{-\infty}^{\infty} |x(t)|^2 dt = \int_{-\infty}^{\infty} x(t)\overline{x}(t)dt = \int_{-\infty}^{\infty} X(f)\overline{X}(f)df = \int_{-\infty}^{\infty} |X(f)|^2 df \tag{3.86}$$

where $|x(t)|^2$ represents how the signal energy is distributed over time, while $|X(f)|^2$ represents how the signal energy is distributed over frequency, and $|X(f)|^2 = S_x(f)$ is defined as the *power density spectrum (PDS)*.

- **Complex conjugate:**

$$\mathcal{F}[\overline{x}(t)] = \overline{X}(-f) \tag{3.87}$$

Proof: Taking the complex conjugate of the inverse Fourier transform, we get:

$$\begin{aligned}\overline{x}(t) &= \overline{\int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df} = \int_{-\infty}^{\infty} \overline{X}(f)e^{-j2\pi ft}df \\ &= \int_{-\infty}^{\infty} \overline{X}(-f')e^{j2\pi f't}df' = \mathcal{F}^{-1}[\overline{X}(-f)] \end{aligned} \tag{3.88}$$

$$\tag{3.89}$$

where we have defined $f' = -f$.

- **Symmetry:**

Let us consider some symmetry properties of the Fourier transform in both time and frequency domains. First note that:

$$\begin{aligned} X(f) &= \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt = \int_{-\infty}^{\infty} x(t)\cos(2\pi ft)dt - j\int_{-\infty}^{\infty} x(t)\sin(2\pi ft)dt \\ &= X_e(f) + jX_o(f) \end{aligned} \tag{3.90}$$

where $X_e(f)$ and $X_o(f)$ are the even and odd components of $X(f)$:

$$X_e(f) = \int_{-\infty}^{\infty} x(t)\cos(2\pi ft)dt = X_e(-f) \tag{3.91}$$

$$X_o(f) = -\int_{-\infty}^{\infty} x(t)\sin(2\pi ft)dt = -X_o(-f) \tag{3.92}$$

We can also represent the spectrum in terms of its real and imaginary parts:

$$X(f) = Re[X(f)] + jIm[X(f)] = X_r(f) + jX_j(f) \tag{3.93}$$

Note, however, here $X_r(f)$ and $X_j(f)$ do not necessarily correspond to $X_e(f)$ and $X_o(f)$, respectively, as $x(t)$ is in general assumed to be complex.

Real signal:

If $x(t)$ is real, then both $X_e(f)$ and $X_o(f)$ are real, and they become respectively the real and imaginary parts of $X(f)$:

$$\begin{cases} X_e(f) = X_r(f) = X_r(-f) \\ X_o(f) = X_j(f) = -X_j(-f) \end{cases} \tag{3.94}$$

As the real part of $X(f)$ is even, and the imaginary part is odd, i.e., the spectrum $X(f)$ of a real signal is a *Hermitian function* satisfying:

$$X(-f) = X_r(-f) + jX_j(-f) = X_r(f) - jX_j(f) = \overline{X}(f) \tag{3.95}$$

The symmetry property of spectrum $X(f)$ indicates that in frequency domain, only half of the data is independent (fifty percent redundancy), which is of course the natural consequence of the fact that only half of the data in time domain, the real part, is independent, as the imaginary part is all zero.

As the spectrum of a real signal symmetric (real part even and imaginary odd), it can be reconstructed by inverse transform by only half of the spectrum for $f > 0$:

$$\begin{aligned} x(t) &= \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df = \int_{-\infty}^{0} X(f)e^{j2\pi ft}df + \int_{0}^{\infty} X(f)e^{j2\pi ft}df \\ &= \int_{0}^{\infty} X(-f)e^{-j2\pi ft}df + \int_{0}^{\infty} X(f)e^{j2\pi ft}df \\ &= \int_{0}^{\infty} [\overline{X}(f)e^{-j2\pi ft} + X(f)e^{j2\pi ft}]df \end{aligned} \tag{3.96}$$

Moreover, depending on whether $x(t)$ is even or odd, we have the following results:

- If $x(t) = x(-t)$ is even $(X_j(f) = X_o(f) = 0)$, $X(f) = X_r(f) = X_e(f)$ is real and even;
- If $x(t) = x(-t)$ is odd $(X_j(f) = X_e(f) = 0)$, $X(f) = X_j(f) = X_o(f)$ is imaginary and odd.

Imaginary signal:

If $x(t)$ is imaginary, then both $X_e(f)$ and $X_o(f)$ are imaginary, and they become respectively the imaginary and real parts of $X(f)$:

$$\begin{cases} X_e(f) = X_j(f) = X_j(-f) \\ X_o(f) = X_r(f) = -X_r(-f) \end{cases} \tag{3.97}$$

and we have

$$X(-f) = X_r(-f) + jX_j(-f) = -X_r(f) + jX_j(f) = -\overline{X}(f) \tag{3.98}$$

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
$x_r(t) = -x_r(-t)$ real, odd	$X_j(f) = -X_j(f)$ imaginary, odd
$x(t) = x_j(t)$ 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
$x_j(t) = -x_j(-t)$ imaginary, odd	$X_r(f) = -X_r(-f)$ real, odd

i.e., the spectrum of an imaginary signal is anti-Hermitian.
Moreover, depending on whether $x(t)$ is even or odd, we have the following results:

- If $x(t) = x(-t)$ is even $(X_r(f) = X_o(f) = 0)$, $X(f) = X_j(f) = X_e(f)$ is imaginary and even;
- If $x(t) = -x(-t)$ is odd $(Im[X(f)] = X_e(f) = 0)$, $X(f) = X_r(f) = X_o(f)$ is real and odd.

- **Time reversal:**

$$\mathcal{F}[x(-t)] = X(-f) \tag{3.99}$$

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:

$$\mathcal{F}[x(-t)] = \int_{-\infty}^{\infty} x(-t)e^{-j2\pi ft}dt = \int_{-\infty}^{\infty} x(t')e^{j2\pi ft'}dt' = X(-f) \tag{3.100}$$

where we have assumed $-t' = t$. In particular, when $x(t) = \overline{x}(t)$ is real,

$$\mathcal{F}[x(-t)] = X(-f) = \int_{-\infty}^{\infty} x(t)e^{j2\pi ft} = \overline{\int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}} = \overline{X(f)} \tag{3.101}$$

- **Time and frequency scaling:**

$$\mathcal{F}[x(at)] = \frac{1}{|a|}X(\frac{f}{a}) \tag{3.102}$$

Proof: First we assume a positive scaling factor $a > 0$ and get

$$\mathcal{F}[x(at)] = \int_{-\infty}^{\infty} x(at)e^{-j2\pi ft}dt = \int_{-\infty}^{\infty} x(u)e^{-j2\pi fu/a}d(\frac{u}{a}) = \frac{1}{a}X(\frac{f}{a})$$

where we have assumed $u = at$. Next we apply the previous property to this result to get:

$$\mathcal{F}[x(-at)] = \frac{1}{a}X(-\frac{f}{a})$$

Letting $a' = -a < 0$, we get the following for a negative scaling factor:

$$\mathcal{F}[x(a't)] = \frac{1}{-a'}X(\frac{f}{a'})$$

Combining the above results for both positive and negative scaling factors, we get Eq.3.102.

If $|a| < 1$, the signal is stretched and its spectrum is compressed and scaled up. When $|a| \to 0$, $x(at)$ 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 $|a| \to \infty$, 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 shifting:**

$$\mathcal{F}[x(t \pm t_0)] = e^{\pm j2\pi f t_0} X(f) \tag{3.103}$$

$$\mathcal{F}^{-1}[X(f \pm f_0)] = e^{\mp j2\pi f_0 t} x(t) \tag{3.104}$$

Proof: We first prove Eq.3.103:

$$\mathcal{F}[x(t \pm t_0)] = \int_{-\infty}^{\infty} x(t \pm t_0) e^{-j2\pi f t} dt \tag{3.105}$$

Let $t' = t \pm t_0$, then $t = t' \mp t_0$, $dt' = dt$, the above becomes

$$\mathcal{F}[x(t \pm t_0)] = \int_{-\infty}^{\infty} x(t') e^{-j2\pi f(t' \mp t_0)} dt' = e^{\pm j2\pi f t_0} X(f) \tag{3.106}$$

A time shift t_0 of the signal corresponds to a phase shift $2\pi f t_0$ for every frequency component $e^{j2\pi f t}$. Note that as the phase shift is proportional to the frequency, a higher frequency component will shift more so that the relative positions of the harmonics remain the same, and the shape of the signal as a superposition of these harmonics remains the same when shifted.

The frequency shift property in Eq.3.104 can be then obtained by applying the time-frequency duality to the time shift property in Eq.3.103.

- **Correlation:**

The *cross-correlation* between two functions $x(t)$ and $y(t)$ is defined as

$$r_{xy}(t) = x(t) \star y(t) = \int_{-\infty}^{\infty} x(\tau)\overline{y}(\tau - t) d\tau \tag{3.107}$$

This property states:

$$\mathcal{F}[r_{xy}(t)] = \mathcal{F}[x(t) \star y(t)] = X(f)\overline{Y}(f) \tag{3.108}$$

Proof:

As $\mathcal{F}[x(\tau)] = X(f)$ and $\mathcal{F}[y(\tau - t)] = Y(f)e^{-j2\pi f t}$, we apply the multiplication theorem to get:

$$\begin{aligned} r_{xy}(t) &= \int_{-\infty}^{\infty} x(\tau)\overline{y}(\tau - t) d\tau = \int_{-\infty}^{\infty} X(f)\overline{Y}(f) e^{j2\pi f t} df \\ &= \int_{-\infty}^{\infty} S_{xy}(f) e^{j2\pi f \tau} df = \mathcal{F}^{-1}[S_{xy}(f)] \end{aligned} \tag{3.109}$$

where

$$S_{xy}(f) = X(f)\overline{Y}(f) = \mathcal{F}[r_{xy}(t)] \tag{3.110}$$

is defined as the *cross power density spectrum* $S_{xy}(f)$ of the two signals. If both signals $\overline{x}(t) = x(t)$ and $\overline{y}(t) = y(t)$ are real, i.e., $\overline{X}(f) = X(-f)$ and $\overline{Y}(f) = Y(-f)$, then we have $S_{xy}(f) = X(f)Y(-f)$. In particular, when $x(t) = y(t)$, we have:

$$r_x(t) = \int_{-\infty}^{\infty} x(\tau)\overline{x}(\tau - t)d\tau = \int_{-\infty}^{\infty} S_x e^{j2\pi f\tau} df = \mathcal{F}^{-1}[S_x(f)] \tag{3.111}$$

where $r_x(t)$ is the *auto-correlation* and $S_x(f) = X(f)\overline{X}(f) = |X(f)|^2$ is the *power density spectrum* of $x(t)$.

- **Convolution theorem:**
  As discussed in the previous chapter, the convolution of two functions $x(t)$ and $y(t)$ is defined as:

$$x(t) * y(t) = \int_{-\infty}^{\infty} x(\tau)y(t - \tau)d\tau \tag{3.112}$$

Note that 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:

$$\mathcal{F}[x(t) * y(t)] = X(f)\, Y(f) \tag{3.113}$$

$$\mathcal{F}[x(t)y(t)] = X(f) * Y(f) \tag{3.114}$$

Proof:

$$\begin{aligned}\mathcal{F}[x(t) * y(t)] &= \int_{-\infty}^{\infty}[\int_{-\infty}^{\infty} x(\tau)y(t - \tau)d\tau]e^{-j2\pi ft}dt \\ &= \int_{-\infty}^{\infty} x(\tau)e^{-j2\pi f\tau}\int_{-\infty}^{\infty} y(t - \tau)e^{-j2\pi f(t-\tau)}dt\, d\tau \\ &= \int_{-\infty}^{\infty} x(\tau)e^{-j2\pi f\tau}Y(f)d\tau = X(f)Y(f)\end{aligned} \tag{3.115}$$

Similarly, we can also prove:

$$\mathcal{F}[x(t)y(t)] = X(f) * Y(f) \tag{3.116}$$

In particular, as shown in Eq.1.66 in Chapter 1, the output $y(t)$ of an LTI system can be found as the convolution of its impulse response $h(t0$ and the input $x(t)$ $y(t) = h(t) * x(t)$. Now according to the convolution theorem, the output of the system can be more conveniently obtained in frequency domain by a multiplication:

$$Y(f) = H(f)X(f) \tag{3.117}$$

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, to be discussed in details later.

- **Time derivative:**

$$\mathcal{F}\,[\frac{d}{dt}x(t)] = j2\pi f X(f) = j\omega X(\omega) \tag{3.118}$$

Proof:

$$\begin{aligned}\frac{d}{dt}x(t) &= \frac{d}{dt}\int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df = \int_{-\infty}^{\infty} X(f)\frac{d}{dt}e^{j2\pi ft}df \\ &= \int_{-\infty}^{\infty} j2\pi f X(f)e^{j2\pi ft}df = \mathcal{F}^{-1}[j2\pi f X(f)]\end{aligned} \tag{3.119}$$

i.e., $\mathcal{F}[x'(t)] = j2\pi f X(f)$. Repeating this process we get

$$\mathcal{F}\,[\frac{d^n}{dt^n}x(t)] = (j2\pi f)^n X(f) \tag{3.120}$$

- **Frequency derivative:**

$$\begin{aligned}\mathcal{F}\,[t\,x(t)] &= j\frac{d}{df}X(f) \\ \mathcal{F}[t^n x(t)] &= j^n\frac{1}{(2\pi)^n}\frac{d^n}{df^n}X(f)\end{aligned} \tag{3.121}$$

The proof is very similar to the above.

- **Time integration:**
The Fourier transform of a time integration is:

$$\mathcal{F}[\int_{-\infty}^{t} x(\tau)d\tau] = \frac{1}{j2\pi f}\,X(f) + \frac{1}{2}X(0)\delta(f) \tag{3.122}$$

Proof:
The integral of a signal $x(t)$ can be considered as its convolution with $u(t)$:

$$x(t) * u(t) = \int_{-\infty}^{\infty} x(\tau)u(t-\tau)d\tau = \int_{-\infty}^{t} x(\tau)d\tau \tag{3.123}$$

Due to the convolution theorem, we have

$$\mathcal{F}[\int_{-\infty}^{t} x(\tau)d\tau] = \mathcal{F}[x(t) * u(t)] = X(f)[\frac{1}{j2\pi f} + \frac{1}{2}\delta(f)] = \frac{1}{j2\pi f}\,X(f) + \frac{X(0)}{2}\delta(f) \tag{3.124}$$

Comparing Eqs.3.118 and 3.122, 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.3.122 is necessary for representing the DC component in signal $x(t)$, while Eq.3.118 does not have a corresponding term as derivative operation is insensitive to DC component in the signal.

3.2.5 Fourier Spectra of Typical Functions

- **Unit impulse:**

The Fourier transform of the unit impulse function is given in Eq.3.62 according to the definition of the Fourier transform:

$$\mathcal{F}[\delta(t)] = \int_{-\infty}^{\infty} \delta(t) e^{-j2\pi ft} dt = 1 \tag{3.125}$$

- **Sign function:**
  The Fourier transform of the sign function $sgn(t)$ is given in Eq.3.64:

$$\mathcal{F}[sgn(t)] = \frac{1}{j\pi f} \tag{3.126}$$

  Moreover, based on the time-frequency duality property, we also get:

$$\mathcal{F}\left[\frac{1}{t}\right] = -j\pi sgn(f) \tag{3.127}$$

- **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:

$$\begin{aligned}\mathcal{F}[e^{-at}u(t)] &= \int_0^{\infty} e^{-at} e^{-j2\pi ft} dt = \frac{1}{-(a + j2\pi f)} e^{-(a+j2\pi f)t}\Big|_0^{\infty} \\ &= \frac{1}{a + j2\pi f} = \frac{a - j2\pi f}{a^2 + (2\pi f)^2} = \frac{a - j\omega}{a^2 + \omega^2}\end{aligned} \tag{3.128}$$

  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:

$$\mathcal{F}[e^{at}u(-t)] = \frac{1}{a - j2\pi f} = \frac{a + j2\pi f}{a^2 + (2\pi f)^2} = \frac{a + j\omega}{a^2 + \omega^2} \tag{3.129}$$

  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:

$$\begin{aligned}\mathcal{F}[e^{-a|t|}] &= \mathcal{F}[e^{-at}u(t)] + \mathcal{F}[e^{at}u(-t)] = \frac{1}{a + j2\pi f} + \frac{1}{a - j2\pi f} \\ &= \frac{2a}{a^2 + (2\pi f)^2} = \frac{2a}{a^2 + \omega^2}\end{aligned} \tag{3.130}$$

- **Unit step functions:**
  The Fourier transform of a unit step is also given before in Eq.3.65:

$$U(f) = \mathcal{F}[u(t)] = \frac{1}{j2\pi f} + \frac{1}{2}\delta(f) \tag{3.131}$$

  As the unit step is the time integral of the unit impulse:

$$u(t) = \int_{-\infty}^{t} \delta(t) dt \tag{3.132}$$

  their Fourier spectra are related according the time integration property. 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:

$$\mathcal{F}[u(-t)] = \frac{1}{2}\delta(-f) + \frac{1}{-j2\pi f} = \frac{1}{2}\delta(f) - \frac{1}{j2\pi f} \tag{3.133}$$

(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:

$$\mathcal{F}[1] = \int_{-\infty}^{\infty} e^{-j2\pi ft} dt$$

  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:

$$\mathcal{F}[1] = \mathcal{F}[u(t)] + \mathcal{F}[u(-t)] = \frac{1}{j2\pi f} + \frac{1}{2}\delta(f) - \frac{1}{j2\pi f} + \frac{1}{2}\delta(f) = \delta(f) \tag{3.134}$$

  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:

$$\mathcal{F}[1] = \int_{-\infty}^{\infty} e^{-j2\pi ft} dt = \delta(f) \tag{3.135}$$

  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:

$$\mathcal{F}[e^{j2\pi f_0 t}] = \int_{-\infty}^{\infty} e^{-j2\pi(f-f_0)t} dt = \delta(f - f_0) \tag{3.136}$$

  and according to Euler's formula, the Fourier transform of cosine function $x(t) = \cos(2\pi f_0 t)$ is:

$$\mathcal{F}[\cos(2\pi f_0 t)] = \mathcal{F}\left[\frac{1}{2}(e^{j2\pi f_0 t} + e^{-j2\pi f_0 t})\right] = \frac{1}{2}[\delta(f - f_0) + \delta(f + f_0)] \tag{3.137}$$

  and similarly the Fourier transform of $x(t) = sin(2\pi f_0 t)$ is:

$$\mathcal{F}[\sin(2\pi f_0 t)] = \frac{1}{2j}[\delta(f - f_0) - \delta(f + f_0)] \tag{3.138}$$

  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.

- **Rectangular function and sinc function:**
  A rectangular function is defined as

$$rect_\tau(t) = \begin{cases} 1 \; 0 < |t| < \tau/2 \\ 0 \text{ otherwise} \end{cases} \tag{3.139}$$

  which can be considered as the difference between two unit step functions:

$$rect(t) = u(t+\tau/2) - u(t-\tau/2) \tag{3.140}$$

  Due to the properties of linearity and time shift, the spectrum of $rect_\tau(t)$ can be found to be

$$\mathcal{F}[rect(t)] = \mathcal{F}[u(t+\tau/2)] - \mathcal{F}[u(t-\tau/2)] = \frac{e^{j\pi f\tau}}{j2\pi f} - \frac{e^{-j\pi f\tau}}{j2\pi f}$$
$$= \frac{\tau}{\pi f\tau}\sin(\pi f\tau) = \tau \; sinc(f\tau) \tag{3.141}$$

  This spectrum is zero at $f = k/\tau$ for any integer k. If we let $\tau \to \infty$, the time function is a constant 1 and its spectrum an impulse function. If we divide both sides of the equation above by τ and let $\tau \to 0$, the time function becomes an impulse and its spectrum a constant.
  On the other hand, in frequency domain, an ideal low-pass filter is defined as:

$$H_{lp}(f) = \begin{cases} 1 \; |f| < f_c \\ 0 \; |f| > f_c \end{cases} \tag{3.142}$$

  then according to time-frequency duality, its time impulse response is

$$h_{lp}(t) = \frac{\sin(2\pi f_c t)}{\pi t} = 2f_c \; sinc(2f_c t) \tag{3.143}$$

  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:**

$$triangle(t) = \begin{cases} 1 - |t|/\tau & |t| < \tau \\ 0 & |t| \geq \tau \end{cases} \tag{3.144}$$

  This triangle function (with width 2τ) is the convolution of two square functions (with width τ) scaled by $1/\tau$:

$$triangle(t) = \frac{1}{\tau} \; rect(t) * rect(t) \tag{3.145}$$

its Fourier transform can be conveniently obtained based on the convolution theorem:

$$\mathcal{F}[triangle(t)] = \frac{1}{\tau}\mathcal{F}[rect(t) * rect(t)] = \frac{1}{\tau}\ \tau\ sinc(f)\ \tau\ sinc(f) = \tau sinc^2(f\tau) \tag{3.146}$$

Alternatively, the spectrum of the triangle function can be obtained by the definition. As this is an even function, its Fourier transform is

$$\begin{aligned}
&\mathcal{F}[triangle(t)] = 2\int_0^\tau (1 - t/\tau)\cos(2\pi ft)dt \\
&= 2[\int_0^\tau \cos(2\pi ft)dt - \frac{1}{\tau}\int_0^\tau t\ \cos(2\pi ft)dt] = \frac{1}{\pi f}[\sin(2\pi f\tau) - \frac{1}{\tau}\int_0^\tau t\ d\ sin(2\pi ft)] \\
&= \frac{1}{\pi f}[\sin(2\pi f\tau) - \frac{t}{\tau}\ \sin(2\pi ft)|_0^\tau + \frac{1}{\tau}\int_0^\tau \sin(2\pi ft)dt] = \frac{-1}{2\tau(\pi f)^2}\cos(2\pi ft)|_0^\tau \\
&= \frac{1}{2\tau(\pi f)^2}(1 - \cos(2\pi f\tau)) = \tau\frac{\sin^2(\pi f\tau)}{(\pi f\tau)^2} = \tau\ sinc^2(f\tau)
\end{aligned} \tag{3.147}$$

This spectrum is zero at $f = k/\tau$ for any integer k.

- **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:

$$\begin{aligned}
&X(f) = \mathcal{F}[\frac{1}{a}e^{-\pi(t/a)^2}] = \frac{1}{a}\int_{-\infty}^{\infty} e^{-\pi(t/a)^2}\ e^{-j2\pi ft}dt = \frac{1}{a}\int_{-\infty}^{\infty} e^{-\pi((t/a)^2 + j2ft)}dt \\
&= \frac{1}{a}e^{\pi(jaf)^2}\int_{-\infty}^{\infty} e^{-\pi[(t/a)^2 + j2ft + (jaf)^2]}dt = e^{-\pi(af)^2}\int_{-\infty}^{\infty} e^{-\pi(t/a + jaf)^2}d(t/a + jaf) \\
&= e^{-\pi(af)^2}
\end{aligned} \tag{3.148}$$

  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 $a \to 0$, $x(t)$ will approach $\delta(t)$, while its spectrum $e^{-\pi(af)^2}$ approaches 1. On the other hand, if we rewrite the above as

$$X(f) = \mathcal{F}[x(t)] = \mathcal{F}[e^{-\pi(t/a)^2}] = ae^{-\pi(af)^2} \tag{3.149}$$

  and let $a \to \infty$, $x(t)$ approaches 1 and $X(f)$ approaches $\delta(f)$.

- **Impulse train:**
  As discussed before the impulse train is a sequence of infinite unit impulses separated by a constant time interval T:

$$comb(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT) \tag{3.150}$$

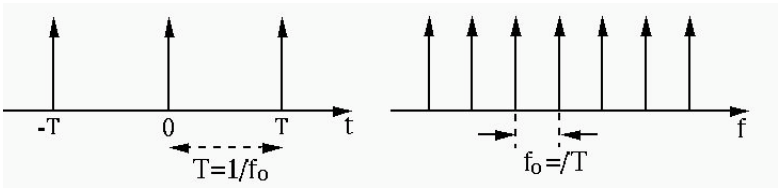

Figure 3.5 Impulse train and its spectrum

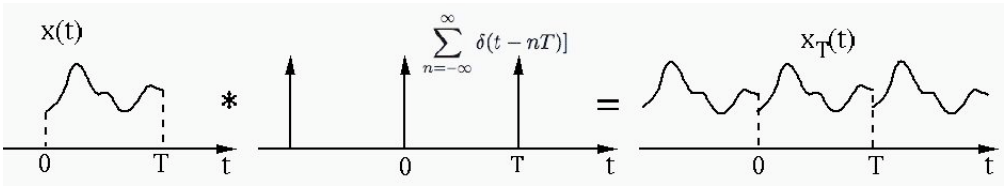

Figure 3.6 Generation of a periodic signal

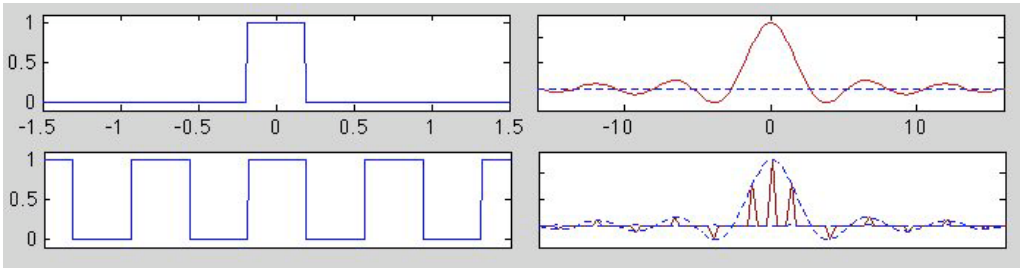

Figure 3.7 A periodic signal and its spectrum

The Fourier transform of this function is:

$$\mathcal{F}[comb(t)] = \int_{-\infty}^{\infty} comb(t)e^{-j2\pi ft}dt = \int_{-\infty}^{\infty} \left[\sum_{n=-\infty}^{\infty} \delta(t-nT)\right] e^{-j2\pi ft}dt$$
$$= \sum_{n=-\infty}^{\infty} \int_{-\infty}^{\infty} \delta(t-nT)e^{-j2\pi ft}dt = \sum_{n=-\infty}^{\infty} e^{-j2\pi nfT}$$

We let $f_0 = 1/T$ and apply Eq.1.28 to the equation above to get

$$\mathcal{F}[comb(t)] = \int_{-\infty}^{\infty} comb(t)e^{-j2\pi ft}dt = f_0 \sum_{n=-\infty}^{\infty} \delta(f-nf_0) = \frac{1}{T}\sum_{n=-\infty}^{\infty} \delta(f-n/T)$$

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:

$$x_T(t) = x(t) * \sum_{n=-\infty}^{\infty} \delta(t - nT) \tag{3.151}$$

The Fourier transform of this periodic signal can be found to be:

$$\mathcal{F}[x_T(t)] = \mathcal{F}[x(t) * \sum_{n=-\infty}^{\infty} \delta(t - nT)] = X(f)\, \mathcal{F}[\sum_{n=-\infty}^{\infty} \delta(t - nT)] \tag{3.152}$$

The second equal sign is due to the convolution theorem. Here the two Fourier transforms on the right-hand side above are, respectively:

$$X(f) = \mathcal{F}[x(t)] = \int_0^T x(t) e^{-j2\pi ft} dt \tag{3.153}$$

and (Eq.1.28)

$$\mathcal{F}[\sum_{n=-\infty}^{\infty} \delta(t - nT)] = \frac{1}{T} \sum_{k=-\infty}^{\infty} \delta(f - k/T) \tag{3.154}$$

We now have:

$$\begin{aligned} \mathcal{F}[x_T(t)] = X(f)\, \mathcal{F}[\sum_{n=-\infty}^{\infty} \delta(t - nT)] &= \int_0^T x(t) e^{-j2\pi ft} dt\, \frac{1}{T} \sum_{k=-\infty}^{\infty} \delta(f - k/T) \\ &= \sum_{k=-\infty}^{\infty} \frac{1}{T} \int_0^T x_T(t) e^{-j2\pi kt/T} dt\, \delta(f - k/T) \\ &= \sum_{k=-\infty}^{\infty} X[k]\, \delta(f - kf_0) \end{aligned} \tag{3.155}$$

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 $X[k]$, same as those in Eq.3.6. As an example, a square wave and its periodic version are shown respectively on top and bottom of Fig.3.7. 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.3.8 shows a set of typical signals and their Fourier spectra.

Example 3.3: In radio and TV broadcasting, a *carrier wave* $c(t) = \cos(2\pi f_c t)$ with *radio frequency (RF)* f_c is first *modulated* by the audio or video signal $s(t)$ before it is transmitted. In particular, in *amplitude modulation (AM)*, the modulation is carried out as a multiplication by a modulator (mixer):

$$x(t) = s(t)c(t) = s(t)\cos(2\pi f_c t) = s(t)\frac{1}{2}[e^{j2\pi f_c t} + e^{-j2\pi fct}] \tag{3.156}$$

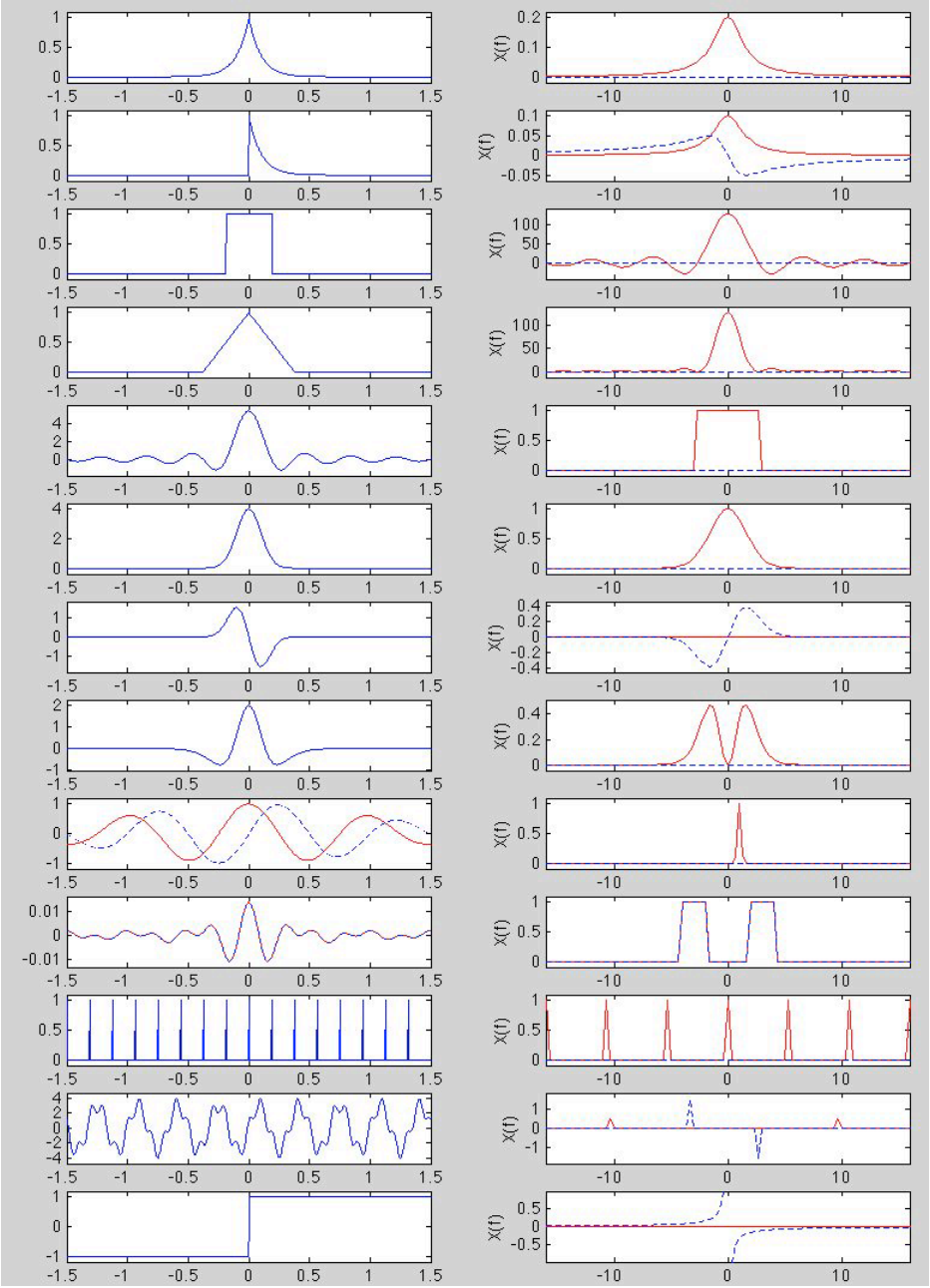

Figure 3.8 Examples of continuous-time Fourier transforms
A set of signals are shown on the left and their Fourier spectra are shown on the right (real and imaginary parts are shown in solid and dashed lines, respectively).

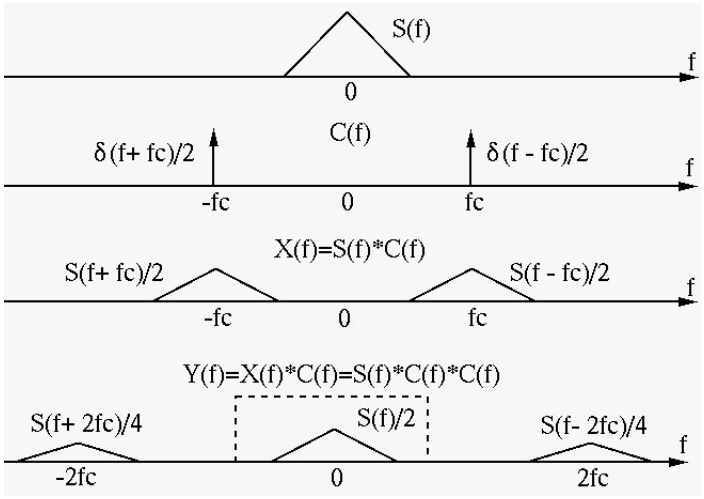

Figure 3.9 AM modulation and demodulation

This multiplication in time domain corresponds to a convolution in frequency domain:

$$X(f) = S(f) * C(f) = S(f) * \frac{1}{2}[\delta(f - f_c) + \delta(f + f_c)] = \frac{1}{2}[S(f - f_c) + S(f + f_c)] \tag{3.157}$$

We note that the bandwidth occupied by the signal is $\Delta f = 2f_m$, twice the highest frequency contained in the signal.

This modulated signal with RF frequency is transmitted and then received by a radio or TV receiver, where the audio or video signal needs to be separated from the carrier wave by a *demodulation* process, which can be easily implemented by another multiplication:

$$y(t) = x(t)\cos(2\pi f_c t) = s(t)\cos^2(2\pi f_c t) = \frac{s(t)}{2} + \frac{s(t)\cos(4\pi f_c t)}{2} \tag{3.158}$$

The signal $s(t)$ can then be obtained by a low-pass filter to remove the higher frequency component at $2f_c$. Note that this demodulation method requires the sinusoid $\cos(2\pi f_c t)$ used in the demodulator of the receiver is synchronous with that used in the modulator of the transmitter. If there is a phase difference between the two, the trigonometry relation used in the equation above is no longer valid. For this reason, alternative methods exist for the purpose of demodulation.

This process of both modulation and demodulation in frequency domain is illustrated in Fig.3.9.

3.2.6 The Uncertainty Principle

According to the property of time and frequency scaling (Eq.3.102), if a time function $x(t)$ is expanded ($a < 1$), its spectrum $X(f)$ will be compressed, and, 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 with in 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:

$$p_x(t) = \frac{|x(t)|^2}{||x(t)||^2} = \frac{|x(t)|^2}{\int_{-\infty}^{\infty} |x(t)|^2 dt} \tag{3.159}$$

As $p_x(t)$ satisfies these conditions

$$p_x(t) > 0 \qquad \text{and} \qquad \int_{-\infty}^{\infty} p_x(t)dt = 1 \tag{3.160}$$

it can be considered as a probability density function over variable t, and how the function $x(t)$ spreads over time, or the dispersion of $x(t)$, can be measured as the variance of this probability density $p_x(t)$:

$$\sigma_t^2 = \int_{-\infty}^{\infty} (t - \mu_t)^2 p_x(t)dt = \frac{1}{||x(t)||^2} \int_{-\infty}^{\infty} (t - \mu_t)^2 |x(t)|^2 dt \tag{3.161}$$

where μ_t is the mean of $p_x(t)$:

$$\mu_t = \int_{-\infty}^{\infty} t p_x(t)dt = \frac{1}{||x(t)||^2} \int_{-\infty}^{\infty} t|x(t)|^2 dt \tag{3.162}$$

The dispersion of the spectrum of the signal can also be similarly measured as:

$$\sigma_f^2 = \frac{1}{||X(f)||^2} \int_{-\infty}^{\infty} (f - \mu_f)^2 |X(f)|^2 df \tag{3.163}$$

with μ_f defined as:

$$\mu_f = \frac{1}{||X(f)||^2} \int_{-\infty}^{\infty} f|X(f)|^2 df \tag{3.164}$$

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 $x(t)$. Then we have*

$$\sigma_t^2 \sigma_f^2 \geq \frac{1}{16\pi^2} \tag{3.165}$$

Proof:
Without loss of generality, we assume $\mu_t = \mu_f = 0$, and consider

$$\sigma_t^2 \sigma_f^2 = \frac{1}{||x(t)||^4} \int_{-\infty}^{\infty} |tx(t)|^2 dt \int_{-\infty}^{\infty} |fX(f)|^2 df \tag{3.166}$$

Due to the time derivative property (Eq.3.118) and the Parseval's identity, we have

$$\frac{1}{j2\pi} \mathcal{F}[x'(t)] = fX(f) \tag{3.167}$$

and

$$\int_{-\infty}^{\infty} |fX(f)|^2 df = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} |x'(t)|^2 dt \tag{3.168}$$

Now the above becomes

$$\sigma_t^2 \sigma_f^2 = \frac{1}{4\pi^2 ||x(t)||^4} \int_{-\infty}^{\infty} |tx(t)|^2 dt \int_{-\infty}^{\infty} |x'(t)|^2 dt \tag{3.169}$$

Applying the Cauchy-Schwarz inequality (Eq.2.34), we get

$$\sigma_t^2 \sigma_f^2 \geq \frac{1}{4\pi^2 ||x(t)||^4} \left[\int_{-\infty}^{\infty} t\overline{x}(t) x'(t)\, dt \right]^2 \tag{3.170}$$

But since

$$[\, |x(t)|^2 \,]' = [x(t)\overline{x}(t)]' = x'(t)\overline{x}(t) + \overline{x}'(t)x(t) = 2\, Re[x'(t)\overline{x}(t)] \leq 2x'(t)\overline{x}(t) \tag{3.171}$$

we can replace $\overline{x}(t)x'(t)$ in the integrand by $[|x(t)|^2]'/2$ and get

$$\sigma_t^2 \sigma_f^2 \geq \frac{1}{4 \cdot 4\pi^2 ||x(t)||^4} \left[\int_{-\infty}^{\infty} t[|x(t)|^2]' dt \right]^2 \tag{3.172}$$

By integration by parts, the integral becomes

$$\int_{-\infty}^{\infty} t[|x(t)|^2]' dt = t|x(t)|^2 \,|_{-\infty}^{\infty} - \int_{-\infty}^{\infty} |x(t)|^2 dt = -\int_{-\infty}^{\infty} |x(t)|^2 dt \tag{3.173}$$

Here we have assumed $\lim_{|t| \leftarrow \infty} tx^2(t) = 0$. Substituting this back into the inequality, we finally get

$$\sigma_t^2 \sigma_f^2 \geq \frac{1}{4 \cdot 4\pi^2 ||x(t)||^4} \left[\int_{-\infty}^{\infty} |x(t)|^2 dt \right]^2 = \frac{1}{16\pi^2} \tag{3.174}$$

This effect is 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.

3.3 The Two-Dimensional Fourier Transform

3.3.1 Two-Dimensional Signals and Their Spectra

All signals considered so far are assumed to be one-dimensional time functions. However, a signal could also be a function over a 1D 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 two or three-dimensional 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 one-dimensional 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 spectrum of a 2-D signal $f(x, y)$ is:

$$F(u, v) = \int \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi(ux+vy)} dx\, dy \tag{3.175}$$

This is the forward transform where u and v represent two spatial frequencies (cycles per unit distance) along the directions of x and y in the 2-D space, respectively. The signal can be reconstructed by the inverse transform:

$$f(x, y) = \int \int_{-\infty}^{\infty} F(u, v) e^{j2\pi(ux+vy)} du\, dv \tag{3.176}$$

by which the signal is expressed as a linear combination of infinite number 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.

Same as the 1-D case, if a 2-D signal is periodic in each of the two dimensions, i.e., $f(x + X, y + Y) = f(x, y)$, where X and Y are the periods in the two spatial dimensions, it can be Fourier expanded to become:

$$f_{XY}(x, y) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} F[k, l] e^{-j2\pi(xku_o+ylv_o)} \tag{3.177}$$

Here $F[k, l]$ is the kl-th coefficient that can be obtained as:

$$F[k, l] = \frac{1}{XY} \int_0^X \int_0^Y f_{XY}(x, y) e^{j2\pi(kxu_o+lyv_o)} dx\, dy \tag{3.178}$$

where $u_o = 1/X$ and $v_o = 1/Y$ are the fundamental frequencies in x and y directions, which are also the intervals between any two consecutive frequency components in the spatial frequencies u and v, respectively. This discrete spectrum can also be represented as a 2-D function:

$$F(u, v) = \sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} F[k, l] \delta[u - ku_0] \delta[v - lv_0] \tag{3.179}$$

When the periods X and Y become infinite: $X \to \infty$ and $Y \to \infty$, i.e., the signal $f(x, y)$ becomes non-periodic, correspondingly in spatial frequency domain, we have $u_0 \to 0$ and $v_0 \to 0$, i.e., the spectrum becomes continuous.

3.3.2 Physical Interpretation

In the discussion of this subsection, we will always assume $f(x, y) = \overline{f}(x, y)$ is a real 2-D signal. The integrand in the 2-D Fourier transform is composed of two parts, the kernel function of the integral transform, which is also the orthogonal basis functions $\phi_{u,v}(x, y) = e^{j2\pi(ux+vy)}$, and the complex coefficient function $F(u, v)$. We first consider each of them individually.

- Complex exponential basis function $e^{j2\pi(ux+vy)}$:
  Let us define two vectors, one in the spatial domain, another in the spatial frequency domain:
  - Define a vector $\boldsymbol{r}$ associated with each spatial point (x, y):
    $$\boldsymbol{r} = [x, y]^T \tag{3.180}$$
  - Define a vector $\boldsymbol{w}$ associated with each frequency point (u, v):
    $$\boldsymbol{w} = [u, v]^T = w[u/w, v/w]^T = w\boldsymbol{n} \tag{3.181}$$
    where $w = \sqrt{u^2 + v^2}$ is the magnitude and $\boldsymbol{n} = [u/w, v/w]^T$ is the unit vector ($||\boldsymbol{n}|| = 1$) along the direction of $\boldsymbol{w}$.

  Now the 2-D basis function $\phi_{u,v}(x, y)$ can be written as:
  $$\phi_{u,v}(x, y) = e^{j2\pi(xu+yv)} = e^{j2\pi w(\boldsymbol{r}^T\boldsymbol{n})} = \cos(2\pi w(\boldsymbol{r}^T\boldsymbol{n})) + j\,\sin(2\pi w(\boldsymbol{r}^T\boldsymbol{n})) \tag{3.182}$$
  where $\boldsymbol{r}^T\boldsymbol{n}$ is the projection of a spatial point $\boldsymbol{r} = [x, y]^T$ onto the direction of $\boldsymbol{w}$. The value of the function $\cos(2\pi(xu + yv)) = \cos(2\pi w(\boldsymbol{r}^T\boldsymbol{n}))$ is the same for all spatial points $\boldsymbol{r} = (x, y)$ along each straight line perpendicular to the direction $\boldsymbol{n}$, as all such points have the same projection $\boldsymbol{r}^T\boldsymbol{n}$.
  In other words, the function $\cos(2\pi(ux + vy)) = \cos(2\pi w(\boldsymbol{r}^T\boldsymbol{n}))$ represents a planar sinusoid in the (x, y) plane with *frequency* $w = \sqrt{u^2 + v^2}$ and *direction* $\boldsymbol{n}$ with an angular difference $\theta = tan^{-1}(v/u)$ from the positive horizontal direction. The same argument can be made for the sine function of the imaginary part $\sin(2\pi w(\boldsymbol{r}^T\boldsymbol{n}))$.
  Two 2-D sinusoid functions $\cos(2\pi(3x + 2y))$ and $\cos(2\pi(2x + 3y))$ are shown in Fig.3.10.
- Complex coefficient function $F(u, v)$:

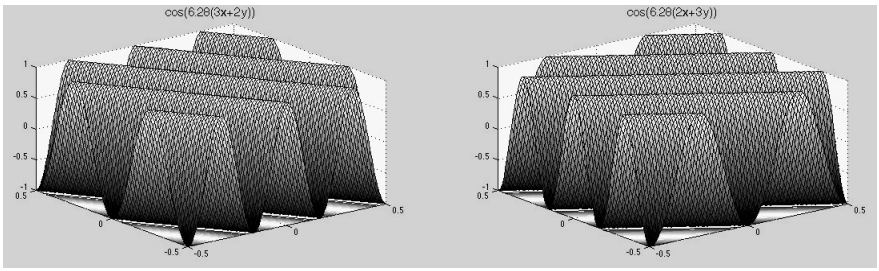

Figure 3.10 Different propagation directions of 2-D sinusoid $\cos(2\pi(ux+vy))$
In the plot on the left for $\cos(2\pi(3x+2y))$, we see $u=3$ cycles per unit length along x dimension (right side of plot) and $v=2$ per unit length along y. In the plot on the right for $\cos(2\pi(2x+3y))$, we see $u=2$ cycles per unit length along x and $v=3$ along y.

As the 2-D signal $f(x,y)$ is assumed real, its Fourier coefficient $F(u,v)$ can be written in terms of the real and imaginary parts:

$$\begin{aligned} F(u,v) &= \int\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi(xu+yv)}dx\,dy \\ &= \int\int_{-\infty}^{\infty} f(x,y)\cos(2\pi(xu+yv))dx\,dy - j\int\int_{-\infty}^{\infty} f(x,y)\sin(2\pi(xu+yv))dx\,dy \\ &= F_r(u,v) + jF_j(u,v) \end{aligned} \tag{3.183}$$

where

$$F_r(u,v) = \int\int_{-\infty}^{\infty} f(x,y)\cos(2\pi(xu+yv))dx\,dy \tag{3.184}$$

$$F_j(u,v) = -\int\int_{-\infty}^{\infty} f(x,y)\sin(2\pi(xu+yv))dx\,dy \tag{3.185}$$

Alternatively $F(u,v)$ can also be represented in polar form in terms of its *amplitude* $|F(u,v)|$ and *phase* $\angle F(u,v)$:

$$F(u,v) = |F(u,v)|e^{j\angle F(u,v)} \tag{3.186}$$

where

$$\begin{cases} |F(u,v)| = \sqrt{F_r^2(u,v)+F_j^2(u,v)} \\ \angle F(u,v) = tan^{-1}[F_j(u,v)/F_r(u,v)] \end{cases}, \quad \begin{cases} F_r(u,v) = |F(u,v)|\cos(\angle F(u,v)) \\ F_j(u,v) = |F(u,v)|\sin(\angle F(u,v)) \end{cases} \tag{3.187}$$

The real part $F_r(u,v)$ is even and the imaginary part $F_j(u,v)$ is odd:

$$\begin{aligned} F_r(-u,-v) &= F_r(u,v), \qquad F_r(u,-v) = F_r(-u,v) \\ F_j(-u,-v) &= -F_j(u,v), \qquad F_j(u,-v) = -F_j(-u,v) \end{aligned} \tag{3.188}$$

The magnitude $|F(u,v)|$ is even and the phase $\angle F(u,v)$ is odd:

$$\begin{aligned} |F(-u,-v)| &= |F(u,v)|, \qquad |F(u,-v)| = |F(-u,v)| \\ \angle F(-u,-v) &= -\angle F(u,v), \qquad \angle F(u,-v) = -\angle F(-u,v) \end{aligned} \tag{3.189}$$

Now a real signal $f(x,y)$ can be expanded in terms of the 2-D basis functions by the 2-D Fourier transform:

$$\begin{aligned} f(x,y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} F(u,v)e^{j2\pi(xu+yv)}du\,dv \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} [F_r(u,v)\cos(2\pi(ux+vy)) - F_j(u,v)\sin(2\pi(ux+vy))]du\,dv \\ &+ j\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} [F_r(u,v)\sin(2\pi(ux+vy)) + F_j(u,v)\cos(2\pi(ux+vy))]du\,dv \end{aligned} \tag{3.190}$$

However, as $f(x,y)$ is real, the imaginary part is zero, therefore we have:

$$\begin{aligned} f(x,y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |F(u,v)|\cos(2\pi(ux+vy) + \angle F(u,v))du\,dv \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |F(u,v)|\cos(2\pi w\boldsymbol{r}^T\boldsymbol{n} + \angle F(u,v))du\,dv \end{aligned} \tag{3.191}$$

We see that $f(x,y)$ is represented as a superposition of a set of infinite and uncountable 2-D spatial sinusoids $|F(u,v)|\cos(2\pi w\boldsymbol{r}^T\boldsymbol{n}) + \angle F(u,v))$ with

- **frequency** $w = \sqrt{u^2+v^2}$ and **direction** $\boldsymbol{n}$ ($\theta = tan^{-1}(v/u)$) determined by the position (u,v) of the coefficient $F(u,v)$, and
- **amplitude** $|F(u,v)| = \sqrt{F_r(u,v)^2 + F_j(u,v)}$ and **phase** $\angle F(u,v) = tan^{-1}(F_j(u,v)/F_r(u,v))$ determined by the complex value of the coefficient $F(u,v)$.

Eq. 3.191 can be further modified to gain some insight of the expansion of a 2-D signal. First, according to the symmetry properties in Eq.3.189, we have

- $|F(u,v)|$ is even: $|F(-u,-v)| = |F(u,v)|$ and $|F(u,-v)| = |F(-u,v)|$
- $\angle F(u,v)$ is odd: $\angle F(-u,-v) = -\angle F(u,v)$ and $\angle F(u,-v) = -\angle F(-u,v)$

Now Eq. 3.191 can be rewritten as

$$\begin{aligned} f(x,y) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} |F(u,v)|\cos(2\pi(ux+vy)+\angle F(u,v))du\,dv \\ &= 2\int_{0}^{\infty}\int_{0}^{\infty} |F(u,v)|\cos(2\pi(ux+vy)+\angle F(u,v))du\,dv \\ &+ 2\int_{-\infty}^{0}\int_{0}^{\infty} |F(u,v)|\cos(2\pi(ux-vy)+\angle F(u,-v))du\,dv \\ &= 2\int_{0}^{\infty}\int_{0}^{\infty} |F(u,v)|\cos(2\pi w(\boldsymbol{r}^T\boldsymbol{n})+\angle F(u,v))du\,dv \\ &+ 2\int_{-\infty}^{0}\int_{0}^{\infty} |F(u,v)|\cos(2\pi w(\boldsymbol{r}^T\boldsymbol{n}')+\angle F(u,-v))du\,dv \end{aligned} \quad (3.192)$$

Here $\boldsymbol{n}$ is the unit vector in the direction determined by the angle $\tan^{-1}(v/u)=\theta$, while $\boldsymbol{n}'$ is the unit vector in the direction determined by the angle $(\tau^{-1}(-v/u) = -\tan^{-1}(v/u) = -\theta$; and $w=\sqrt{u^2+v^2}$ is the spatial frequency represented by the coefficient $F(u,v)$. This equation is the 2-D version of Eq. 3.70. The first integral represents a 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:

$$X[k,l] = |X[k,l]|e^{j\angle X[k,l]} \quad (3.193)$$

where

$$\begin{cases} |X[k,l]| = \sqrt{X_r^2[k,l]+X_j^2[k,l]} \\ \angle X[k,l] = tan^{-1}[X_j[k,l]/X_r[k,l]] \end{cases} \quad \text{and} \quad \begin{cases} X_r[k,l] = |X[k,l]|\cos(\angle X[k,l]) \\ X_j[k,l] = |X[k,l]|\sin(\angle X[k,l]) \end{cases} \quad (3.194)$$

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:

$$\begin{aligned} x[m,n] &= \frac{1}{\sqrt{MN}}\sum_{l=0}^{N-1}\sum_{k=0}^{M-1} X[k,l]e^{j2\pi(\frac{mk}{M}+\frac{nl}{N})} \\ &= \frac{1}{\sqrt{MN}}\sum_{-M/2+1}^{M/2}\sum_{-N/2+1}^{N/2} |X[k,l]|\cos(2\pi(\frac{mk}{M}+\frac{nl}{N})+\angle X[k,l]), \\ &\qquad (m=0,\cdots,M-1,\ n=0,\cdots,N-1) \end{aligned} \quad (3.195)$$

3.3.3 Fourier Transform of Typical 2-D Functions

- The function below represents a planar wave in 2-D space:

$$f(x,y) = \cos(2\pi(3x-2y)) = \frac{1}{2}[e^{j2\pi(3x-2y)}+e^{-j2\pi(3x-2y)}] \quad (3.196)$$

and its 2-D Fourier spectrum can be found to be:

$$\begin{aligned}
F(u,v) &= \int\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi(ux+vy)}dx\,dy \\
&= \frac{1}{2}\int\int_{-\infty}^{\infty}[e^{j2\pi(3x-2y)} + e^{-j2\pi(3x-2y)}]e^{-j2\pi(ux+vy)}dx\,dy \\
&= \frac{1}{2}\int\int_{-\infty}^{\infty} e^{-j2\pi((u-3)x+(v+2)y)}dx\,dy + \frac{1}{2}\int\int_{-\infty}^{\infty} e^{-j2\pi((u+3)x+(v-2)y)}dx\,dy \\
&= \frac{1}{2}\int_{-\infty}^{\infty} e^{-j2\pi(u-3)x}dx\int_{-\infty}^{\infty} e^{-j2\pi(v+2)y}dy + \frac{1}{2}\int_{-\infty}^{\infty} e^{-j2\pi(u+3)x}dx\int_{-\infty}^{\infty} e^{-j2\pi(v-2)y}dy \\
&= \frac{1}{2}[\delta(u-3)\delta(v+2)] + \frac{1}{2}[\delta(u+3)\delta(v-2)] \qquad (3.197)
\end{aligned}$$

This signal and its spectrum are shown in Fig.3.11(a)

- The function below is a 2-D signal composed of three frequency components:

$$f(x,y) = 3\ \cos(2\pi 2x) + 2\ \cos(2\pi 3y) + \cos(2\pi 5(x-y)); \qquad (3.198)$$

and its 2-D Fourier spectrum is given below. The signal and its spectrum are shown in Fig.3.11(b).

$$\begin{aligned}
F(u,v) &= \int\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi(ux+vy)}dx\,dy \\
&= \frac{3}{2}[\delta(u-2)+\delta(u+2)]\delta(v) + \delta(u)[\delta(v-3)+\delta(v+3)] \\
&+ \frac{1}{2}[\delta(u-5)\delta(v+5) + \delta(u+5)\delta(v-5)] \qquad (3.199)
\end{aligned}$$

- The function below is a 2-D rectangular impulse:

$$f(x,y) = \begin{cases} 1 \text{ if } (-\frac{a}{2} < x < \frac{a}{2},\ -\frac{b}{2} < y < \frac{b}{2}) \\ 0 \text{ else} \end{cases} \qquad (3.200)$$

Note that this 2-D function can be separated to become $f(x,y) = f_x(x)f_y(y)$, where $f_x(x)$ and $f_y(y)$ are each a 1-D square impulse function. The spectrum is, not too surprisingly, 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. The signal and its spectrum are shown in Fig.3.11(c).

$$\begin{aligned}
F(u,v) &= \int\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi(ux+vy)}dx\,dy = \int_{-\infty}^{\infty} f_x(x)e^{-j2\pi ux}dx\ \int_{-\infty}^{\infty} f_y(y)e^{-j2\pi vy}dy \\
&= \int_{-a/2}^{a/2} e^{-j2\pi ux}dx\int_{-b/2}^{b/2} e^{-j2\pi vy}dy = \frac{sin(\pi ua)}{\pi u}\frac{sin(\pi vb)}{\pi v} \qquad (3.201)
\end{aligned}$$

- This function has a cylindrical shape, which cannot be separated to become a product of two 1-D functions:

$$f(x,y) = \begin{cases} 1\ x^2+y^2 < R^2 \\ 0\ else \end{cases} \qquad (3.202)$$

To find the spectrum, it is more convenient to use polar coordinate system in both spatial and frequency domains. Let

$$\begin{cases} x = r\ \cos\theta, & y = r\ \sin\theta \\ r = \sqrt{x^2 + y^2},\ \theta = tan^{-1}(y/x) \end{cases} \tag{3.203}$$

$$dx\,dy = rdr\,d\theta \tag{3.204}$$

and

$$\begin{cases} u = \rho\ \cos\phi, & v = \rho\ \sin\phi \\ \rho = \sqrt{u^2 + v^2},\ \phi = tan^{-1}(v/u) \end{cases} \tag{3.205}$$

$$du\,dv = \rho d\rho\,d\phi \tag{3.206}$$

then we have:

$$\begin{aligned} F(u,v) &= \int\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi(ux+vy)}dx\,dy = \int_0^R [\int_0^{2\pi} e^{-j2\pi r\rho(\cos\theta\cos\phi+\sin\theta\sin\phi)}d\theta]rdr \\ &= \int_0^R [\int_0^{2\pi} e^{-j2\pi r\rho\cos(\theta-\phi)}d\theta]rdr = \int_0^R [\int_0^{2\pi} e^{-j2\pi r\rho\cos\theta}d\theta]rdr \end{aligned} \tag{3.207}$$

To continue, we need to use the 0th order Bessel function $J_0(x)$ defined as

$$J_0(x) = \frac{1}{2\pi}\int_0^{2\pi} e^{-jx\ \cos\theta}d\theta \tag{3.208}$$

which is related to the 1st order Bessel function $J_1(x)$ by

$$\frac{d}{dx}(x\,J_1(x)) = x\,J_0(x) \tag{3.209}$$

i.e.

$$\int_0^x x\,J_0(x)dx = x\,J_1(x) \tag{3.210}$$

Substituting $2\pi r\rho$ for x, we have

$$F(u,v) = F(\rho,\phi) = \int_0^R 2\pi r\,J_0(2\pi r\rho)dr = \frac{1}{\rho}R\,J_1(2\pi\rho R) \tag{3.211}$$

We see that the spectrum $F(u,v) = F(\rho,\phi)$ is independent of angle ϕ and therefore is central symmetric sinc-like function. This signal and its spectrum are shown in Fig.3.11(d).

- The function in Eq. 3.202 can also be defined in frequency domain as an ideal low-pass filter:

$$F(u,v) = \begin{cases} 1\ u^2 + v^2 < R^2 \\ 0\ else \end{cases} \tag{3.212}$$

When the spectrum of a 2-D signal is multiplied by this filter, all its frequency components inside the radius R from the DC component at the origin will be kept, while all higher frequency components outside the circle are suppressed

to zero. The inverse transform of this ideal filter is the same 2-D sinc-like function shown in Eq.3.211. The filtering effect of this ideal low-pass filter will be discussed later.

- The 1-D Gaussian function discussed in the previous chapter can be expanded to become a 2-D Gaussian function, which is separable:

$$f(x,y) = \frac{1}{a^2} e^{-\pi(x^2+y^2)/a^2} = \frac{1}{a} e^{-\pi(x/a)^2} \frac{1}{a} e^{-\pi(y/a)^2} \tag{3.213}$$

The spectrum of this function can be found as:

$$\begin{aligned} F(u,v) &= \int\int_{-\infty}^{\infty} f(x,y) e^{-j2\pi(ux+vy)} dx\, dy \\ &= \frac{1}{a} \int_{-\infty}^{\infty} e^{-\pi(x/a)^2} e^{-j2\pi ux} dx \;\; \frac{1}{a} \int_{-\infty}^{\infty} e^{-\pi(y/a)^2} e^{-j2\pi vy} dy \\ &= e^{-\pi(au)^2} e^{-\pi(av)^2} \end{aligned} \tag{3.214}$$

The last equation is due to Eq.3.149. 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. This 2-D Gaussian function and its Gaussian spectrum are shown in Fig.3.11(e).

3.4 Some Applications of the Fourier Transform

3.4.1 Frequency Response Function of Continuous LTI Systems

In the discussions above, we mostly considered the Fourier transform applied to a given signal $x(t)$ to produce its spectrum $\mathcal{F}[x(t)] = X(f)$ that characterizes the frequency contents of the signal. However, the Fourier transform can also be used to characterize the LTI systems. Recall that the output of a continuous LTI system can be found as the convolution of the input $x(t)$ and the impulse response function $h(t)$ of the system (Eq. 1.66 in Chapter 1):

$$y(t) = \mathcal{O}[x(t)] = h(t) * x(t) = \int_{-\infty}^{\infty} h(\tau) x(t-\tau) d\tau \tag{3.215}$$

and this convolution can also be more conveniently represented in frequency domain as a multiplication (Eq.3.117):

$$Y(f) = H(f)X(f)$$

In particular, when the input be a complex exponential $x(t) = e^{j2\pi ft}$, then the output becomes:

$$\begin{aligned} y(t) = \mathcal{O}[e^{j2\pi ft}] &= \int_{-\infty}^{\infty} h(\tau) e^{j2\pi f(t-\tau)} d\tau = e^{j2\pi ft} \int_{-\infty}^{\infty} h(\tau) e^{-j2\pi f\tau} d\tau \\ &= H(f)\, e^{j2\pi ft} = |H(f)| e^{j2\pi H(f)} e^{j2\pi ft} = |H(f)| e^{j(2\pi tt + \angle H(f))} \end{aligned} \tag{3.216}$$

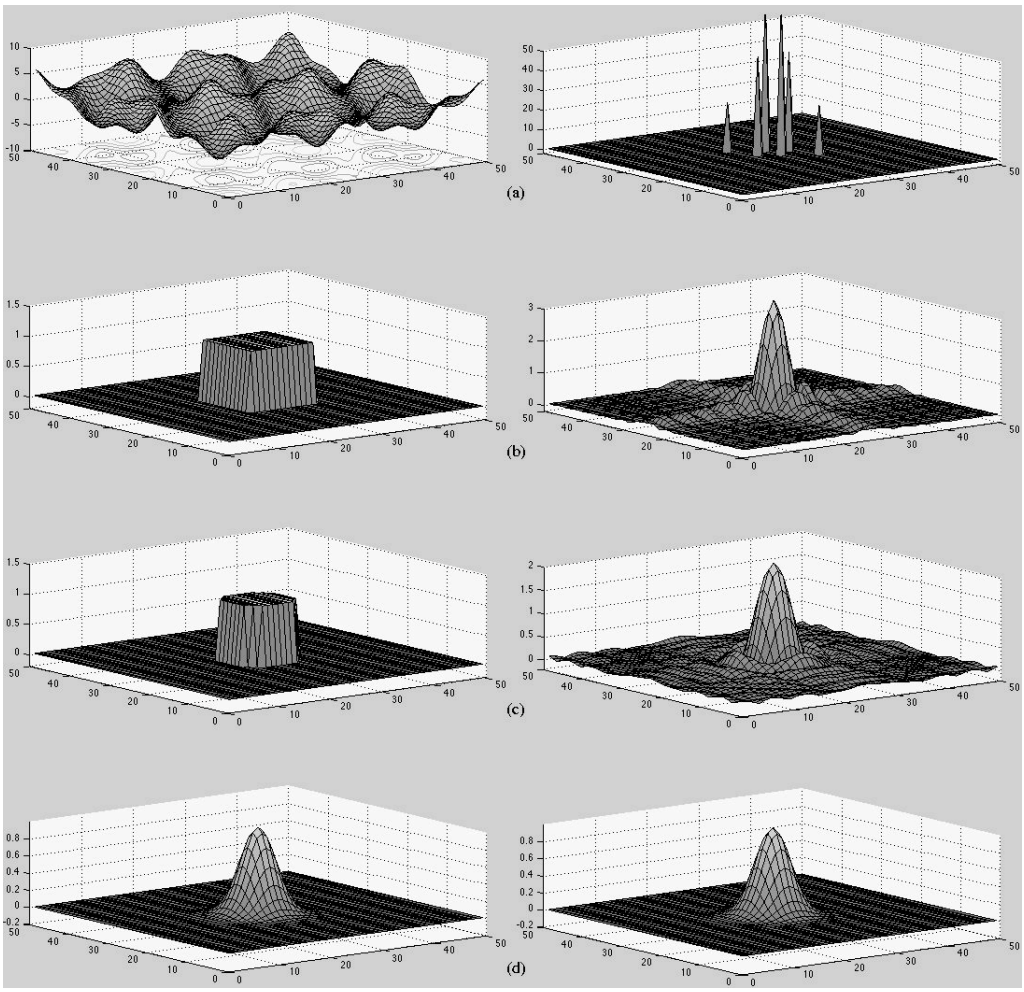

Figure 3.11 Some 2-D signals (left) and their spectra (right)

Here $H(f)$ happens to be the Fourier transform of the impulse response function $h(t)$, the *frequency response function (FRF)* of the system:

$$H(f) = \int_{-\infty}^{\infty} h(t)e^{-j2\pi ft}dt = \mathcal{F}[h(t)] \tag{3.217}$$

This equation is an eigenequation indicating that the effect of the LTI system applied to a sinusoidal input, the eigenfunction of the system, is the same as a multiplication of the input by a constant $H(f)$, the eigenvalue. Also, as the complex exponential input $x(t) = e^{j2\pi ft}$ is independent of any specific $h(t)$), it is the eigenfunction of *all* LTI systems.

Due to the linearity of the LTI system, the input-output relationship $y(t) = \mathcal{O}[x(t)] = h(t) * x(t)$ can be expressed as:

$$\begin{aligned} y(t) &= y_r(t) + jy_j(t) = \mathcal{O}[x(t)] = \mathcal{O}[x_r(t) + jx_j(t)] \\ &= \mathcal{O}[x_r(t)] + j\mathcal{O}[x_j(t)] = h(t) * x_r(t) + jh(t) * x_i(t) \end{aligned} \tag{3.218}$$

where $x_r(t)$ and $x_j(t)$ are the real and imaginary parts of $x(t)$ and $y_r(t)$ and $y_j(t)$ are the real and imaginary parts of $y(t)$, respectively. Also, as the system is always assumed to be real, i.e., its impulse response function $h(t)$ is real, we have:

$$y_r(t) = \mathcal{O}[x_r(t)] = h(t) * x_r(t), \quad \text{and} \quad y_j(t) = \mathcal{O}[x_j(t)] = h(t) * j_r(t) \tag{3.219}$$

i.e., the real (imaginary) part of the system output is its response of the real (imaginary) part of the input. Now taking the real part on both sides of Eq.3.216, we get:

$$\mathcal{O}[Re[e^{j2\pi ft}]] = \mathcal{O}[\cos 2\pi ft = Re[\ |H(f)|e^{j(2\pi ft + \angle H(f))}\ ] = |H(f)|\cos(2\pi ft + \angle H(f)) \tag{3.220}$$

Similarly, taking the imaginary part of Eq.3.216, we get:

$$\mathcal{O}[\sin 2\pi ft] = |H(f)|\sin(2\pi ft + \angle H(f)) \tag{3.221}$$

In other words, the response of any LTI system to a sinusoidal input is the same sinusoid with its amplitude scaled by the magnitude $|H(f)|$ of the FRF, and its phase shifted by the phase angle $\angle H(f)$ of the FRF.

The results above can be summarized in Fig.3.12, by which the essential role of the Fourier transform in LTI system analysis is illustrated. We see that an LTI system can be described by its impulse response function $h(t)$ in time domain, or its frequency response function $H(f) = \mathcal{F}[h(t)]$ in frequency domain. Correspondingly, the system's response 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, where $Y(f) = \mathcal{F}[y(t)]$, $X(f) = \mathcal{F}[x(t)]$, and $H(f) = \mathcal{F}[h(t)]$ are the Fourier transforms of $y(t)$, $x(t)$, and $h(t)$, respectively. Although both the forward and inverse Fourier transforms are needed for the frequency domain method, we gain some benefits not possible in time domain. Most obviously, the response of an LTI system to an input $x(t)$ can be much more conveniently obtained in frequency domain by a multiplication, instead of the corresponding convolution in time domain. For this reason, the Fourier transform is a powerful tool in the analyze and design of LTI systems. Specifically, given any two of the three variables $X(f)$, $H(f)$ and $Y(f)$, the third can be obtained.

1. Find output $Y(f)$ given input $X(f)$ to the system $H(f)$.
   This operation can also be carried out equivalently as a convolution in time domain.
2. Find system $H(f)$ given input $X(f)$ to output $Y(f)$

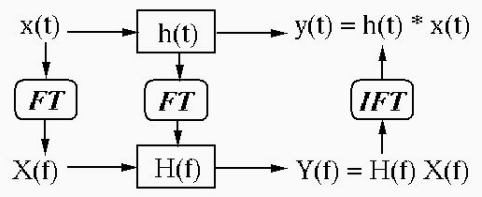

Figure 3.12 Signal through system in time and frequency domains

This kind of problems are called *system identification*, by which an unknown system $H(f)$ can be identified from its input $X(f)$ and output $Y(f)$. This process can also be applied to the design of a system $H(f)$ (called a *filter* in signal processing) given the input and desired output.

3. Find input $X(f)$ given output $Y(f)$ of the system $H(f)$
   This kind of problems are called *signal restoration*, by which the original signal $X(f)$ is obtained based on the observed output $Y(f)$ of a measuring system $H(f)$.

Note that it is rather difficult to carry out the last two operations in time domain.

As an example, consider a very important type of LTI systems that is described by a linear constant-coefficient ordinary differential equation (LCCDE). Here the input $x(t)$ and output $y(t)$ of the system are related by the differential equation as:

$$\sum_{k=0}^{n} a_k \frac{d^k}{dt^k} y(t) = \sum_{k=0}^{m} b_k \frac{d^k}{dt^k} x(t) \tag{3.222}$$

If the input is assumed to be a complex exponential $x(t) = e^{j2\pi ft}$, then according to Eq.3.216, the output is also a complex exponential $y(t) = H(f)e^{j2\pi ft}$ with a complex coefficient $H(f)$, the FRF of the system. Note that the output here is the *steady state response* of the system to the complex exponential input, when the *transient response* is completely attenuated. Substituting such $x(t)$ and $y(t)$ into the differential equation above we get:

$$H(f) \sum_{k=0}^{n} a_k (j2\pi f)^k e^{j2\pi ft} = \sum_{k=0}^{m} b_k (j2\pi f)^k e^{j2\pi ft} \tag{3.223}$$

Now the frequency response function of the system can then be obtained as

$$H(f) = \frac{\sum_{k=0}^{m} b_k (j2\pi f)^k}{\sum_{k=0}^{n} a_k (j2\pi f)^k} = \frac{N(f)}{D(f)} \tag{3.224}$$

where $N(f) = \sum_{k=0}^{m} b_k (j2\pi f)^k$ and $D(f) = \sum_{k=0}^{n} a_k (j2\pi f)^k$ are the numerator and denominator of $H(f)$, respectively.

More generally, consider an input $x(t) = Xe^{j2\pi ft}$, here the complex coefficient $X = |X|e^{j\angle X}$ is called *phasor* that represents the amplitude $|X(f)|$ and phase $\angle X(f)$ (but not the frequency) of the input signal. The corresponding output can also be assumed to be a complex exponential $y(t) = Ye^{j2\pi ft}$ with a phasor

coefficient $Y = |Y|e^{j\angle Y}$ for the amplitude and phase of the output. Substituting $x(t) = Xe^{j2\pi ft}$ and $y(t) = Ye^{j2\pi ft}$ into the differential equation, we get

$$\sum_{k=0}^{n} a_k \frac{d^k}{dt^k} y(t) = Y \sum_{k=0}^{n} a_k (j2\pi f)^k e^{j2\pi ft} = \sum_{k=0}^{m} b_k (j2\pi f)^k e^{j2\pi ft} = X \sum_{k=0}^{m} b_k \frac{d^k}{dt^k} x(t) \tag{3.225}$$

Now the same frequency response function of the system can be found as the ratio between the output phasor Y and the input phasor X:

$$H(f) = \frac{Y}{X} = \frac{\sum_{k=0}^{m} b_k (j2\pi f)^k}{\sum_{k=0}^{n} a_k (j2\pi f)^k} = \frac{N(f)}{D(f)} \tag{3.226}$$

This is the general definition of the frequency response function of a given LTI system.

The result can be further generalized much beyond sinusoidal inputs to cover any input so long as it can be expressed as a linear combination of a set of sinusoids (inverse Fourier transform):

$$x(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi ft} df \tag{3.227}$$

Here the weighting function $X(f) = \mathcal{F}[x(t)]$, the phasor for frequency component $e^{j2\pi ft}$, is of course the Fourier spectrum of $x(t)$. As the system is linear, we can get the output as:

$$y(t) = \mathcal{O}[x(t)] = \int_{-\infty}^{\infty} X(f) \mathcal{O}[e^{j2\pi ft}] df = \int_{-\infty}^{\infty} X(f) H(f) e^{j2\pi ft} df \tag{3.228}$$

We see that the output $y(t)$ happens to be the inverse Fourier transform of $H(f)X(f)$, i.e.,

$$y(t) = \mathcal{F}^{-1}[Y(f)] = \mathcal{F}^{-1}[H(f)X(f)] \tag{3.229}$$

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 of the input and the frequency response function, i.e., $Y(f) = H(f)X(f)$. This is, of course, the same as the conclusion of the convolution theorem.

Similar result can also be obtained for a periodic input $x_T(t + T) = x_T(t)$ with period $T = 1/f_0$, which can be Fourier series expanded to become:

$$x_T(t) = \sum_{k=-\infty}^{\infty} X[k] e^{j2k\pi f_0 t} \tag{3.230}$$

where the coefficient $X[k]$, the phasor of the frequency component $e^{j2k\pi f_0 t}$, is the kth Fourier coefficient of the signal. The corresponding output can be found

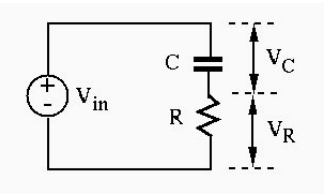

Figure 3.13 An RC circuit

to be:

$$y(t) = \mathcal{O}[x_T(t)] = \sum_{k=-\infty}^{\infty} X[k]\mathcal{O}[e^{j2k\pi f_0 t}] = \sum_{k=-\infty}^{\infty} X[k]H(kf_0)e^{j2k\pi f_0 t} = \sum_{k=-\infty}^{\infty} Y[k]e^{j2k\pi f_0 t} \tag{3.231}$$

Of course this is the Fourier expansion of the output with phasor

$$Y[k] = H(kf_0)X[k] \tag{3.232}$$

where $H(kf_0)$ is the frequency response function $H(f)$ of the system evaluated at $f = kf_0$.

Example 3.4: In a circuit composed of a resistor R and a capacitor C as shown in Fig.3.13, the voltage across both R and C in series is the input $x(t) = v_{in}(t)$, and the voltage across C is the output $y(t) = v_C(t)$. Find the impulse response function and the frequency response function of the system.

- Set up the differential equation for the system:
  The current through both C and R is $i(t) = C\, dv_C(t)/dt = C\dot{y}(t)$. According to Ohm's law, the voltage across R is $v_R(t) = Ri(t) = RC\, \dot{y}(t)$. Also, according to Kirchhoff's voltage law, the input voltage $x(t)$ is the sum of $v_R(t)$ and $v_C(t)$:

  $$v_R(t) + v_C(t) = RC\dot{y}(t) + y(t) = v_{in}(t) = x(t)$$

  i.e.,

  $$\tau\dot{y}(t) + y(t) = x(t), \quad \text{or} \quad \dot{y}(t) + \frac{1}{\tau}y(t) = \frac{1}{\tau}x(t)$$

  where $\tau = RC$ is the time constant of the system.
- Find step response o a unit step input $x(t) = u(t)$:
  - Find homogeneous solution $y_h(t)$ when the right-hand side is zero:
    Assume $y_h(t) = Ae^{st}$ and get $\dot{y}_h(t) = sAe^{st}$, now the homogeneous differential equation becomes:

    $$(s\tau + 1)Ae^{st} = 0, \quad \text{i.e.,} \quad s = -\frac{1}{\tau}$$

    and we get $y_h(t) = Ae^{-t/\tau}$.

- 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 input $x(t) = u(t)$:

$$y(t) = y_h(t) + y_p(t) = Ae^{-t/\tau} + 1$$

  We further assume initially $y(t)\big|_{t=0} = y(0) = y_0$ and get $A = y_0 - 1$, and the complete response to $x(t) = u(t)$ is:

$$y(t) = [(y_0 - 1)e^{-t/\tau} + 1]u(t) = [1 - e^{-t/\tau}]e^{-t/\tau} + y_0 e^{-t/\tau}$$

  The first term is the charging process of the capacitor due to the step input and the second term represents the discharge of the initial voltage on the capacitor. In particular, when $y_0 = 0$, we have:

$$y(t) = (1 - e^{-t/\tau})u(t)$$

- Find impulse response $h(t)$ to an impulse input $x(t) = \delta(t)$:
  Due to the linearity of the system $y(t) = \mathcal{O}[x(t)]$, and under zero initial condition, we can take derivative on both sides to get $\dot{y}(t) = \mathcal{O}[\dot{x}(t)]$. (Derivative $\dot{x}(t) = \lim_{\triangle t=0}[x(t + \triangle t) - x(t)]/\triangle t$ is a linear combination of $x(t + \triangle t)$ and $x(t)$.) Therefore, based on the previous result, we see that if the input is $\dot{x}(t) = dt\, u(t)/dt = \delta(t)$, the corresponding output $\dot{y}(t)$ is the impulse response $h(t)$:

$$\begin{aligned} h(t) = \frac{d}{dt}y(t) &= \frac{d}{dt}[(1 - e^{-t/\tau})u(t)] \\ &= \frac{1}{\tau}e^{-t/\tau}u(t) + (1 - e^{-t/\tau})\delta(t) = \frac{1}{\tau}e^{-t/\tau}u(t) \end{aligned}$$

  Note that the second term is zero as $1 - e^{-t/\tau} = 0$ when $t = 0$.
- Alternative approach to Find $h(t)$:
  As the system is causal, $h(t) = 0$ for all $t < 0$ when the input is zero, we can assume

$$h(t) = f(t)u(t) = \begin{cases} f(t) & t > 0 \\ 0 & t < 0 \end{cases}$$

  and have:

$$\dot{h}(t) = \dot{f}(t)u(t) + f(t)\dot{u}(t) = \dot{f}(t)u(t) + f(0)\delta(t)$$

  where $f(t)$ is a function to be determined. Now the differential equation above becomes:

$$\tau\dot{f}(t)u(t) + \tau f(0)\delta(t) + f(t)u(t) = \delta(t)$$

Separating terms containing $u(t)$ and $\delta(t)$, we get two equations:

$$\begin{cases} \tau\dot{f}(t) + f(t) = 0 \\ f(0) = 1/\tau \end{cases}$$

This homogeneous equation with an initial condition can be solved to get

$$f(t) = \frac{1}{\tau}e^{-t/\tau}$$

and the impulse response is

$$h(t) = f(t)u(t) = \frac{1}{\tau}e^{-t/\tau}u(t)$$

- Find the frequency response function $H(f)$:
  When the input is a complex exponential $x(t) = e^{j2\pi ft}$, the output also takes the form of a complex exponential $y(t) = H(f)e^{j2\pi ft}$ and $\dot{y}(t) = j2\pi fH(f)e^{j2\pi ft}$. Substituting these into the equation we get:

$$\tau\dot{y}(t) + y(t) = H(f)(j2\pi f\tau + 1)e^{j2\pi ft} = x(t) = e^{j2\pi ft}$$

  Solving this we get the frequency response function:

$$H(f) = \frac{1}{j2\pi f\tau + 1}$$

- Verify $H(f)$ is the Fourier transform of $h(t)$:

$$\begin{aligned} \mathcal{F}[h(t)] &= \int_{-\infty}^{\infty} h(t)e^{-j2\pi ft}dt = \int_{-\infty}^{\infty} \frac{1}{\tau}e^{-t/\tau}u(t)e^{-j2\pi ft}dt \\ &= \frac{1}{\tau}\int_{0}^{\infty} e^{-(j2\pi f+1/\tau)t}dt = \frac{1}{j2\pi f\tau + 1} = H(f) \end{aligned}$$

3.4.2 Signal Filtering in Frequency Domain

From the signal processing point of view, the process of a signal $x(t)$ going through an LIT system $h(t)$, as either a convolution $y(t) = h(t) * x(t)$ in time, or a multiplication $Y(f) = H(f)X(f)$ in frequency domain, can be treated as a filtering process, while the LTI system is treated as a filter. While the two representations in time and frequency domains are equivalent in the sense that the total amount of energy or information is conserved (the Parseval's theorem), the frequency representation has the benefit that the signal can be manipulated by various filtering methods in frequency domain, which in many ways are more advantageous and convenient than the manipulations in time domain. Due to the frequency locality gained by the transform (with the cost of losing the temporal locality), we can modify and manipulate the phase as well as the magnitude of the frequency components of the signal:

$$|Y(f)| = |H(f)|X(f)|, \quad \text{and} \quad \angle Y(f) = \angle H(f)\angle X(f) \tag{3.233}$$

We now consider the filtering effects in both aspects.

First, we consider various filtering schemes 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 filters can be classified as one of these different types: low-pass, high-pass, band-pass, and band-stop filters, as illustrated in Fig.3.14. Moreover, sometimes it may also be the case that $|H(f)| = c$ 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:

- **Cutoff frequency**
  As the cutoff frequency f_c the magnitude of $H(f)|$ is reduced to $1/\sqrt{2}$ of the maximum magnitude:

$$|H(f_c)| = \frac{1}{\sqrt{2}}|H_{max}|, \quad \text{i.e.} \quad |H(f_c)|^2 = \frac{1}{2}|H_{max}|^2$$

  where $|H_{max}|$ is the maximum magnitude of the frequency response function $H(f)$ at some peak frequency. (When the passing band of the filter is flat, $|H(f)| = |H_{mmax}|$ occurs within a range of frequencies.) In other words, at the cutoff frequency f_c, the power of the filtered signal (proportional to its magnitude squared) is reduced to half of maximum power, and therefore the cutoff frequency is also called the *half-power frequency*.
- **Bandwidth** $\triangle f$
  The bandwidth is the interval between two cutoff frequencies of a bandpass filter:

$$\triangle f = f_{c1} - f_{c2}$$

  where f_{c1} and f_{c2} are the two cutoff frequencies on either side of the peak frequency. When Bandwidth is used to describe a lowpass filter, the lower cutoff frequency f_{c2} will be replace by 0 and $\triangle f = f_c$.

Next, we consider how the phase angle $\angle H(f)$ of the filter affects the filtering process. We first consider a simple signal containing two sinusoidal components of frequencies $f_1 = 2$ and $f_2 = 4$ Hz, respectively, as shown in the top panel of Fig.3.15:

$$x(t) = \cos(2\pi f_1 t) + \cos(2\pi f_2 t) = \cos(2\pi 2t) + \cos(2\pi 4t)$$

Assume the filter $H(f)$ is all-pass with a unity gain $|H(f)| = 1$ and a linear phase, $\angle H(f) = -2\pi f\tau$, i.e., the phase shift of a frequency component is proportional to its frequency f, then the phase shifts of the sinusoids of 2 and 4 Hz are $4\pi\tau$ and $8\pi\tau$, respectively, and their relative positions in time remain the same, and consequently the shape of the signal remains the same before and after filtering, except it delayed in time by a constant amount τ, as shown in the middle panel of Fig.3.15.

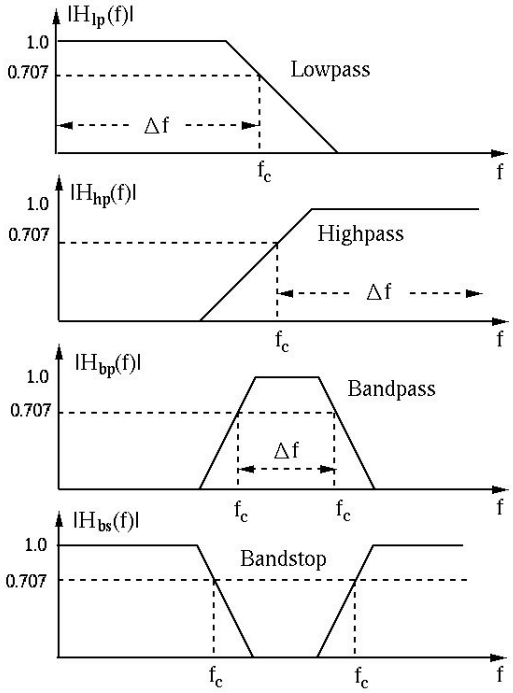

Figure 3.14 Illustration of four different types of filters (lowpass, highpass, bandpass and bandstop)

This result can be generalized to any signal

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

The output of the filter corresponding to any frequency component $e^{j2\pi ft}$ of the signal is:

$$\mathcal{O}[e^{j2\pi ft}] = H(f)e^{j2\pi ft} = |H(f)|e^{j\angle H(f)}e^{j2\pi ft} = e^{j2\pi f(t-\tau)}$$

and the output corresponding to $x(t)$ is the linear combination of all these outputs:

$$y(t) = \int_{-\infty}^{\infty} X(f) e^{j2\pi f(t-\tau)} df = x(t-\tau)$$

which is just a delayed version of the input. Of course we also realize this result is actually stated by the time shift property of the Fourier transform:

$$\mathcal{F}[x(t-\tau)] = X(f)e^{-j2\pi f\tau}$$

We can also conclude that the time delay caused by a linear phase filter $H(f)$ can be simply obtained from its phase $\angle H(f)$ as:

$$\tau = -\frac{\angle H(f)}{f} = -\frac{2\pi f\tau}{f} \tag{3.234}$$

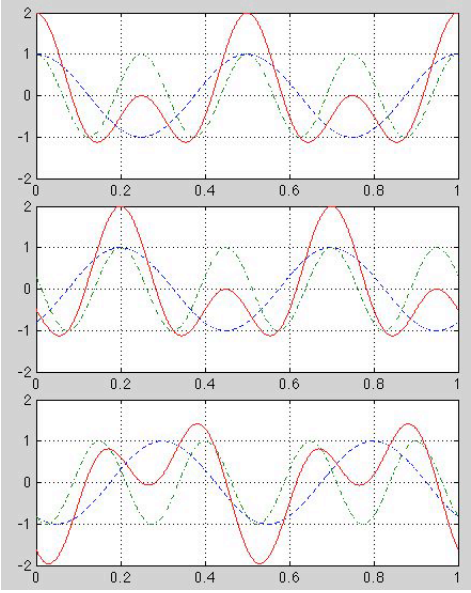

Figure 3.15 Filtering with linear and non-linear phase shift
The original signal containing two sinusoidal components (top panel) of frequencies $f_1 = 2$ and $f_2 = 4$, respectively, is filtered linearly (middle panel) and nonlinearly (bottom panel). Th signals before and after are plotted in solid lines while the two frequency components are plotted in dashed lines.

This is called the *phase delay* of the linear phase filter.

On the other hand, when the phase of the filter is not a linear function of frequency, the relative positions in time of the frequency components in the signal will no longer remain the same, and the shape of the signal will be distorted by the filtering. Again, in the example above, if the phase shift of the filter is $6\pi\tau$ for both frequencies f_1 and f_2 (i.e., $-6\pi\tau$ for $-f_1$ and $-f_2$), then the shape of the filtered signal looks very different from the original, as shown in the bottom panel of Fig.3.15. Another example is shown in Fig.3.16, where a square impulse signal is filtered, first by a linear phase filter (top), which caused no distortion but a pure time delay, 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:

$$\tau_g = -\frac{d\angle H(f)}{df} \tag{3.235}$$

This is defined as the *group delay* of the non-linear filter, representing approximately the time delay of a group of frequency components within a narrow band around this frequency f.

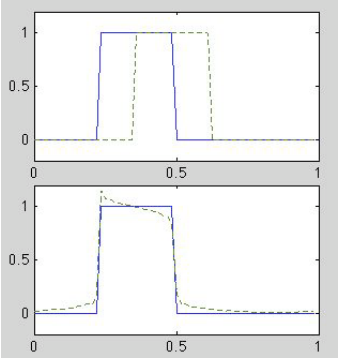

Figure 3.16 Filtering with linear and constant phase shift

The frequency response function $H(f)$ can also be 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:

$$LmH(f) = 20\ \log_{10}|H(f)| \tag{3.236}$$

The unit of the log-magnitude is *decibel* or dB.

Based on the log-magnitude representation, we have:

$$20\ \log_{10}\frac{|H(f_c)|}{|H_{max}|} = 20\ \log_{10}\frac{1}{\sqrt{2}} = -3.01\ dB \approx -3\,dB \tag{3.237}$$

In other words, 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 $H(f) = N(f)/[D_1(f)D_2(f)$, then we can get:

$$LmH(f) = Lm\left[\frac{N(f)}{D_1(f)D_2(f)}\right] = LmN(f) - LmD_1(f) - LmD_2(f)$$

which becomes the same as the phase plot:

$$\angle H(f) = \angle\left[\frac{N(f)}{D_1(f)D_2(f)}\right] = \angle N(f) - \angle D_1(f) - \angle D_2(f)$$

Similarly, the Bode plot of a cascade of two filters can be found as the sum of the their individual Bode plots.

3.4.3 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 $1/\pi t$:

$$\mathcal{H}[x(t)] = \hat{x}(t) = x(t) * \frac{1}{\pi t} = \frac{1}{\pi}\int_{-\infty}^{\infty}\frac{x(\tau)}{t-\tau}d\tau = \frac{1}{\pi}\int_{-\infty}^{\infty}\frac{x(t-\tau)}{\tau}d\tau \tag{3.238}$$

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 defined as:

$$\mathcal{H}[x(t)] = \frac{1}{\pi}\lim_{\epsilon\to 0}[\int_{-\infty}^{-\epsilon}\frac{x(t-\tau)}{\tau}d\tau + \int_{\epsilon}^{\infty}\frac{x(t-\tau)}{\tau}d\tau] \tag{3.239}$$

In particular, if $x(t) = c$ is a constant, the sum of the two integrals above is zero, indicating the Hilbert transform, as a linear operator, will remove the DC component of the signal.

The Hilbert transform can be much more conveniently discussed in frequency domain as a multiplication corresponding to the time convolution in Eq.3.238. To do so, we assume $X(f) = \mathcal{F}[x(t)]$ and find the spectrum of $1/\pi t$ by applying the property of time-frequency duality to the Fourier transform of the sign function $sgn(t)$ (Eq.3.64):

$$\mathcal{F}^{-1}[\frac{1}{\pi t}] = -j\,Sgn(f) = -j\begin{cases} -1 & (f<0) \\ 0 & (f=0) \\ 1 & (f>0) \end{cases} = \begin{cases} j & (f<0) \\ 0 & (f=0) \\ -j & (f>0) \end{cases} \tag{3.240}$$

Now the Hilbert transform can be expressed in frequency domain as a multiplication:

$$\hat{X}(f) = \mathcal{F}[\hat{x}(t)] = -j\;Sgn(f)\,X(f) = \begin{cases} jX(f) & (f<0) \\ 0 & (f=0) \\ -jX(f) & (f>0) \end{cases} \tag{3.241}$$

The effect of the Hilbert transform applied of a signal $x(t)$ becomes very 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:

$$\mathcal{F}[\mathcal{H}^n[x(t)]] = [-j\,Sgn(f)]^n X(f) \tag{3.242}$$

In particular, as $Sgn^2(f) = 1$, we have

$$[-j\,Sgn(f)]^2 = -1, \qquad [-j\,Sgn(f)]^3 = j\,Sgn(f), \qquad [-j\,Sgn(f)]^4 = 1 \tag{3.243}$$

Correspondingly in time domain, we have:

$$\mathcal{H}[x(t)] = \hat{x}(t), \quad \mathcal{H}^2[x(t)] = -x(t), \quad \mathcal{H}^3[x(t)] = -\hat{x}(t), \quad \mathcal{H}^4[x(t)] = x(t) \tag{3.244}$$

In other words, 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, i.e., this is the inverse Hilbert transform:

$$\begin{cases} \mathcal{H}[x(t)] = x(t) * 1/\pi t = \hat{x}(t) \\ \mathcal{H}^{-1}[\hat{x}(t)] = \mathcal{H}^3[\hat{x}(t)] = -\mathcal{H}[\hat{x}(t)] = x(t) \end{cases} \tag{3.245}$$

Example 3.5: Consider a simple sinusoid:

$$\cos(2\pi f_0 t) = \frac{e^{j2\pi f_0 t} + e^{-j2\pi f_0 t}}{2} = \frac{1}{2}e^{j2\pi f_0 t} + \frac{1}{2}e^{-j2\pi f_0 t} \tag{3.246}$$

Here the coefficients for $f = f_0 > 0$ and $f = -f_0 < 0$ are both $1/2$. When the Hilbert transform is applied to the signal, the coefficient $1/2$ for $f < 0$ is rotated by 90° to become $e^{j\pi/2}/2$ while the other $1/2$ for $f > 0$ is rotated by -90° to become $e^{-j\pi/2}/2$, i.e., the transformed signal becomes:

$$\mathcal{H}[\cos(2\pi ft)] = \frac{e^{-j2\pi}}{2}e^{j2\pi f_0 t} + \frac{e^{j\pi/2}}{2}e^{-j2\pi f_0 t} = \sin(2\pi ft) \tag{3.247}$$

Similarly we have

$$\mathcal{H}[\sin(2\pi ft)] = -\cos(2\pi ft), \quad \mathcal{H}[-\cos(2\pi ft)] = -\sin(2\pi ft), \quad \mathcal{H}[-\sin(2\pi ft)] = \cos(2\pi ft) \tag{3.248}$$

Next let us 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 $f < 0$. Given a usual signal $x(t)$, we can always construct an analytic signal by multiplying its spectrum $X(f) = \mathcal{F}[x(t)]$ with a step function $2u(f)$ in frequency domain:

$$X_a(f) = X(f)2u(f) = \begin{cases} 0 & (f < 0) \\ X(0) & (f = 0) \\ 2X(f) & (f > 0) \end{cases} \tag{3.249}$$

and the corresponding analytic signal can be obtained as

$$\begin{aligned} x_a(t) &= \mathcal{F}^{-1}[X_a(f)] = \mathcal{F}^{-1}[X(f)] * \mathcal{F}^{-1}[2u(f)] = x(t) * [\delta(t) + \frac{j}{\pi t}] \\ &= x(t) + j\, x(t) * \frac{1}{\pi t} = x(t) + j\, \hat{x}(t) \end{aligned} \tag{3.250}$$

where the inverse Fourier transform of the unit step spectrum $u(f)$ is

$$\mathcal{F}^{-1}[u(f)] = \frac{1}{-j2\pi t} + \frac{1}{2}\delta(-t) = \frac{j}{2\pi t} + \frac{1}{2}\delta(t) \tag{3.251}$$

which can be obtained by the time-frequency duality applied to the spectrum of the unit step $u(t)$ in time given in Eq.3.65.

Alternatively, an analytic signal can also be initially defined in time domain by Eq.3.250, and if we take the Fourier transform on both sides, we have

$$X_a(f) = X(f) + j\,\hat{X}(f) = X(f) + j \begin{cases} jX(f) & (f<0) \\ 0 & (f=0) \\ -jX(f) & (f>0) \end{cases} = \begin{cases} 0 & (f<0) \\ X(0) & (f=0) \\ 2X(f) & (f>0) \end{cases} \tag{3.252}$$

where $\hat{X}_a(f) = \mathcal{F}[\hat{x}(t)]$.

When the signal $x(t)$ is real, the real and imaginary parts of its spectrum are even and odd, respectively ($X_r(f) = X_r(-f)$ and $X_j(f) = -X_j(-f)$), i.e., its spectrum $X(f)$ is Hermitian:

$$X(f) = \overline{X}(-f) \tag{3.253}$$

This means that the corresponding analytic signal $x_a(t) = x(t) + j\,\hat{x}(t)$ will still contain 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 $X(f) = X_a(f)/2$ from Eq.3.240. When $f < 0$, we have

$$X(f) = \overline{X}(-f) = \overline{X}(|f|) = \frac{1}{2}\overline{X}_a(|f|) \tag{3.254}$$

Combining these two cases, we have:

$$X(f) = \frac{1}{2}\begin{cases} X_a(f) & (f>0) \\ \overline{X}_a(|f|) & (f<0) \end{cases} = \frac{X_a(f) + \overline{X}_a(-f)}{2} \tag{3.255}$$

the last equality is due to the fact that $\overline{X}_a(-f) = 0$ when $f > 0$ and $X_a(f) = 0$ when $f < 0$.

Example 3.6: Given $x(t) = \cos(\omega_0 t)$, we can construct an analytic signal

$$x_a(t) = x(t) + j\,\hat{x}(t) = \cos(\omega_0 t) + j\,\sin(\omega_0 t) = e^{j\omega_0 t} \tag{3.256}$$

with spectrum $X_a(f) = \delta(\omega - \omega_0)$. Similarly, if $y(t) = \sin(\omega_0 t)$, the corresponding analytic signal is

$$y_a(t) = y(t) + j\,\hat{y}(t) = \sin(\omega_0 t) - j\,\cos(\omega_0 t) = -je^{j\omega_0 t} \tag{3.257}$$

with spectrum $Y_a(f) = -j\delta(\omega - \omega_0)$. In both cases, the negative half of the spectrum is zero.

Example 3.7: From the previous discussion regarding the modulation and demodulation in AM broadcasting, we see that the bandwidth $\Delta\omega = 2\omega_m$ taken

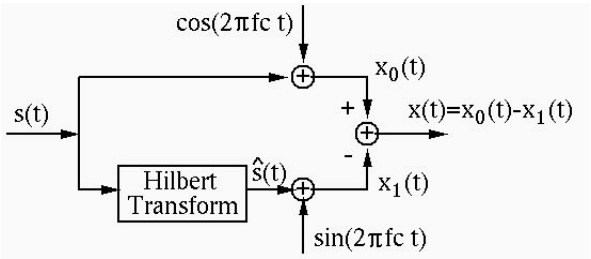

Figure 3.17 Single sideband modulation using Hilbert transform

by a transmission is twice of the highest frequency contained in the signal, one sideband of ω_m on each side of the carrier frequency ω_c (double sideband). However, in order to efficiently use the broadcasting spectrum as a limited resource, it is desirable to minimize the bandwidth needed for the radio or TV transmission. And *single-sideband modulation (SSB)* is such a method by which the bandwidth is reduced by half (from $2\omega_m$ to ω_m). One way to implement SSB is to use the idea of the Hilbert transform and analytic signals, taking advantage of the fact that the negative half of the spectrum of an analytic signal is completely zero and therefore does not need to be transmitted.

Specifically, an analytic signal is first constructed based on the real signal $s(t)$ to be transmitted:

$$s_a(t) = s(t) + j\,\hat{s}(t) \tag{3.258}$$

where $\hat{s}(t) = \mathcal{H}[x(t)]$ is the Hilbert transform of $s(t)$. Using this analytic signal to modulate a carrier frequency $2\pi f_c$, represented as an complex exponential $e^{j2\pi f_c}$, we get $s_a(t)e^{j2\pi f_c t}$ and then transmit its real part:

$$\begin{aligned} x(t) &= Re[s_a(t)e^{j2\pi f_c t}] = Re[(s(t) + j\,\hat{s}(t))\,(\cos(2\pi f_c t) + j\sin(2\pi f_c t))] \\ &= s(t)\cos(2\pi f_c t) - \hat{s}(t)\sin(2\pi f_c t) = x_0(t) - x_1(t) \end{aligned} \tag{3.259}$$

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.3.17. To show that the bandwidth of the modulated signal $x(t)$ indeed has only a single sideband spectrum, we consider this modulation in frequency domain:

$$\begin{aligned} X(f) &= S(f) * \frac{1}{2}[\delta(f - f_c) + \delta(f + f_c)] - \hat{S}(f) * \frac{1}{2j}[\delta(f - f_c) - \delta(f + f_c)] \\ &= \frac{1}{2}[S(f - f_c) + S(f + f_c)] - \frac{1}{2j}[\hat{S}(f - f_c) - \hat{S}(f + f_c)] \\ &= X_0(f) - X_1(f) \end{aligned} \tag{3.260}$$

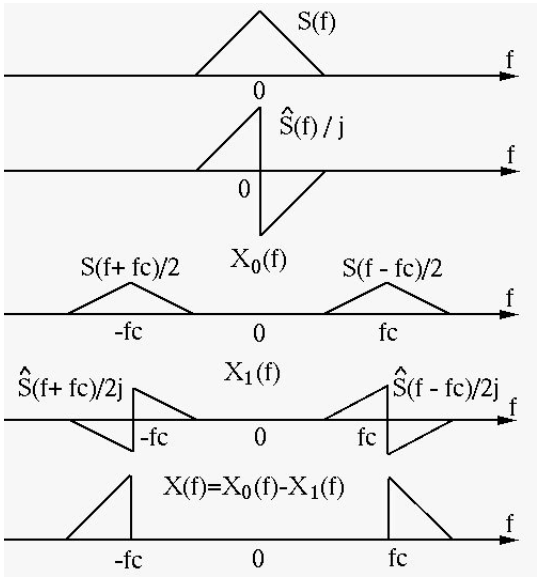

Figure 3.18 Spectra

The spectra of the signals in the process are shown in Fig.3.18. We can also confirm this result by considering the following four cases:

$$\begin{aligned}
&f - f_c < 0 (f < f_c): \quad \hat{S}(f - f_c) = jS(f - f_c),\ X(f - f_c) = 0 \\
&f - f_c > 0 (f > f_c): \quad \hat{S}(f - f_c) = -jS(f - f_c),\ X(f - f_c) = 2S(f - f_c) \\
&f + f_c < 0 (f < -f_c): \hat{S}(f + f_c) = jS(f + f_c),\ X(f + f_c) = 2S(f + f_c) \\
&f + f_c > 0 (f > -f_c): \hat{S}(f + f_c) = -jS(f + f_c),\ X(f + f_c) = 0.
\end{aligned} \tag{3.261}$$

It is therefore clear that the bandwidth of this modulated signal $x(t)$ is indeed reduced by half.

3.4.4 Radon Transform and Image Restoration from Projections

The Radon transform is an integral transform that integrates a 2-D function $f(x, y)$ along a certain direction t specified by an angle θ from the positive x direction to obtain a 1-D function $g_\theta(s)$, where s is a variable along the direction perpendicular to t. The function $g_\theta(s)$ can be considered as a projection of $f(x, y)$ onto 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:

$$g(y) = \int_{-\infty}^{\infty} f(x, y)\, dx, \quad \text{or} \quad g(x) = \int_{-\infty}^{\infty} f(x, y)\, dy$$

If such projections is available for all directions θ, the resulting projections can be considered as a 2-D function $g(s, \theta)$, and the original 2-D function can be

reconstructed by the inverse Radon transform. Therefore the forward and inverse Radon transforms can be expressed as:

$$\begin{cases} g(s,\theta) = \mathcal{R}[f(x,y)] \\ f(x,y) = \mathcal{R}^{-1}[g(s,\theta)] \end{cases} \tag{3.262}$$

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 image can also be obtained based on a sequence of such slices along one particular direction. 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), then the detected intensity I can be obtained according to this simple model:

$$I = I_o exp\left(-\int_L f(x,y)\, dt\right)$$

Here t is the integral variable along the pathway L of the X-ray through the tissue. Given I, the cross section of the tissue can be obtained as:

$$f(x,y) = \ln\left(I_o/I\right)$$

We first formulate the forward Radon transform. The straight line L along which the projection of a 2-D function is obtained can be specified by the following equation:

$$x\cos\theta + y\sin\theta - s = 0 \tag{3.263}$$

with two parameters s for the distance between L to the origin and θ for the angle between the normal direction of L and the horizontal axis of the space, as shown in Fig.3.20 (left).

The forward Radon transform: The Radon transform of a given 2-D function $f(x,y)$ is defined as a 1-D integral along such a straight line:

$$g(s,\theta) = \mathcal{R}[f(x,y)] = \int\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\theta + y\sin\theta - s)\, dx\, dy \tag{3.264}$$

where $-\infty < s < \infty$ and $0 \le \theta < 2\pi$. We see that the Radon transform converts a 2-D spatial function $f(x,y)$ into a function $g(s,\theta)$ in a 2-D parameter space.

A new coordinate system (s,t) of the 2-D space can be obtained by rotating the (x,y) coordinate system by an angle θ:

$$\begin{cases} s = x\cos\theta + y\sin\theta \\ t = -x\sin\theta + y\cos\theta \end{cases} \quad \text{or} \quad \begin{cases} x = s\cos\theta - t\sin\theta \\ y = s\sin\theta + t\cos\theta \end{cases} \tag{3.265}$$

Note that $x^2 + y^2 = s^2 + t^2$. In this new (s,t) system, the Radon transform can be expressed as a 1-D integral along the direction of t:

$$g(s,\theta) = \mathcal{R}[f(x,y)] = \int_{-\infty}^{\infty} f(s\cos\theta - t\sin\theta, s\sin\theta + t\cos\theta)\, dt \tag{3.266}$$

Projection-slice theorem: The 1-D Fourier transform of the Radon transform $g(s,\theta) = \mathcal{R}[f(x,y)]$ with respective to s is equal to the slice of the 2-D

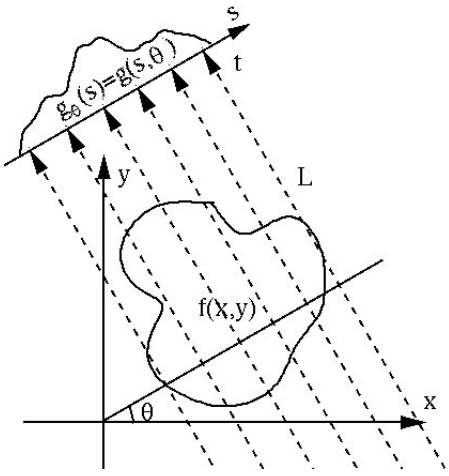

Figure 3.19 Radon transform

Fourier transform $F(u,v) = \mathcal{F}[f(x,y)]$ through the origin along the direction θ:

$$G(w,\theta) = \mathcal{F}[g(s,\theta)] = F_\theta(u,v) \tag{3.267}$$

where $F_\theta(u,v)$ denotes a slice of $F(u,v)$ through the origin along direction θ.

Proof: Consider the 1-D Fourier transform of the Radon transform $g(s,\theta) = \mathcal{R}[f(x,y)]$ with respect to s:

$$G(w,\theta) = \mathcal{F}[g(s,\theta)] = \int_{-\infty}^{\infty} g(s,\theta) e^{-j2\pi ws}\, ds$$

where w is the spatial frequency of $f(x,y)$ along the direction of s, and θ is treated as a parameter in the Fourier transform. Substituting the expression of $g(s,\theta)$ in Eq.3.264 into the above equation, we get:

$$\begin{aligned} G(w,\theta) &= \int_{-\infty}^{\infty} \left[\int\int_{-\infty}^{\infty} f(x,y)\delta(x\cos\theta + y\sin\theta - s)\, dx\, dy \right] e^{-j2\pi ws}\, ds \\ &= \int\int_{-\infty}^{\infty} f(x,y) \left[\int_{-\infty}^{\infty} \delta(x\cos\theta + y\sin\theta - s)\, e^{-j2\pi ws}\, ds \right] dx\, dy \\ &= \int\int_{-\infty}^{\infty} f(x,y)\, e^{-j2\pi w(x\cos\theta + y\sin\theta)}\, dx\, dy \\ &= F(w\cos\theta,\ w\sin\theta) = F_\theta(u,v) \end{aligned}$$

where $F(w\cos\theta,\ w\sin\theta) = F_\theta(u,v)$ 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, thus Eq.3.267 is proved.

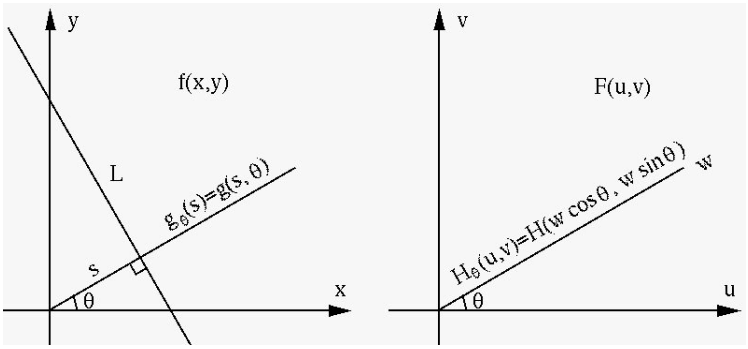

Figure 3.20 Radon transform and projection-slice theorem

The inverse Radon transform: Given its Radon transform $g(s,\theta)$, the original 2-D signal $f(x,y)$ can be reconstructed by the inverse transform:

$$f(x,y) = \mathcal{R}^{-1}[g(s,\theta)] = \frac{1}{2\pi^2}\int_0^{\pi}\int_{-\infty}^{\infty}\left[\frac{\partial}{\partial s}g(s,\theta)\right]\frac{1}{x\cos\theta + y\sin\theta - s}\,ds\,d\theta \tag{3.268}$$

Or in polar form:

$$f(r,\phi) = \frac{1}{2\pi^2}\int_0^{\pi}\int_{-\infty}^{\infty}\left[\frac{\partial}{\partial s}g(s,\theta)\right]\frac{1}{r\cos(\phi-\theta) - s}\,ds\,d\theta \tag{3.269}$$

where

$$\begin{cases} x = r\cos\phi \\ y = r\sin\phi \end{cases} \quad \text{or} \quad \begin{cases} r = \sqrt{x^2+y^2} \\ \phi = \tan^{-1}(y/x) \end{cases}$$

Proof:

The inverse Fourier transform

$$f(x,y) = \int\int_{-\infty}^{\infty} F(u,v)e^{j2\pi(ux+vy)}\,du\,dv$$

can also be expressed in polar form. We let

$$\begin{cases} u = w\cos\theta \\ v = w\sin\theta \end{cases} \quad \text{or} \quad \begin{cases} w = \sqrt{u^2+v^2} \\ \theta = \tan^{-1}(v/u) \end{cases}$$

then the Fourier spectrum $F(u,v)$ can be written as $F(w,\theta)$ and the inverse transform above becomes:

$$\begin{aligned} f(x,y) &= \int_0^{2\pi}\int_0^{\infty} F(w,\theta)e^{j2\pi w(x\cos\theta + y\sin\theta)}\,w\,dw\,d\theta \\ &= \int_0^{\pi}\int_{-\infty}^{\infty} F(w,\theta)e^{j2\pi w(x\cos\theta + y\sin\theta)}\,|w|dw\,d\theta \end{aligned}$$

But according to the projection-slice theorem, $F(w,\theta)$ in the inner integral with respect to w, a slice of $F(u,v)$ along the direction θ, 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:

$$\begin{aligned} f(x, y) &= \int_0^{\pi} \left[\int_{-\infty}^{\infty} |w| G(w, \theta) e^{j2\pi w(x\cos\theta + y\sin\theta)} \, dw \right] d\theta \\ &= \int_0^{\pi} g'(x\cos\theta + y\sin\theta, \theta) \, d\theta \end{aligned} \tag{3.270}$$

Here we have defined $g'(s, \theta)$ as the inverse Fourier transform of $|w|G(w, \theta)$:

$$\begin{aligned} g'(s, \theta) &= g'(x\cos\theta + y\sin\theta, \theta) = \mathcal{F}^{-1}[|w|G(w, \theta)] \\ &= \int_{-\infty}^{\infty} |w| G(w, \theta) e^{j2\pi w(x\cos\theta + y\sin\theta)} \, dw \end{aligned}$$

As $|w|G(w, \theta)$ can be considered as a filtering process of $g(s, \theta)$ by a filter $|w|$ (a high-pass filter) in frequency domain, and $g'(s, \theta)$ is a simply a filtered version of $g(s, \theta)$. The absolute value $|w|$ can be written as w multiplied by the sign function $sgn(w)$: $|w| = w \; sgn(w)$, and due to convolution theorem of the Fourier transform and Eq.3.118 and 3.127, we have:

$$\begin{aligned} g'(s, \theta) &= \mathcal{F}^{-1}[wG(w, \theta)] * \mathcal{F}^{-1}[sgn(w)] \\ &= \left[\frac{1}{j2\pi} \frac{d}{ds} g(s, \theta) \right] * \left[\frac{1}{-j\pi s} \right] = \frac{1}{2\pi^2} \int_{-\infty}^{\infty} \left[\frac{d}{dt} g(t, \theta) \right] \frac{1}{s - t} dt \end{aligned} \tag{3.271}$$

Comparing this expression with the definition of the Hilbert transform in Eq.3.238, we see that the filtered version $g'(s, \theta)$ is also the Hilbert transform of $\partial g(s, \theta)/partials/2\pi$:

$$g'(s, \theta) = \mathcal{H}[\frac{1}{2\pi} \frac{\partial}{\partial s} g(s, \theta)]$$

Substituting this result back into the equation above for $f(x, y)$, we get

$$f(x, y) = \frac{1}{2\pi^2} \int_0^{\pi} \int_{-\infty}^{\infty} \left[\frac{d}{dt} g(t, \theta) \right] \frac{1}{s - t} \, dt \, d\theta$$

Replacing s by $x\cos\theta + y\sin\theta$ and then t by s, we get Eq.3.268. This completes the proof.

In practice, the inverse Radon transform can be carried out based on Eq.3.270, instead of Eq.3.268 or 3.269, in the following steps:

1. Fourier transform of $g(s, \theta)$ with respect to s for all directions θ:

$$G(w, \theta) = \mathcal{F}[g(s, \theta)]$$

2. Filtering in frequency domain by $|w|$:

$$G'(w, \theta) = |w| G(w, \theta)$$

3. Inverse Fourier transform:

$$g'(s,\theta) = \mathcal{F}^{-1}[G'(w,\theta)]$$

4. Summation of $g'(x\cos\theta + y\sin\theta, \theta)$ over all directions θ (called "back projection"):

$$f(x,y) = \int_0^\pi g'(s,\theta)\, d\theta = \int_0^\pi g'(x\ \cos\theta + y\ \sin\theta, \theta)\, d\theta$$

We make two additional comments:

- The filtering in step 2 above can also be carried out in spatial domain by convolution.
- For most signals, their higher frequency components contain little energy and are therefore more susceptible to noise (lower signal-to-noise ratio). On the other hand, as the magnitude of the filter $|w|$ increases linearly as a function of frequency, it has the effect of amplifying noise. For this reason, the filter is typically modified so that its magnitude is reduced in the high frequency range.

Here we show two examples of Radon transform, both forward transform for projection and the inverse transform for reconstruction. The first example is a shape in black-white image, while the second example is a gray scale image, as shown in Fig.3.21. In both cases, the projections $g(s,\theta)$ (2nd from left) of all 180 projections, one degree apart, of the image $f(x,y)$ (1st on the left) are obtained. The image is then reconstructed, first without filtering by $|w|$ (3rd from left) and then with filtering (right). We see that the binary shape and the gray scale image are both almost perfectly reconstructed.

Example 3.8: First consider the Radon transform of a 2-D Gaussian function $f(x,y) = e^{-(x^2+y^2)} = e^{-(s^2+t^2)}$:

$$g(s,\theta) = \int_{-\infty}^{\infty} e^{-(s^2+t^2)} dt = e^{-s^2} \int_{-\infty}^{\infty} e^{-t^2} dt = \sqrt{\pi}\, e^{-s^2}$$

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

$$f(x,y) = \cos(2\pi(2x+3y)) = \frac{1}{2}[e^{j2\pi(2x+3y)} + e^{-j2\pi(2x+3y)}]$$

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

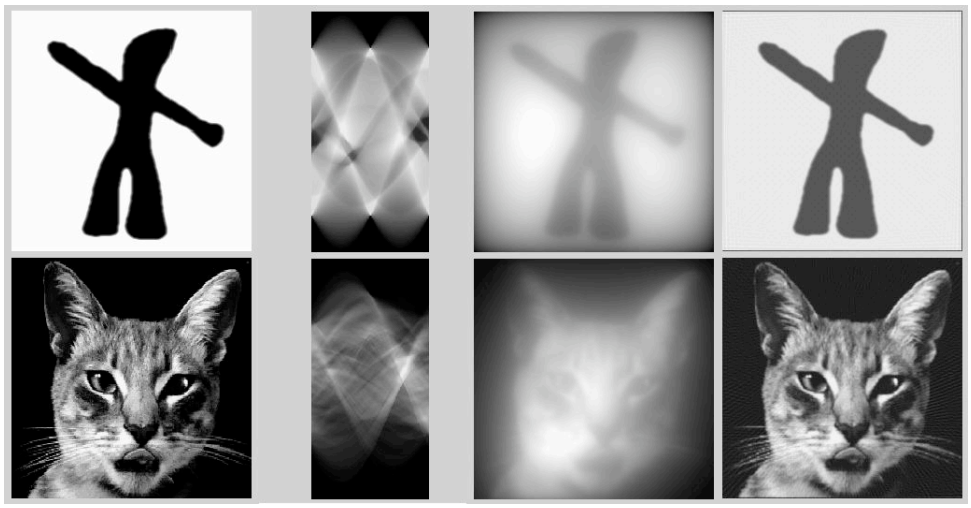

Figure 3.21 Forward and inverse Radon transform
From left to right: Original image $f(x, y)$, Radon projections $g(s, \theta)$, Back project without filtering, Back projection with filtering.

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:

$$\begin{aligned} \mathcal{G}[e^{j2\pi(2x+3y)}] &= e^{j2\pi s(2\cos\theta+3\sin\theta)} \int_{-\infty}^{\infty} e^{j2\pi t(-2\sin\theta+3\cos\theta)} dt \\ &= e^{j2\pi s(2\cos\theta+3\sin\theta)} \delta(-2\sin\theta + 3\cos\theta) \end{aligned} \tag{3.272}$$

Similarly we can get:

$$\mathcal{G}[e^{-j2\pi(2x+3y)}] = e^{-j2\pi s(2\cos\theta+3\sin\theta)} \delta(2\sin\theta - 3\cos\theta)$$

Adding these two results we get

$$\mathcal{G}[\cos(2\pi(2x + 3y))] = \cos(2\pi s(2\cos\theta + 3\sin\theta))\delta(2\sin\theta - 3\cos\theta)$$

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 for the Radon transform is perpendicular to the propagation direction of the plane wave. In this case the the Radon transform is a delta function (due to the infinite integral of a constant along the direction of L), weighted by a sinusoidal function of s along the direction of propagation. When $\theta \neq \phi$, the integrand in Eq.3.272 along L is a sinusoid with frequency $3\cos\theta - 2\sin\theta$, and the infinite integral is always zero.

Before consider the inverse Radon transform, we first define an back-projection operator as:

$$b(x,y) = \mathcal{B}[g(s,\theta)] = \int_0^\pi g(s,\theta)d\theta = \int_0^\pi g(x\cos\theta + y\sin\theta, \theta)d\theta \tag{3.273}$$

The back-projection can also be expressed in polar form:

$$b(r,\phi) = \mathcal{B}[g(s,\theta)] = \int_0^\pi g(r\cos(\theta-\phi), \theta)d\theta \tag{3.274}$$

where $r = \sqrt{x^2+y^2}$ and $\phi = \tan^{-1}(y/x)$, or equivalently, $x = r\cos\phi$ and $y = r\sin\phi$.

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);               % size of projection, d=1.414*n
    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 by F 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;           % origin of x-y plane is in image center
                t=fix(x*c+y*s)+d2;
                im(i,j)=im(i,j)+g(t); % back projection
           end
        end
    end
end
```

3.5 Problems

1. Show that the Fourier transform of the step function $u(t)$ same as in Eq.3.65 can be obtained by:

$$\mathcal{F}[u(t)] = \lim_{a\to 0} \mathcal{F}[e^{-at}u(t)] = \lim_{a\to 0} \frac{a}{a^2+\omega^2} + \lim_{a\to 0} \frac{-j\omega}{a^2+\omega^2}$$

Hint: The first term approaches $\delta(f)/2$, i.e.,

$$\lim_{a\to 0} \frac{a}{a^2+\omega^2} = \begin{cases} \infty & f = 0 \\ 0 & f \neq 0 \end{cases} \qquad \text{and} \qquad \int_{-\infty}^{\infty} \frac{a}{a^2+\omega^2} df = \frac{1}{2}$$

You may need to use this integral:

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

Solution:

$$\mathcal{F}[u(t)] = \lim_{a\to 0} \mathcal{F}[e^{-at}u(t)] = \lim_{a\to 0} \frac{a}{a^2+\omega^2} + \lim_{a\to 0} \frac{-j\omega}{a^2+\omega^2}$$

$$= \lim_{a\to 0} \frac{a}{a^2+\omega^2} + \frac{1}{j\omega}$$

Consider the first term:

$$\lim_{a\to 0} \frac{a}{a^2+\omega^2} = \begin{cases} \infty & \omega = 0 \\ 0 & \omega \neq 0 \end{cases}$$

and its integral is 1/2 independent of a:

$$\int_{-\infty}^{\infty} \frac{a}{a^2+\omega^2} df = \frac{a}{2\pi} \int_{-\infty}^{\infty} \frac{1}{a^2+\omega^2} d\omega = \frac{1}{2\pi} \tan^{-1}\left(\frac{\omega}{a}\right) \Big|_{-\infty}^{\infty} = \frac{1}{2\pi}[\frac{\pi}{2} - (-\frac{\pi}{2})] = \frac{1}{2}$$

where we have used the integral formula:

$$\int \frac{dx}{a^2+x^2} = \frac{1}{a} \tan^{-1} \frac{x}{a}$$

The first term is therefore a delta function:

$$\lim_{a \to 0} \frac{a}{a^2+\omega^2} = \frac{1}{2}\delta(f)$$

and we have:

$$\mathcal{F}[u(t)] = \frac{1}{2}\delta(f) + \frac{1}{j2\pi f}$$

2. Consider the same RC circuit in Example 3.4 (Fig.3.13), 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 3.4 and the Kirchhoff's voltage law: $v_{in}(t) = v_C(t) + v_R(t)$:

$$v_R(t) = v_{in}(t) - v_C(t) = \delta(t) - \frac{1}{\tau} e^{-t/\tau} u(t)$$

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 3.4.
(b) Solve the differential equation for $y'(t) = f(t)u(t)$ when $x'(t) = u(t)$. Then find $h(t) = y(t) = \dot{y}'(t)$ 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 $H(\omega) = \mathcal{F}[h(t)]$.
Solution:
– Set up the differential equation for the system:
The voltage across R is $v_R(t) = i(t)R$ and the voltage across C is

$$v_C(t) = \frac{1}{C} \int i(t)dt = \frac{1}{C} \int \frac{v_R(t)}{R} dt = \frac{1}{\tau} \int v_R(t)dt$$

According to Kirchhoff's voltage law, we have

$$v_R(t) + v_C(t) = v_R(t) + \frac{1}{\tau} \int v_R(t)dt = v_{in}(t)$$

Taking derivative on both sides we get:

$$\dot{y}(t) + \frac{1}{\tau} y(t) = \dot{x}(t)$$

Here $x(t) = v_{in}(t)$ is the input while $y(t) = v_R(t)$ is the corresponding output.

– Find impulse response $y(t) = h(t)$ to an impulse input $x(t) = \delta(t)$:
In Example 3.4, when $x(t) = v_{in}(t) = \delta(t)$, the output is $v_C(t) = h(t) = e^{-t/\tau}/\tau$, i.e., $v_R(t) = v_{in}(t) - v_C(t) = \delta(t) - e^{-t/\tau}/\tau$, which is the impulse response of the system when the output is $y(t) = v_R(t)$.
Alternatively, we can also solve the system for $y(t) = h(t)$ when $x(t) = \delta(t)$. However, for convenience, we first obtain the response $y'(t)$ to an input $x'(t) = u(t)$:

$$\dot{y'}(t) + \frac{1}{\tau}y'(t) = \dot{x'}(t) = \dot{u}(t) = \delta(t)$$

Assume $y'(t) = f(t)u(t)$ and $\dot{y'}(t) = \dot{f}(t)u(t) + f(0)\delta(t)$. Substituting these into the equation we get:

$$\dot{f}(t)u(t) + f(0)\delta(t) + \frac{1}{\tau}f(t)u(t) = \delta(t)$$

Separating terms containing $u(t)$ and $\delta(t)$ we get:

$$\begin{cases} \dot{f}(t) + f(t)/\tau = 0 \\ f(0) = 1 \end{cases}$$

Solving this homogeneous equation we get $f(t) = e^{-t/\tau}$ and $y'(t) = f(t)u(t) = e^{-t/\tau}u(t)$. Taking time derivative on both input $x'(t)$ and output $y'(t)$ we get the impulse input $x(t) = \dot{x'}(t) = \dot{u}(t) = \delta(t)$ and the impulse response:

$$h(t) = y(t) = \dot{y'}(t) = -\frac{1}{\tau}e^{-t/\tau}u(t) + e^{-t/\tau}\delta(t) = \delta(t) - \frac{1}{\tau}e^{-t/\tau}u(t)$$

– Find the frequency response function $H(\omega)$:
When the input is a complex exponential $x(t) = e^{j\omega t}$, we have $\dot{x}(t) = j\omega e^{j\omega t}$, and the output also takes the form of a complex exponential $y(t) = H(\omega)e^{j\omega t}$ and $\dot{y}(t) = j\omega H(\omega)e^{j\omega t}$. Substituting these into the equation we get:

$$\dot{y}(t) + y(t) = [j\omega + \frac{1}{\tau}]H(\omega)e^{j\omega t} = j\omega e^{j\omega t}$$

Solving this we get the frequency response function:

$$H(\omega) = \frac{j\omega\tau}{j\omega\tau + 1}$$

– Now we verify that $H(\omega)$ is indeed the Fourier transform of the impulse response function:

$$\begin{aligned} \mathcal{F}[h(t)] &= \int_{-\infty}^{\infty} h(t)e^{-j\omega t}dt = \int_{-\infty}^{\infty} [\delta(t) - \frac{1}{\tau}e^{-t/\tau}u(t)]e^{-j\omega t}dt \\ &= 1 - \frac{1}{\tau}\int_0^{\infty} e^{-(j\omega+1/\tau)t}dt = 1 - \frac{1}{j\omega\tau+1} = \frac{j\omega}{j\omega\tau+1} \end{aligned}$$

4 Discrete-Time Fourier Transform

4.1 Discrete-Time Fourier Transform

4.1.1 Fourier Transform of Discrete Signals

To use the modern digital technology to process a time signal $x(t)$, an analog to digital (A to D or A/D) converter is needed to sample the continuous signal and convert it into a discrete signal composed of a set of time samples $x[m] = x(mt_0) = x(m/F)$ $(m = \cdots, -1, 0, 1, \cdots)$, where t_0 is the sampling period, the time interval between two consecutive samples, and $F = 1/t_0$ is the *sampling frequency*. Mathematically the sampled signal $x_s(t)$ can be represented as the product of the time signal and an impulse train, the sampling function (also called a comb function):

$$x_s(t) = x(t)\, comb(t) = x(t) \sum_{m=-\infty}^{\infty} \delta(t - mt_0) = \sum_{m=-\infty}^{\infty} x[m]\delta(t - mt_0) \tag{4.1}$$

where $x[m] = x(mt_0)$ is the mth sample, the signal $x(t)$ evaluated at $t = mt_0$. The Fourier transform of this sampled signal can be found as

$$X_s(f) = \int_{-\infty}^{\infty} x_s(t)e^{-j2\pi ft}dt = \int_{-\infty}^{\infty} [\sum_{m=-\infty}^{\infty} x[m]\delta(t - mt_0)]e^{-j2\pi ft}dt$$

$$= \sum_{m=-\infty}^{\infty} x[m] \int_{-\infty}^{\infty} \delta(t - mt_0)e^{-j2\pi ft}dt = \sum_{m=-\infty}^{\infty} x[m]e^{-j2m\pi ft_0} \tag{4.2}$$

This is the spectrum of the discrete signal $x[m]$ $(m = \cdots, -1, 0, 1, \cdots)$, which can also be expressed in terms of angular frequency $\omega = 2\pi f$:

$$X_s(\omega) = \sum_{m=-\infty}^{\infty} x[m]e^{-jm\omega t_0} \tag{4.3}$$

This spectrum is periodic with the sampling frequency $F = 1/t_0$ as the period:

$$X(f + F) = \sum_{m=-\infty}^{\infty} x[m]e^{-j2m\pi(f+F)t_0}dt = \sum_{m=-\infty}^{\infty} x[m]e^{-j2m\pi ft_0}e^{-j2m\pi Ft_0}dt = X(f) \tag{4.4}$$

where $e^{-j2m\pi Ft_0} = e^{-j2m\pi} = 1$ as $F = 1/t_0$. We can therefore express this periodic spectrum as $X_F(f)$, just as $x_T(t)$ representing a periodic time signal with

period T. This spectrum could also be expressed as $X_\Omega(\omega + \Omega)$ in terms of angular frequency $\omega = 2\pi f$ and period $\Omega = 2\pi F$.

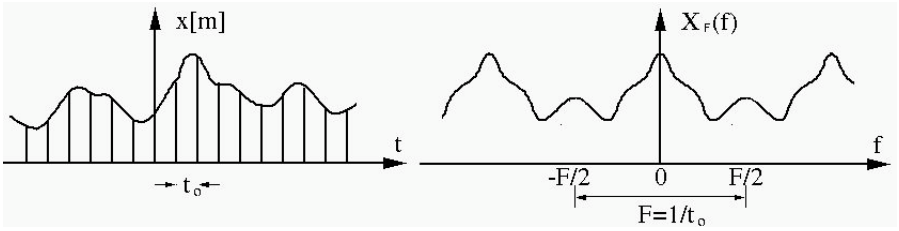

Figure 4.1 Fourier transform of discrete signals

To get the time samples of the discrete signal back from its spectrum $X_F(f)$, we multiply $e^{j2n\pi ft_0}/F$ on both sides of Eq. 4.2 and integrate over period F:

$$\frac{1}{F}\int_0^F X(f)e^{j2n\pi ft_0}df = \frac{1}{F}\sum_{m=-\infty}^{\infty} x[m]\int_0^F e^{-j2(m-n)\pi ft_0}df$$

$$= \sum_{m=-\infty}^{\infty} x[m]\delta[m-n] = x[n] \tag{4.5}$$

Here we have used Eq.1.27 (with different variables). This is the inverse Fourier transform. Equations 4.2 and 4.5 form a pair of the Fourier transform of a discrete signal, called the discrete-time Fourier transform (DTFT):

$$X_F(f) = \sum_{m=-\infty}^{\infty} x[m]e^{-j2m\pi f/F}$$

$$x[m] = \frac{1}{F}\int_0^F X_F(f)e^{j2m\pi f/F}df \tag{4.6}$$

which can also be expressed in terms of $\omega = 2\pi f$ as:

$$X_\Omega(\omega) = \sum_{m=-\infty}^{\infty} x[m]e^{-jm\omega t_0}$$

$$x[m] = \frac{1}{\Omega}\int_0^F X_\Omega(\omega)e^{jm\omega t_0}d\omega \tag{4.7}$$

The discrete-time Fourier transform can be considered as the representation of a signal vector $\boldsymbol{x} = [\cdots, x[m], \cdots]^T$ $(m = -\infty, \cdots, -1, 0, 1, \cdots, \infty)$ in an vector space, as a linear combination (an integral) of a set of orthonormal basis (uncountable) vectors $\boldsymbol{\phi}(f) = [\cdots, e^{j2\pi mf/F}, \cdots]^T/\sqrt{F}$ $(0 < f < F)$ that spans the space:

$$\boldsymbol{x} = \frac{1}{\sqrt{F}}\int_F X(f)\boldsymbol{\phi}(f)df \tag{4.8}$$

The component form of this integral is the inverse DTFT (with a different scaling factor):

$$x[m] = \frac{1}{F}\int_0^F X(f)e^{j2m\pi f/F}df = \frac{1}{\sqrt{F}}\int_0^F X(f)e^{j2m\pi t_0 f}df, \quad \text{(for all } m\text{)} \tag{4.9}$$

The coefficient function $X(f)$ can be found by multiplying both sides by $e^{-j2\pi mt_0 f'}/\sqrt{F}$ and taking a summation:

$$\begin{aligned}\frac{1}{\sqrt{F}}\sum_{m=-\infty}^{\infty} x[m]e^{j2\pi mt_0 f'} &= \int_0^F X(f)\frac{1}{F}\sum_{m=-\infty}^{\infty} e^{j2m\pi t_0(f-f')}df \\ &= \int_0^F X(f)\sum_{k=-\infty}^{\infty}\delta(f-f'-kF)df = X(f')\end{aligned} \tag{4.10}$$

Here we have used the result in Eq.1.28. This is the forward DTFT (with a different scaling factor), which also indicates that the coefficient function $X(f)$ has a period F.

We further consider the inner product of two signals $\boldsymbol{x}$ and $\boldsymbol{y}$ before the transform and that of their spectra after the transform:

$$\begin{aligned}<\boldsymbol{x},\boldsymbol{y}> &= \sum_{m=-\infty}^{\infty} x[m]\overline{y}[m] = \sum_{m=-\infty}^{\infty} x[m]\overline{y}[m] \\ &= \sum_{m=-\infty}^{\infty}\frac{1}{\sqrt{F}}\int_F X(f)e^{j2\pi mt_0 f}df\frac{1}{\sqrt{F}}\int_F \overline{Y}(f')e^{-j2\pi mt_0 f'}df' \\ &= \int_F\int_F X(f)\overline{Y}(f')\frac{1}{F}\sum_{m=-\infty}^{\infty} e^{-j2\pi mt_0(f-f')}df\,df' \\ &= \int_F\int_F X(f)\overline{Y}(f')\delta(f-f'-kF)df\,df' = \int_F X(f)\overline{Y}(f)df \\ &= <X(f),Y(f)>\end{aligned} \tag{4.11}$$

As inner product is preserved by the discrete-time Fourier transform, it is a unitary transformation. In particular, if we let $\boldsymbol{x} = \boldsymbol{y}$, we get Parseval's identity:

$$\sum_{m=-\infty}^{\infty}|x[m]|^2 = \int_F |X(f)|^2 df \tag{4.12}$$

indicating that the energy or information contained in the signal is preserved by the DTFT.

Comparing this pair of equations to the Fourier series expansion of a periodic signal $x_T(t)$ in Eq. 3.5:

$$\begin{aligned}X[k] &= \frac{1}{T}\int_T x_T(t)e^{-jk\omega_0 t}dt = \frac{1}{T}\int_T x_T(t)e^{-j2k\pi t/T}dt \\ x_T(t) &= \sum_{k=-\infty}^{\infty} X[k]e^{jk\omega_0 t} = \sum_{k=-\infty}^{\infty} X[k]e^{j2k\pi t/T}\end{aligned}$$

we see a perfect duality between time and frequency domains:

- When a time signal is periodic with period T, its spectrum is discrete with a frequency interval $f_0 = 1/T$ between two consecutive frequency components (the fundamental and its harmonics).
- When a time signal is discrete with time interval t_0 between two consecutive samples, its spectrum is periodic with period $F = 1/t_0$.

This duality between time and frequency should not be a surprise, due to the symmetry of the definition of the forward and inverse Fourier transforms in Eq. 3.53.

Once a continuous signal is sampled to become a set 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 unit $t_0 = 1$ and the sampling frequency is also unit $F = 1/t_0 = 1$, and the Fourier transform pair of the discrete signal can be simply expressed as:

$$X(f) = \sum_{m=-\infty}^{\infty} x[m]e^{-j2m\pi f}, \quad \text{or} \quad X(\omega) = \sum_{m=-\infty}^{\infty} x[m]e^{-jm\omega}$$
$$x[m] = \int_0^1 X(f)e^{j2m\pi f}df = \frac{1}{2\pi}\int_0^{2\pi} X(f)e^{jm\omega}d\omega \tag{4.13}$$

In some literatures, the Fourier spectrum of a discrete signal is also denoted by $X(e^{j\omega})$, because it takes this form when considered as a special case of the z-transform, to be discussed in the next chapter. However, we note that all these different forms of the spectrum are just some notational variations all representing essentially the same fact: the spectrum is simply a function of frequency f or angular frequency $\omega = 2\pi f$. No confusion should be caused given the specific context of the discussion. In the following, we will use $X(f)$, $X(\omega)$ as well as $X(e^{j\omega})$ interchangeably for the spectrum of a discrete signal, whichever is more convenient and suitable in each specific case.

Example 4.1: Consider the Fourier transform of a few special signals:

- The Kronecker delta or unit impulse function:

$$\mathcal{F}[\delta[m]] = \sum_{m=-\infty}^{\infty} \delta[m]e^{-j2m\pi f} = e^{-j2\pi 0f} = 1 \tag{4.14}$$

- Constant function (an impulse train in time domain):

$$\mathcal{F}[x[m] = \mathcal{F}[1] = \sum_{m=-\infty}^{\infty} e^{j2m\pi f} = \sum_{n=-\infty}^{\infty} \delta(f-n) \tag{4.15}$$

The last equal sign is due to Eq.1.28. This is an impulse train in frequency domain.

- The sign function:

$$sgn[m] = \begin{cases} -1 & m < 0 \\ 0 & m = 0 \\ 1 & m > 0 \end{cases}$$

$$\mathcal{F}[sgn[m]] = -\sum_{m=-\infty}^{-1} e^{-j2m\pi f} + \sum_{m=1}^{\infty} e^{-j2m\pi f} = -\sum_{m=1}^{\infty} e^{j2m\pi f} + \sum_{m=1}^{\infty} e^{-j2m\pi f}$$

Consider the first summation as the following limit when a real parameter $|a| < 1$ approaches zero:

$$-\lim_{a\to 1}\sum_{m=1}^{\infty}(a\,e^{j2\pi f})^m = 1 - \lim_{a\to 1}\sum_{m=0}^{\infty}(a\,e^{j2\pi f})^m = \lim_{a\to 1}\frac{-ae^{j2\pi f}}{1-ae^{j2\pi f}} = \frac{-e^{j2\pi f}}{1-e^{j2\pi f}}$$

Similarly the second integral can be found to be:

$$\sum_{m=1}^{\infty} e^{-j2m\pi f} = \frac{e^{-j2\pi f}}{1-e^{-j2\pi f}}$$

Now we get:

$$\mathcal{F}[sgn[m]] = \frac{-e^{j2\pi f}}{1-e^{j2\pi f}} + \frac{e^{-j2\pi f}}{1-e^{-j2\pi f}} = \frac{1+e^{-j2\pi f}}{1-e^{-j2\pi f}} = \frac{j\ \sin(2\pi f)}{\cos(2\pi f)-1} \quad (4.16)$$

- Unit step function:

$$u[m] = \frac{1}{2}\left[1 + \delta[m] + sgn[m]\right] = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$$

Note that $u[0] = 1$, unlike in the continuous case $u(0) = 1/2$, it can be constructed as the sum of three functions:

$$u[m] = \frac{1}{2}\left[1 + \delta[m] + sgn[m]\right] \quad (4.17)$$

As the Fourier transform is obviously linear, we have:

$$\mathcal{F}[u[m]] = \frac{1}{2}\left[\mathcal{F}[1] + \mathcal{F}[\delta[m]] + \mathcal{F}[sgn[m]]\ \right] = \frac{1}{2}\left[\sum_{n=-\infty}^{\infty}\delta(f-n) + 1 + \frac{1+e^{-j2\pi f}}{1-e^{-j2\pi f}}\right]$$

$$= \frac{1}{1-e^{-j2\pi f}} + \frac{1}{2}\sum_{n=-\infty}^{\infty}\delta(f-n) \quad (4.18)$$

4.1.2 The Properties

As a special case of the generic form of the Fourier transform, the discrete-time Fourier transform shares many of the properties discussed considered in previous chapter, but in different forms. Here we assume $X(f) = \mathcal{F}[x[m]]$ and $Y(f) = \mathcal{F}[y[m]]$. Proofs of many of these properties are not given as they can

be easily derived from the definition. (Many of these can be left for homework problems.)

- **Linearity**

$$\mathcal{F}[ax[m] + by[m]] = aX(f) + bY(f) \tag{4.19}$$

- **Periodicity**

$$X(f + k) = X(f) \tag{4.20}$$

where k is an integer.
Proof:

$$X(f+k) = \sum_{m=-\infty}^{\infty} x[m]e^{-j2m\pi(f+k)} = \sum_{m=-\infty}^{\infty} x[m]e^{-j2m\pi f}e^{-j2mk\pi} = X(f)$$

as $e^{-j2mk\pi} = 1$.

- **Complex conjugate**

$$\mathcal{F}[\overline{x}[m]] = \overline{X}(-f) \tag{4.21}$$

- **Parseval's identity**

$$\sum_{m=-\infty}^{\infty} |x[m]|^2 = \int_0^1 |X(f)|^2 df \tag{4.22}$$

- **Time reversal**

$$\mathcal{F}[x[-m]] = X(-f) \tag{4.23}$$

If $\overline{x}[m] = x[m]$ is real, then

$$\mathcal{F}[x[-m]] = X(-f) = \overline{X}(f) \tag{4.24}$$

- **Time and frequency shifting**

$$\mathcal{F}[x[m \pm m_0]] = e^{\pm j2m_0 f}X(f) \tag{4.25}$$

$$\mathcal{F}[e^{\mp j2m\pi f_0}x[m] = X(f \pm f_0) \tag{4.26}$$

- **Correlation**
The cross-correlation between two functions $x[m]$ and $y[m]$ is defined as

$$r_{xy}[m] = x[m] \star y[m] = \sum_n x[n]\overline{y}[n-m] \tag{4.27}$$

This property states:

$$\mathcal{F}[x[m] \star y[m]] = X(f)\overline{Y}(f) \tag{4.28}$$

Proof:

$$\begin{aligned}\mathcal{F}[x[m] \star y[m]] &= \sum_m [\sum_n x[n]\overline{y}[n-m]]e^{-j2m\pi f} = \sum_n x[n][\sum_m \overline{y}[n-m]e^{-j2m\pi f}] \\ &= \sum_n x[n][\sum_{m'} \overline{y}[m']e^{-j2(n-m')\pi f}] = \sum_n x[n]e^{-j2n\pi f}\sum_{m'} \overline{y}[m']e^{j2m'\pi f} \\ &= X(f)\overline{Y}(f) = S_{xy}(f) \end{aligned} \tag{4.29}$$

where we have assumed $m' = n - m$, and $S_{xy}(f) = X(f)\overline{Y}(f)$ is the cross power density spectrum of the two signals. In particular, if both signals $\overline{x}[m] = x[m]$ and $\overline{y}[m] = y[m]$ are real, we have

$$\mathcal{F}[x[m] \star y[m]] = X(f)Y(-f) \tag{4.30}$$

- **Time and frequency convolution**

$$\mathcal{F}[x[m] * y[m]] = X(f)Y(f), \qquad \mathcal{F}[x[m]y[m]] = X(f) * Y(f) \tag{4.31}$$

Proof:

$$\begin{aligned}\mathcal{F}[x[m] * y[m]] &= \sum_m \sum_n x[n]y[m-n]e^{-j2m\pi f} = \sum_n x[n]\sum_m y[m-n]e^{-j2m\pi f} \\ &= \sum_n x[n]\sum_{m'} y[m']e^{-j2(m'+n)\pi f} = \sum_n x[n]\sum_{m'} y[m']e^{-j2(m')\pi f}e^{-j2(n)\pi f} \\ &= X(f)Y(f)\end{aligned}$$

where we have assumed $m' = m - n$. Also, the second equation can be derived:

$$\begin{aligned}\mathcal{F}[x[m]y[m]] &= \sum_m \left[\int_0^1 X(f')e^{j2m\pi f'}df'\right] y[m]e^{-j2m\pi f} = \int_0^1 X(f')\left[\sum_m y[m]e^{-j2m\pi(f-f')}\right]df' \\ &= \int_0^1 X(f')Y(f-f')df' = X(f) * Y(f)\end{aligned}$$

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, the first differencing of a discrete signal is simply defined as $x[m] - x[m-1]$. Based on the time shifting property, we have:

$$\mathcal{F}\left[x[m] - x[m-1]\right] = (1 - e^{-j2\pi f})X(f) \tag{4.32}$$

- **Time accumulation**
  Similar to the integral a continuous signal, the accumulation of a discrete signal is a summation of all its samples $x[m]$ from $m = -\infty$ up to $m = n$, and its Fourier transform is:

$$\mathcal{F}\left[\sum_{n=-\infty}^{m} x[n]\right] = \frac{1}{1 - e^{-j2\pi f}}X(f) + \frac{X(0)}{2}\sum_{n=-\infty}^{\infty}\delta(f-n) \tag{4.33}$$

Proof: The accumulation can be expressed as a convolution:

$$\sum_{n=-\infty}^{m} x[n] = \sum_{n=-\infty}^{\infty} u[m-n]x[n] = u[m] * x[m]$$

whose Fourier transform can be easily found according to the time convolution property:

$$\mathcal{F}\left[\sum_{n=-\infty}^{m} x[n]\right] = \mathcal{F}\left[u[m] * x[m]\right] = \mathcal{F}[u[m]]\,\mathcal{F}[x[m]]$$

$$= \left[\frac{1}{1-e^{-j2\pi f}} + \frac{1}{2}\sum_{n=-\infty}^{\infty}\delta(f-n)\right]X(f) = \frac{1}{1-e^{-j2\pi f}}X(f) + \frac{X(0)}{2}\sum_{n=-\infty}^{\infty}\delta(f-n)$$

Here we have used the fact that $X(f)$ is periodic and $X(k) = X(0)$. Comparing Eqs.4.32 and 4.33, 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.3.118 and 3.122). The second term of the right-hand side in Eq.4.33 represents the DC component in the signal $x[m]$, which is not needed in Eq.4.32 as differencing operation is insensitive to DC component.

- **Modulation**
  Here modulation means every odd sample of the signal $x[m]$ is negated.

$$\mathcal{F}[(-1)^m x[m]] = X\left(f + \frac{1}{2}\right) \tag{4.34}$$

Proof:

$$\sum_{n=-\infty}^{\infty} x[n](-1)^m e^{-j2m\pi f} = \sum_{n=-\infty}^{\infty} x[n]e^{-jm\pi}e^{-j2m\pi f}$$

$$= \sum_{n=-\infty}^{\infty} x[n]e^{-jm2\pi(f+1/2)} = X\left(f + \frac{1}{2}\right) \tag{4.35}$$

- **Up-sampling (time expansion)**

$$\mathcal{F}[x^{(k)}[m]] = X(kf) \tag{4.36}$$

Here $x^{(k)}[m]$ is defined as:

$$x^{(k)}[m] = \begin{cases} x[m/k] & \text{if } m \text{ is a multiple of } k \\ 0 & \text{else} \end{cases} \tag{4.37}$$

i.e. $x^{(k)}[m]$ is obtained by inserting $k-1$ zeros between every two consecutive samples of $x[m]$. Correspondingly its spectrum $X(kf)$ in frequency domain is compressed k times with the same magnitude. Note that this up-sampling is quite different from the time scaling of a continuous signal in Eq.3.102 with $a = 1/k$:

$$\mathcal{F}[x(t/k)] = kX(kf)$$

in which case the signal $x(t)$ is expanded k times (without any gap), and consequently the magnitude of its Fourier spectrum $X(f)$ is scaled up also k times.

Proof:

$$\mathcal{F}[x^{(k)}[m]] = \sum_{m=-\infty}^{\infty} x[m/k]e^{-j2m\pi f} = \sum_{n=-\infty}^{\infty} x[n]e^{-j2kn\pi f/k} = X(kf) \quad (4.38)$$

Note that the change of the summation index from m to $n = m/k$ has no effect as the terms skipped are all zeros.

- **Down-sampling**

$$\mathcal{F}[x_{(2)}[m]] = \mathcal{F}[x[2m]] = \frac{1}{2}\left[X\left(\frac{f}{2}\right) + X\left(\frac{f+1}{2}\right)\right] \quad (4.39)$$

Here the down-sampled version $x_{(2)}[m]$ of a signal $x[m]$ is composed of all the even terms of the signal with all odd terms dropped, i.e., $x_{(2)}[m] = x[2m]$. Down-sampling of a discrete signal corresponds to the compression of a continuous signal (Eq.3.102 with $a = 2$):

$$\mathcal{F}[x(2t)] = \frac{1}{2}X\left(\frac{f}{2}\right)$$

Proof:

$$\begin{aligned}
\mathcal{F}[x_{(2)}[m]] &= \sum_{m=-\infty}^{\infty} x[2m]e^{-j2\pi mf} = \sum_{n=\cdots,-2,0,2,\cdots} x[n]e^{-j\pi nf} \\
&= \frac{1}{2}[\sum_{n=-\infty}^{\infty} x[n]e^{-j\pi nf} + \sum_{n=-\infty}^{\infty} (-1)^n x[n]e^{-j\pi nf}] \\
&= \frac{1}{2}[\sum_{n=-\infty}^{\infty} x[n]e^{-j\pi nf} + \sum_{n=-\infty}^{\infty} x[n]e^{-j\pi n(f+1)}] \\
&= \frac{1}{2}\left[X\left(\frac{f}{2}\right) + X\left(\frac{f+1}{2}\right)\right] \quad (4.40)
\end{aligned}$$

Conceptually, the down-sampling of a given discrete signal $x[m]$ can be realized in the following three steps:

- Obtain its modulation $x[m](-1)^m = x[m]e^{jm\pi}$. Due to the frequency shift property, this corresponds to the spectrum shifted by 1/2:

$$\mathcal{F}\left[(-1)^m x[m]\right] = \mathcal{F}\left[e^{jm\pi}x[m]\right] = X(f + 1/2)$$

- Obtain the average of the signal and its modulation in both time and frequency domains:

$$\mathcal{F}\left[\frac{1}{2}[x[m] + x[m](-1)^m]\right] = \frac{1}{2}\left[X(f) + X\left(f + \frac{1}{2}\right)\right]$$

– Remove odd samples of the average to get $x_{(2)}[m]$. In frequency domain, this corresponds to replacing f by $f/2$:

$$\mathcal{F}\left[x_{(2)}[m]\right] = \frac{1}{2}\left[X\left(\frac{f}{2}\right) + X\left(\frac{f+1}{2}\right)\right]$$

Example 4.2: Here we consider the up-sampling, modulation and down-sampling of a discrete signal of square wave $x[m]$ 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 $x[m]$.
- The up-sampled version $x^{(2)}[m]$ 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 $x^{(3)}[m]$ 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 $m = -2$ and $m = 2$ with a DC component 1 (due to the time sample at $m = 0$.
- 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.

Example 4.3: Here we consider the convolution of two finite discrete signals $x[m]$ of length M and $h[m]$ of size N, i.e., $x[m]$ is zero outside the range $0 \le m \le M-1$; and $h[m]$ is zero outside the range $0 \le m \le N-1$. Their convolution is:

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

Note that the range for $h[m-n]$ is $0 \le m-n < N$, i.e., $n \le m < N+n$. But as $0 \le n \le M-1$, we get the range $0 \le n \le m \le N+n-1 \le N+M-1$ for $y[m]$ outside which $y[m] = 0$. Specifically, let:

$$\boldsymbol{x} = [1\ 2\ 3\ 4\ 5\ 6\ 7\ 8]^T, \qquad \boldsymbol{h} = [1\ 2\ 3]^T$$

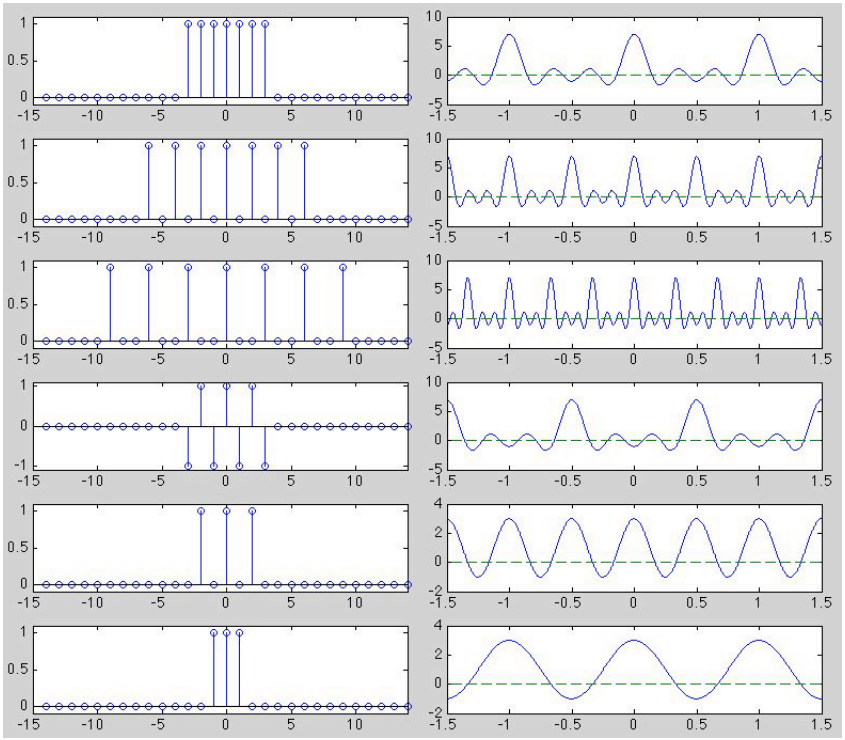

Figure 4.2 Modulation, up and down-sampling
The square wave and its modulation, up and down sampling versions on the left, and their spectra (showing three periods) on the right.

Here $M = 8$, $N = 3$, and the length of the result of this convolution is $M + N - 1 = 8 + 3 - 1 = 10$, any $y[m]$ outside the range of $0 < m < 9$ is zero. This convolution can be illustrated below:

n	$\cdots$	-1	0	1	2	3	4	5	6	7	8	9	10	$\cdots$
$x[n]$	$\cdots$	0	1	2	3	4	5	6	7	8	0	0	0	$\cdots$
$h[0-n]$	$\cdots$	2	1											$\cdots$
$h[1-n]$	$\cdots$	3	2	1										$\cdots$
$h[2-n]$	$\cdots$		3	2	1									$\cdots$
$h[3-n]$	$\cdots$			3	2	1								$\cdots$
$h[4-n]$	$\cdots$				3	2	1							$\cdots$
$h[5-n]$	$\cdots$					3	2	1						$\cdots$
$h[6-n]$	$\cdots$						3	2	1					$\cdots$
$h[7-n]$	$\cdots$							3	2	1				$\cdots$
$h[8-n]$	$\cdots$								3	2	1			$\cdots$
$h[9-n]$	$\cdots$									3	2	1		$\cdots$
$h[10-n]$	$\cdots$										3	2	1	$\cdots$
$y[m]$	$\cdots$	0	1	4	10	16	22	28	34	40	37	24	0	$\cdots$

4.1.3 Discrete Time Fourier Transform of Typical Functions

- **Constant:** If we let $x[m] = 1$ in Eq.4.1, we get an impulse train in time domain:

$$x_s(t) = comb(t) = \sum_{m=-\infty}^{\infty} \delta(t - mt_0) = \sum_{m=-\infty}^{\infty} \delta(t - mt_0) \tag{4.41}$$

whose discrete-time Fourier transform is also an impulse train in frequency domain:

$$\mathcal{F}[x[m] = \mathcal{F}[1] = \sum_{m=-\infty}^{\infty} e^{j2m\pi f} = \sum_{n=-\infty}^{\infty} \delta(f - n) \tag{4.42}$$

The last equal sign is due to Eq.1.28.

- **Complex exponential**
Applying frequency shift property to the result above we get:

$$\mathcal{F}[e^{j2m\pi f_0}] = \sum_{n=-\infty}^{\infty} \delta(f - f_0 - n) \tag{4.43}$$

- **Sinusoids**

$$\begin{aligned}\mathcal{F}[\cos(2m\pi f_0)] &= \frac{1}{2}[\mathcal{F}[e^{j2m\pi f_0}] + \mathcal{F}[e^{-j2m\pi f_0}]] \\ &= \frac{1}{2}\left[\sum_{n=-\infty}^{\infty} \delta(f - f_0 - n) + \sum_{n=-\infty}^{\infty} \delta(f + f_0 - n)\right]\end{aligned} \tag{4.44}$$

Similarly we have:

$$\begin{aligned}\mathcal{F}[\sin(2m\pi f_0)] &= \frac{1}{2j}[\mathcal{F}[e^{j2m\pi f_0}] - \mathcal{F}[e^{-j2m\pi f_0}]] \\ &= \frac{1}{2j}\left[\sum_{n=-\infty}^{\infty} \delta(f - f_0 - n) - \sum_{n=-\infty}^{\infty} \delta(f + f_0 - n)\right]\end{aligned} \tag{4.45}$$

- **Kronecker delta**

$$\mathcal{F}\left[\delta[m]\right] = \sum_{m=-\infty}^{\infty} \delta[m] e^{j2m\pi f} = e^{j0} = 1 \tag{4.46}$$

- **Sign function**

$$\mathcal{F}\left[sgn[m]\right] = \frac{-e^{j2\pi f}}{1 - e^{j2\pi f}} + \frac{e^{-j2\pi f}}{1 - e^{-j2\pi f}} = \frac{1 + e^{-j2\pi f}}{1 - e^{-j2\pi f}}$$

This is given in Eq.4.16.

- **Unit step function**

$$\mathcal{F}[u[m]] = \frac{1}{1-e^{-j2\pi f}} + \frac{1}{2}\sum_{n=-\infty}^{\infty}\delta(f-n)$$

This is given in Eq.4.18.

- **Exponential function**
First consider a right-sided exponential function:

$$x[m] = a^m u[m], \qquad (|a|<1)$$

$$\mathcal{F}[a^m u[m]] = \sum_{m=0}^{\infty}(ae^{-j2\pi f})^m = \frac{1}{1-ae^{-j2\pi f}} \tag{4.47}$$

Next consider the two-sided version:

$$x[m] = a^{|m|} = a^m u[m] + a^{-m}u[-m-1], \qquad (|a|<1)$$

The transform of the first term is the same as before, while the transform of the second term is:

$$\begin{aligned}\mathcal{F}\left[a^{-m}u[-m-1]\right] &= \sum_{m=-\infty}^{-1} a^{-m}e^{-j2m\pi f} = \sum_{m=0}^{\infty}(ae^{j2\pi f})^m - 1\\ &= \frac{ae^{j2\pi f}}{1-ae^{j2\pi f}}\end{aligned} \tag{4.48}$$

The over all transform is:

$$\mathcal{F}\left[a^{|m|}\right] = \frac{1}{1-ae^{-j2\pi f}} + \frac{ae^{j2\pi f}}{1-ae^{j2\pi f}} = \frac{1-a^2}{1+a^2-2a\cos(2\pi f)} \tag{4.49}$$

- **Square wave**

$$x[m] = \begin{cases} 1 & |m| \le N \\ 0 & |m| > N \end{cases}$$

The Fourier transform of this square wave of width $2N+1$ can be found to be:

$$\begin{aligned}\mathcal{F}[x[m]] &= \sum_{m=-N}^{N} e^{-jm\omega} = \sum_{m=-N}^{0} e^{-jm\omega} + \sum_{m=0}^{N} e^{-jm\omega} - 1\\ &= \frac{1-e^{j(N+1)\omega}}{1-e^{j\omega}} + \frac{1-e^{-j(N+1)\omega}}{1-e^{-j\omega}} - 1 = \frac{e^{j(N+1)\omega}-e^{-jN\omega}}{e^{j\omega}-1}\,\frac{e^{-j\omega/2}}{e^{-j\omega/2}}\\ &= \frac{e^{j(2N+1)\omega/2}-e^{-j(2N+1)\omega/2}}{e^{j\omega/2}-e^{-j\omega/2}} = \frac{sin((2N+1)\omega/2)}{\sin(\omega/2)}\end{aligned} \tag{4.50}$$

- **Triangle wave**

$$x[m] = \begin{cases} 1-|m|/N & |m| \le N \\ 0 & |m| > N \end{cases}$$

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:

$$\mathcal{F}\left[x[m]\right] = \frac{1}{N}\left[\frac{sin(N\omega/2)}{\sin(\omega/2)}\right]^2 \tag{4.51}$$

Fig.4.3 shows a set of typical discrete signals and their discrete-time Fourier transforms.

4.1.4 The Sampling Theorem

An important issue in sampling is the determination of the sampling frequency. On the one hand, We want 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 to storage and transmission. On the other hand, we also want to avoid losing information contained in the signal which may happen if the sampling frequency is too low.

According to Parseval's identity, a continuous signal $x(t)$ can be transformed to its spectrum $X(f) = \mathcal{F}[x(t)]$ and vice versa without any energy loss, and the original signal $x(t)$ can be perfectly recovered from its spectrum $X(f)$ by the inverse transform $x(t) = \mathcal{F}^{-1}[X(f)]$. However, if $x(t)$ is sampled, it is represented by its samples $x[m]$ ($m = 0, \pm1, \pm2, \cdots$), or the spectrum $X_F(f)$ of these discrete samples, can $x(t)$ still be recovered?

To answer this question, consider how the spectrum $X_F(f)$ of the samples $x[m]$ is related to the spectrum $X(f)$ of the continuous signal $x(t)$. As shown in Eq. 4.1), the sampled signal $x_s(t)$ can be obtained as the product of the signal and an impulse train, the comb function:

$$x_s(t) = x(t)\, comb(t) = x(t) \sum_{m=-\infty}^{\infty} \delta(t - mt_0) \tag{4.52}$$

Due to the convolution theorem, this multiplication of two functions in time domain corresponds to the convolution of their spectra in frequency domain:

$$X_F(f) = X(F) * Comb(f) \tag{4.53}$$

where $Comb(f)$ is the spectrum of the comb function given in Eg. 3.2.5:

$$Comb(f) = \frac{1}{T} \sum_{k=-\infty}^{\infty} \delta(f - k/T) \tag{4.54}$$

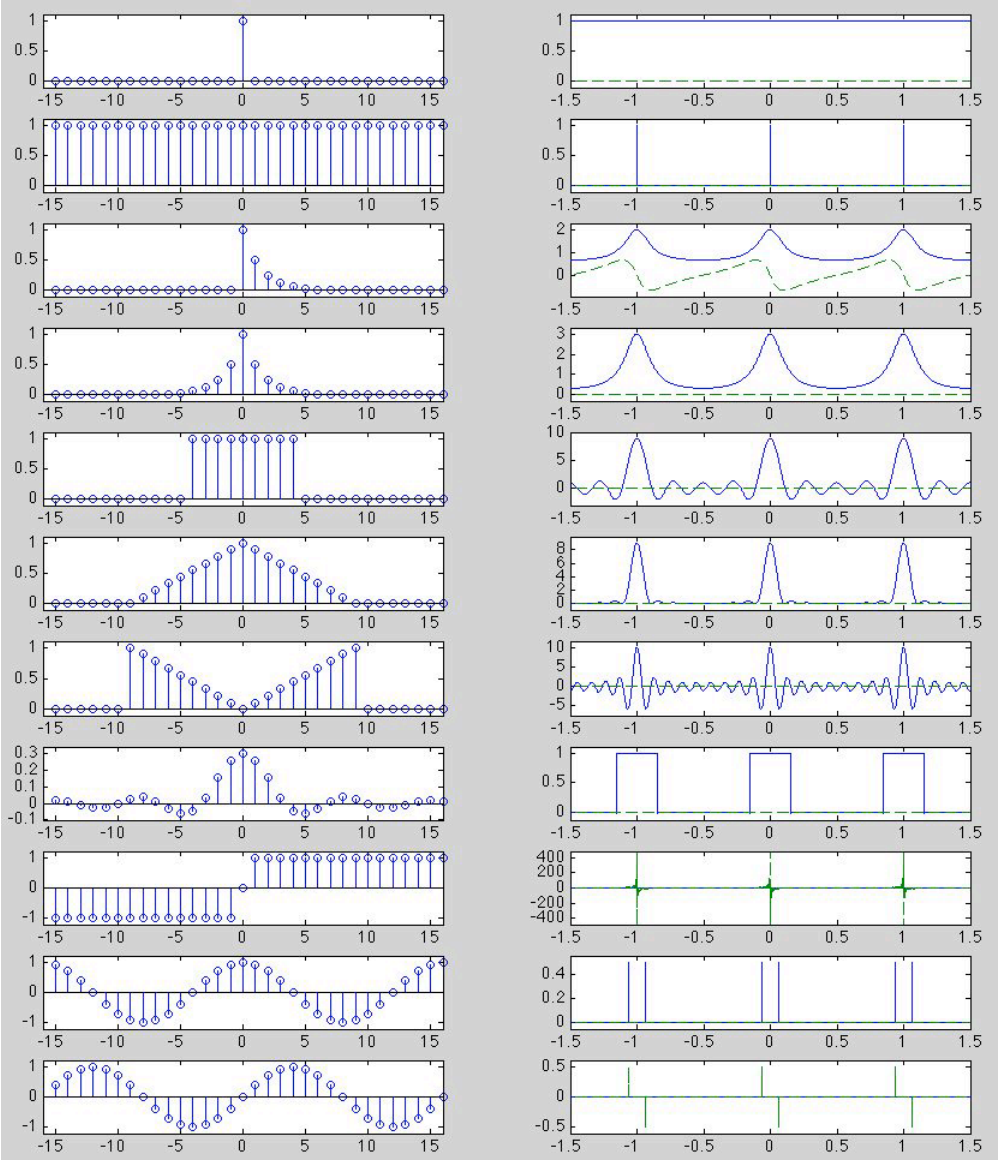

Figure 4.3 Examples of discrete-time Fourier transforms
A set of discrete signals are shown on the left and their DTFT spectra (showing three periods) are shown on the right (real and imaginary parts are shown in solid and dashed lines, respectively).

Substituting this in the equation for $X_F(f)$, we get:

$$X_F(f) = X(f) * \frac{1}{t_0}\sum_{k=-\infty}^{\infty}\delta(f - k/t_0) = \int_{-\infty}^{\infty} X(f-f')\frac{1}{t_0}\sum_{k=-\infty}^{\infty}\delta(f' - k/t_0)df'$$

$$= \frac{1}{t_0}\sum_{k=-\infty}^{\infty} X(f - k/t_0) = F\sum_{k=-\infty}^{\infty} X(f - kF) \tag{4.55}$$

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 recovered.

Consider the following two cases, also illustrated in Fig.4.4, where the maximum non-zero frequency component in $X(f)$ is assumed to be f_{max}.

- 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 recovered by a filtering process:

$$X(f) = H_{lp}X_F(f) \tag{4.56}$$

  where $H_{lp}(f)$ is an ideal low-pass filter defined as

$$H_{lp}(f) = \begin{cases} 1/F & |f| < f_c = F/2 \\ 0 & \text{otherwise} \end{cases} \tag{4.57}$$

  This filter scales $X_F(f)$ by $1/F = t_0$ but suppresses any frequency higher than the cut-off frequency $f_c = F/2$ to zero.
- If $F/2 < f_{max}$, *aliasing* occurs as the replicas in $X_F(f)$ are no longer separable, and it is no longer possible 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).

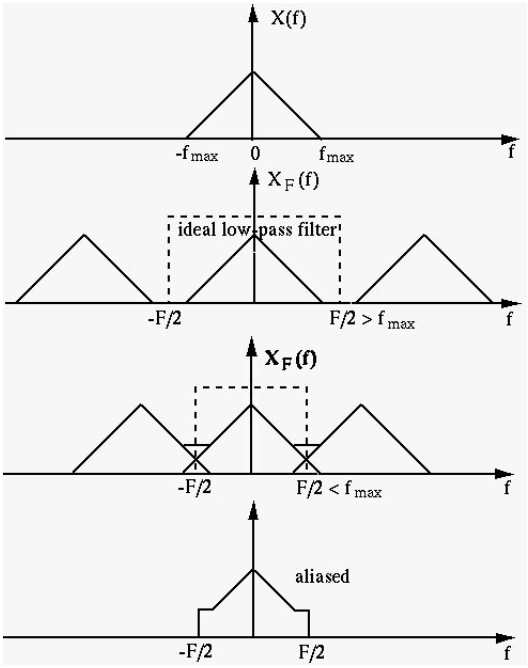

Figure 4.4 Reconstruction of time signal in frequency domain

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:*

$$f_{max} < f_{Nyquist} = F/2 \tag{4.58}$$

where f_{max} is the highest frequency contained in the signal. This is also 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 $F > 2f_{max}$, otherwise aliasing occurs and the original signal can not be perfectly recovered. In practice, however, it is often the case that the signal to be sampled contains frequency components higher than the Nyquist frequency. In such cases, one can still avoid aliasing by anti-aliasing low-pass filtering to remove all frequencies higher than the Nyquist frequency before sampling.

To fully understand the sampling theorem we consider the following two 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 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 ($F = 24$ frames per second), as illustrated in Fig.4.5.

We consider the following five cases where the signal frequency f_0 takes a set of different values. The sampling process can be represented in time domain as well as in frequency domain by the time samples of the signal:

$$x[m] = x(t)\big|_{t=mt_0} = x(mt_0) = x(m/F) = x(m/4) \tag{4.59}$$

- $f_0 = 1 < F/2 = 2$ Hz:

$$x[m] = x(m/4) = \frac{1}{2j}[e^{j2m\pi/4} - e^{-j2m\pi/4}] = \sin(2m\pi/4) \tag{4.60}$$

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 $f = \pm 1 \pm 4k$ for all integer k. In the 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:

$$x[m] = x(m/4) = \frac{1}{2j}[e^{j2m\pi 2/4} - e^{-j2m\pi 2/4}] = \frac{1}{2j}[e^{jm\pi} - e^{-jm\pi}] = 0 \quad (4.61)$$

The signal is sampled two times per period and in this case both samples happen to be zero, same as samples from a all zero signal $x(t) = 0$. In the 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 moving, as shown in the second row of Fig.4.5.

- $f_0 = 3 > F/2 = 2$ Hz:

$$\begin{aligned} x[m] = x(m/4) &= \frac{1}{2j}[e^{j2m\pi 3/4} - e^{-j2m\pi 3/4}] = \frac{1}{2j}[e^{j2m\pi(4-1)/4} - e^{-j2m\pi(4-1)/4}] \\ &= \frac{1}{2j}[e^{-j2m\pi/4} - e^{j2m\pi 1/4}] = -\sin(2m\pi/4) \end{aligned} \quad (4.62)$$

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:

$$x[m] = x(m/4) = \frac{1}{2j}[e^{j2m\pi 4/4} - e^{-j2m\pi 4/4}] = \frac{1}{2j}[e^{j2m\pi} - e^{-j2m\pi}] = 0 \quad (4.63)$$

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,

$$\begin{aligned} x[m] = x(m/4) &= \frac{1}{2j}[e^{j2m\pi 5/4} - e^{-j2m\pi 5/4}] = \frac{1}{2j}[e^{j2m\pi(4+1)/4} - e^{-j2m\pi(4+1)/4}] \\ &= \frac{1}{2j}[e^{j2m\pi/4} - e^{-j2m\pi/4}] = \sin(2m\pi/4) \end{aligned} \quad (4.64)$$

The samples are identical to those 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 general, even if we know in theory the object could have rotated an angle of $\phi \pm 2k\pi$ per flash, the perceived frequency 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.

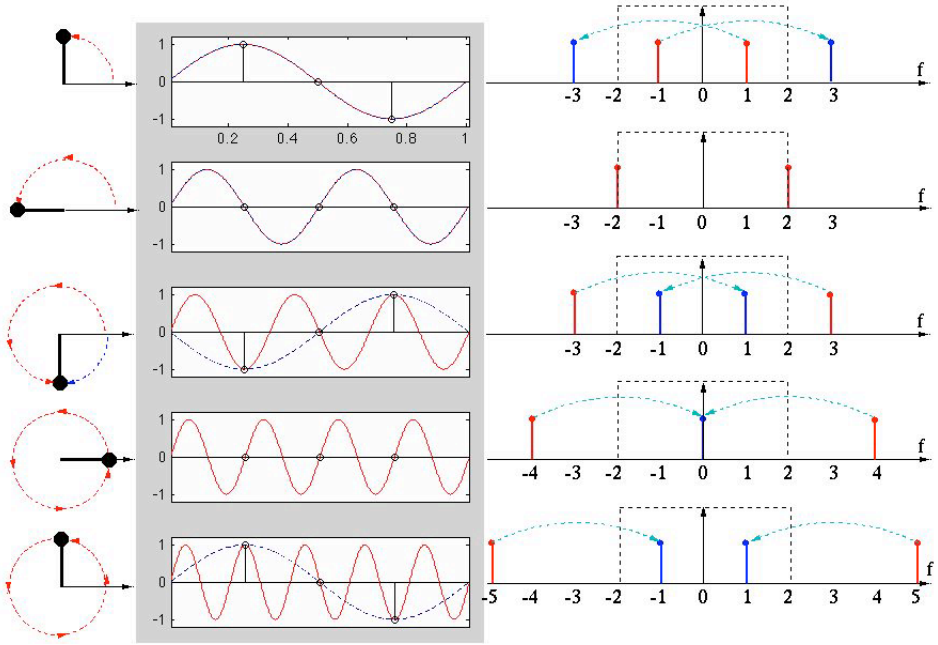

Figure 4.5 Aliasing and folding in time and frequency domains
Model of rotating object illuminated by a strobe light (left, only the first flash is shown), sampling of the vertical displacement (middle), and the aliased frequency (perceived rotation) (right)

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 with a phase shift $x(t) = \sin(2\pi f_0 t + \phi)$. When it is sampled at rate $F = 2f_0$,

the values of its samples depend on the phase ϕ:

$$x[m] = x(m/F) = \sin(2m\pi f_0/F + \phi) = \sin(m\pi + \phi) \tag{4.65}$$

This is indeed zero when $\phi = 0$ as shown before. However, when $\phi \neq 0$, we have:

$$x[m] = \sin(m\pi + \phi) = \begin{cases} \sin\phi \text{ if } m \text{ is even} \\ -\sin\phi \text{ if } m \text{ is odd} \end{cases} \tag{4.66}$$

i.e., the sign of $x[m]$ alternates for any phase $\phi \neq 0$ and $\phi \neq \pi$. In other words, in the marginal case $f_0 = F/2$, 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. This is shown in Fig.4.6. In particular when $\phi = \pi/2$, $x[m] = 1$ if m is even and $x[m] = -1$ if m is odd, i.e., the amplitude of the signal is accurately represented by its samples.

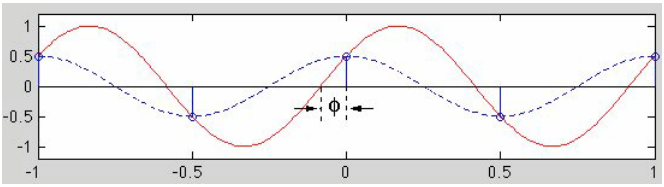

Figure 4.6 Marginal sampling: signal frequency equals Nyquist frequency $f_0 = F/2$

Example 4.5: This example further illustrates the effect of sampling and aliasing/folding. Consider a continuous signal

$$x(t) = \cos(2\pi ft + \phi) = \frac{1}{2}[e^{j(2\pi ft+\phi)} + e^{-j(2\pi ft+\phi)}] = c_1 e^{j2\pi ft} + c_{-1}e^{-j2\pi ft}$$

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:

$$\begin{aligned} x[m] = \cos(2\pi fmt_0 + \phi) = \cos(2\pi fm/F + \phi) &= \frac{e^{j\phi}}{2}e^{j2\pi fm/F} + \frac{e^{-j\phi}}{2}e^{-j2\pi fm/F} \\ &= c_1 e^{-j2\pi fm/F} + c_{-1}e^{-j2\pi fm/F} \end{aligned}$$

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.

- When $f = 1 < F/2 = 3$, the two non-zero frequency components $e^{j2\pi ft}$ and $e^{-j2\pi ft}$ are inside the middle period $-3 \sim 3$ Hz of the spectrum, based on which the signal can be perfectly reconstructed.
- When $f = 2 < F/2 = 3$, $e^{2\pi ft}$ and $e^{-j2\pi ft}$ will move outward for a higher frequency of ± 2 Hz, which are still inside the middle period.
- When $f = 3 = F/3$, the signal is marginally aliased. Depending on the relative phase difference between the signal and the sampling process, the signal is distorted to different extent. In the worst case, when the two samples happen to be taken at the zero-crossings of the signal, they are both zero and the signal $x(t) = \cos(2\pi 4f + \phi)$ is aliased to a zero signal $x(t) = 0$.
- When $f = 4 > F/2 = 3$, the two coefficients for $f = \pm 4$ are out of the middle period, but the replica of $f = 4$ Hz on the negative side moves from the left into the middle period to appear as $4 - 6 = -2$ Hz, and the replica of $f = -4$ Hz on the positive side moves from right into the middle period to appear as $-4 + 6 = 2$ Hz. The reconstructed signal based on these folded frequency components is $x'(t) = \cos(2\pi 2t - \phi)$, different from the original signal $x(t) = \cos(2\pi 4t + \phi)$.
- When $f = 5 >= F/2 = 3$, similar folding occurs and the reconstructed signal is $x'(t) = \cos(2\pi t - \phi)$.
- When $f = 6 = F$, one sample is taken per period, the aliased frequency is zero, and the reconstructed signal is $x'(t) = \cos(\phi)$
- When $f = 7 = F + 1$, the two coefficients for $f = \pm 7$ are out of the middle period, but the replica of $f = -7$ Hz on the negative side moves from the left into the middle period to appear as $-7 + 6 = -1$ Hz, and the replica of $f = 7$ on the positive side moves from the right into the middle period to appear as $7 - 6 = 1$ Hz. Based on these aliased frequency components, the reconstructed signal is $x'(t) = \cos(2\pi ft)$, which appears to be the same as the non-aliased cases when $f = 1$.
- When $f = 8 = F + 2$, similar aliasing occurs and the reconstructed signal is $x'(t) = \cos(2\pi 2t)$, which appears the same as the non-aliased case of $f = 2$.
- When $f = 9 = F + F/2$, 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.
- When $f = 12 = 2F$, 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.

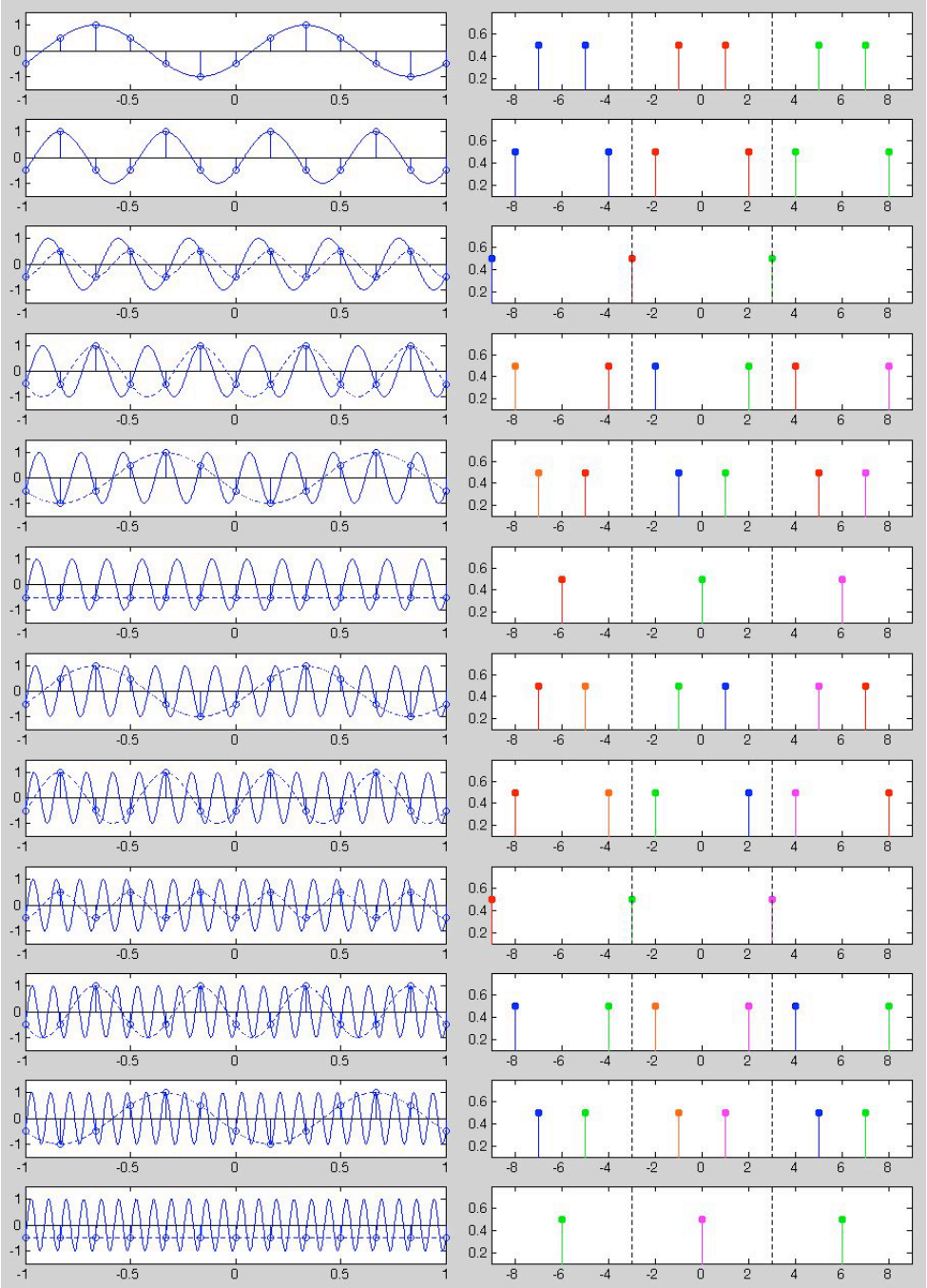

Figure 4.7 Aliasing in time and frequency domains

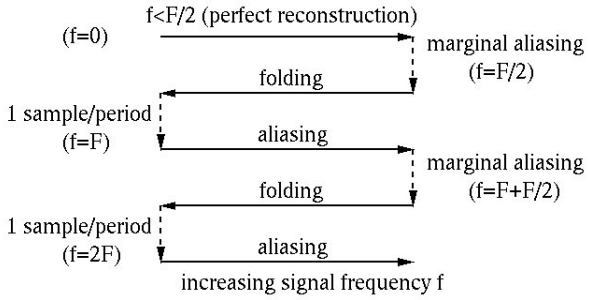

Figure 4.8 Aliasing-folding cycle as signal frequency increases

4.1.5 Reconstruction by Interpolation

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:

$$\hat{X}(f) = H(f)X_s(f) \tag{4.67}$$

Here $H(f)$ is an ideal low-pass filter that preserves all frequency components inside the central period $-F/2 < f < F/2$ of the periodic spectrum $X_s(f)$, while completely suppressing all their replicas at higher frequencies outside this period. After such an ideal low-pass filtering, we get $\hat{X}(f) = X(f)$, i.e., the signal $x(t)$ is perfectly reconstructed. In practice, as the ideal LP filter is hard to implement, a non-ideal LP filter is often used and the reconstructed signal is an approximation of the real one.

If the sampling frequency F is lower than the Nyquist frequency, any signal components with frequency $f > F/2$ will be outside the central period and therefore filtered out. However, it is possible for such frequency components to have some periodic replicas inside the central period, but they will appear to be at some lower frequencies, i.e., aliasing occurs.

In time domain, the reconstruction of the continuous signal $x(t)$ from its sampled version $x_s(t)$ is an interpolation process that fills the gaps between samples. The interpolation can be considered as a convolution of $x_s(t)$ with a certain function $h(t)$:

$$\hat{x}(t) = h(t) * x_s(t) \tag{4.68}$$

Let us consider the following reconstruction methods based on different $h(t)$ functions:

- **Zero-order hold**
  The impulse response of a *zero-order hold* filter is:

$$h_0(t) = \begin{cases} 1 \; 0 \le t < t_0 \\ 0 \text{ otherwise} \end{cases} \tag{4.69}$$

which is the rectangular function discussed before (Eq. 3.139) shifted by $t_0/2$. Based on $h_0(t)$, a continuous signal $\hat{x}_0(t)$ can be generated by:

$$\hat{x}_0(t) = h_0(t) * x_s(t) \tag{4.70}$$

which is a series of square impulses with their heights modulated by $x[m]$. The interpolation corresponds a low-pass filtering in frequency domain by

$$H_0(f) = \mathcal{F}[h_0(t)] = \frac{1}{\pi f}\sin(\pi f t_0)e^{-j2\pi f t_0/2} \tag{4.71}$$

(Eq. 3.141 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:

$$h_1(t) = \begin{cases} 1 - |t|/t_0 & |t| < t_0 \\ 0 & \text{otherwise} \end{cases} \tag{4.72}$$

which is the triangle function discussed before ($\tau = t_0$ in Eq. 3.144). A continuous signal $\hat{x}_1(t)$ can be generated by:

$$\hat{x}_1(t) = h_1(t) * x_s(t) \tag{4.73}$$

which is the linear interpolation of the sample train $x[m]$ (a straight line connecting every two consecutive samples). This interpolation corresponds a low-pass filtering in frequency domain by the following (Eq. 3.147)

$$H_1(\omega) = \mathcal{F}[h_1(t)] = \frac{1}{(\pi f)^2 t_0}\sin^2(\pi f t_0) \tag{4.74}$$

- **Ideal reconstruction**
  The reconstructed signals $\hat{x}_0(t)$ and $\hat{x}_1(t)$ are just approximations of the original signal $x(t)$, as these interpolations correspond to non-ideal low-pass filters. To find the interpolation function for a perfect reconstruction of $x(t)$, we have to use an ideal low-pass filter (scaled by t_0) in frequency domain:

$$H_{lp}(f) = \begin{cases} t_0 & |f| < f_c \\ 0 & else \end{cases} \tag{4.75}$$

with time domain impulse response (Eq. 3.143):

$$h_{lp}(t) = \mathcal{F}[H_{lp}(f)] = t_0 \, \frac{sin(2\pi f_c t)}{\pi t} \tag{4.76}$$

The reconstruction of the continuous signal $X(f)$ is realized by applying this ideal low-pass filter to the sampled signal $X_s(f)$:

$$\hat{X}(f) = H_{lp}(f) \; X_s(f) \tag{4.77}$$

If $f_{max} < F/2$, then the cut-off frequency f_c can be higher than f_{max} but lower than $F - f_{max}$, so that the central portion of $X_s(f)$ is extracted and scaled by factor t_0, while all other replicas beyond the cut-off frequency are suppressed to zero, and the original signal is perfectly reconstructed: $\hat{X}(f) = X(f)$.

This ideal low-pass filtering corresponds to the convolution in time domain:

$$\begin{aligned} h_{lp}(t) * x_s(t) &= t_0 \, \frac{sin(2\pi f_c t)}{\pi t} * \sum_{m=-\infty}^{\infty} x[m]\delta(t - mt_0) \\ &= \frac{t_0}{\pi} \sum_{m=-\infty}^{\infty} x[m] \int_{-\infty}^{\infty} \delta(\tau - mt_0) \frac{sin(2\pi f_c(t-\tau))}{t-\tau} d\tau \\ &= \frac{t_0}{\pi} \sum_{m=-\infty}^{\infty} x[m] \frac{sin(2\pi f_c(t - mt_0))}{\pi(t - mt_0)} \end{aligned} \tag{4.78}$$

4.1.6 Frequency Response Function of discrete LTI Systems

Recall that the output of a discrete LTI system $y[n] = \mathcal{O}[x[n]]$ can be found as the convolution of the input $x[n]$ and the impulse response function $h[n]$ of the system (Eq. 1.83 in Chapter 1):

$$y[n] = \mathcal{O}[x[n]] = h[n] * x[n] = \sum_{\nu=-\infty}^{\infty} x(\nu)h(n-\nu) = \sum_{\nu=-\infty}^{\infty} h(\nu)x(n-\nu) \tag{4.79}$$

In particular, let the input be a complex exponential $x[n] = e^{j\omega n} = \cos \omega n + j \sin \omega n$, then the output becomes

$$y[n] = \mathcal{O}[e^{j\omega n}] = \sum_{\nu=-\infty}^{\infty} h[\nu] e^{j\omega(n-\nu)} = e^{j\omega n} \sum_{\nu=-\infty}^{\infty} h[\nu] e^{-j\omega\nu} = e^{j\omega n} \, H(e^{j\omega}) \tag{4.80}$$

where $H(e^{j\omega})$ is defined as

$$H(e^{j\omega}) = \sum_{\nu=-\infty}^{\infty} h[\nu] e^{-j\omega\nu} = \mathcal{F}[h[n]] \tag{4.81}$$

which happens to be the discrete Fourier transform of the impulse response function $h[n]$, and is called the *frequency response function (FRF)* of the discrete LTI system. We see that this equation is an eigenequation indicating that the effect of the LTI system applied to a sinusoidal input, the eigenfunction of the system, is the same as a multiplication of the input by a constant $H(e^{j\omega})$, the eigenvalue. Also, as the sinusoidal input $x[n] = e^{j\omega n}$ is independent of any specific $h[n]$), it is the eigenfunction of *all* discrete LTI systems.

Equation 4.80 can be further written as

$$y[n] = H(\omega) \, e^{j\omega n} = |H(\omega)| e^{j\angle H(\omega)} \, e^{j\omega n} = |H(\omega)| e^{j(\omega n + \angle H(\omega))} \tag{4.82}$$

Due to the linearity of the LTI system, we can also get its response to any sinusoidal input $x[n] = \cos(\omega n) = Re[e^{j\omega n}]$ as

$$y[n] = \mathcal{O}[\cos \omega n] = Re[\, |H(e^{j\omega})| e^{j(\omega n + \angle H(e^{j\omega}))} \,] = |H(\omega)| \cos(\omega n + \angle H(\omega)) \tag{4.83}$$

In other words, the response of any discrete LTI system to a sinusoidal input is the same sinusoid with its amplitude scaled by the magnitude $|H(e^{j\omega})|$ of the FRF, and its phase shifted by the phase angle $\angle H(e^{j\omega})$ of the FRF.

As an example, consider an important class of LTI systems which can be described by a linear constant-coefficient difference equation. Here the input $x[n]$ and output $y[n]$ of the system are related by the difference equation as:

$$\sum_{k=0}^{N} a_k y[n-k] = \sum_{k=0}^{M} b_k x[n-k] \tag{4.84}$$

When the input is a complex exponential $x[n] = e^{j\omega n}$, we can assume the output is also a complex exponential $y[n] = Ye^{j\omega n}$ with a complex coefficient Y. Comparing this output $y[n]$ with Eq.4.80, we see that the coefficient Y is simply the FRF $H(e^{j\omega})$. Substituting these $x[n]$ and $y[n]$ into the differential equation above we get:

$$Y\sum_{k=0}^{N} a_k (e^{j\omega})^k e^{j\omega n} = \sum_{k=0}^{M} b_k (e^{j\omega})^k e^{j\omega n} \tag{4.85}$$

and the frequency response function can then be obtained as

$$H(e^{j\omega}) = Y = \frac{\sum_{k=0}^{M} b_k (e^{j\omega})^k}{\sum_{k=0}^{N} a_k (e^{j\omega})^k} = \frac{N(e^{j\omega})}{D(e^{j\omega})} \tag{4.86}$$

where $N(e^{j\omega}) = \sum_{k=0}^{M} b_k (e^{j\omega})^k$ and $D(e^{j\omega}) = \sum_{k=0}^{N} a_k (e^{j\omega})^k$ are the numerator and denominator of $H(e^{j\omega})$, respectively.

More generally, if the input is $x[n] = Xe^{j\omega n}$ with a complex coefficient $X = |X|e^{j\angle X}$, we can still assume an output $y[n] = Ye^{j\omega n}$, and have

$$\sum_{k=0}^{N} a_k y[n-k] = Y\sum_{k=0}^{N} a_k (e^{j\omega})^k e^{j\omega n} = \sum_{k=0}^{M} b_k (e^{j\omega})^k e^{j\omega n} = X\sum_{k=0}^{M} b_k x[n-k] \tag{4.87}$$

Now the frequency response function can be found as the ratio between the complex coefficients Y and X:

$$H(e^{j\omega}) = \frac{Y}{X} = \frac{\sum_{k=0}^{M} b_k (e^{j\omega})^k}{\sum_{k=0}^{N} a_k (e^{j\omega})^k} = \frac{N(e^{j\omega})}{D(e^{j\omega})} \tag{4.88}$$

The result above is of essential significance in the analysis of discrete LTI systems, as it can be extended much beyond sinusoidal inputs to cover any input so long as it can be expressed as a linear combination of a set of sinusoids (inverse Fourier transform):

$$x[n] = \sum_{0}^{1} X(f)e^{j2\pi ft}df = \frac{1}{2\pi}\int_{0}^{2\pi} X(e^{j\omega})e^{j\omega t}d\omega \tag{4.89}$$

Here the weighting function $X(e^{j\omega}) = \mathcal{F}[x[n]]$ is of course the spectrum of $x[n]$ obtained by the discrete-time Fourier transform. As the system is linear, we can

get the output as:

$$y[n] = \mathcal{O}[x[n]] = \frac{1}{2\pi}\int_0^{2\pi} X(e^{j\omega})\mathcal{O}[e^{j\omega t}]d\omega = \frac{1}{2\pi}\int_0^{2\pi} X(e^{j\omega})H(e^{j\omega})e^{j\omega t}d\omega \tag{4.90}$$

We see that the output $y[n]$ happens to be the inverse discrete-time Fourier transform of $H(e^{j\omega})X(e^{j\omega})$, i.e.,

$$y[n] = \mathcal{F}^{-1}[Y(e^{j\omega})] = \mathcal{F}^{-1}[H(e^{j\omega})X(e^{j\omega})] \tag{4.91}$$

In other words, in frequency domain the output is the product of the input and the frequency response function:

$$Y(e^{j\omega}) = H(e^{j\omega})X(e^{j\omega}) \tag{4.92}$$

while in time domain, the output is the convolution of the input and the impulse response function:

$$y[n] = h[n] * x[n] = \sum_{\nu=-\infty}^{\infty} h[\nu]x[n-\nu] \tag{4.93}$$

Of course, we realize this result is the same as the conclusion of the convolution theorem discussed before. Similar result can also be obtained for a periodic input which can be Fourier series expanded.

Summarizing the results above, we see the Fourier transform method is of essential significance in system analysis, as we can analyze and design an LTI system in Frequency domain to enjoy many benefits not possible in time domain. First, most obviously, the response of an LTI system to an input $x(t)$ can be much more conveniently obtained in frequency domain by a multiplication $Y(\omega) = H(\omega)X(\omega)$, instead of the corresponding convolution $y(t) = h(t) * x(t)$ in time domain. Moreover, in many applications, it may only be possible to carry out certain system analysis and design task in frequency domain. For example, if we need to design a system so that it will generate a desired response $y(t)$ to a certain input $x(t)$. In time domain, given $x(t)$ and $y(t)$, it is difficult to obtain the impulse response function $h(t)$ that satisfies $y(t) = h(t) * x(t)$. However, in frequency domain, given $X(\omega)$ and $Y(\omega)$, it is relatively straight forward to find the frequency response function $H(\omega) = \mathcal{F}[h(t)]$ by a simple division $H(\omega) = Y(\omega)/X(\omega)$.

Figure of block diagram...

4.2 Discrete Fourier Transform (DFT)

4.2.1 Formulation of DFT

In practice, most time signals are continuous and non-periodic, and their analytical expressions are not available in general. In order to obtain its spectrum in

frequency domain, which is non-periodic and continuous, by a digital computer, the signal needs to be modified in two ways:

- First, we need to truncate the signal so that it has a finite time duration from 0 to T, with the underlying assumption that the signal repeats itself outside the interval $0 < t < T$, i.e., it becomes a periodic with period T. Correspondingly, its spectrum becomes discrete. The Fourier transform of this periodic signal is the Fourier series expansion discussed before
- Second, we need to sample the signal with some sampling frequency F so that it becomes discrete to be processed by a digital computer. Correspondingly, the spectrum of the signal becomes periodic.

We can also reverse the order so that the continuous signal is sampled before truncated. In either case, only when the signal is both finite and discrete, can we apply the *discrete Fourier transform (DFT)* to find its spectrum, which is also discrete and finite.

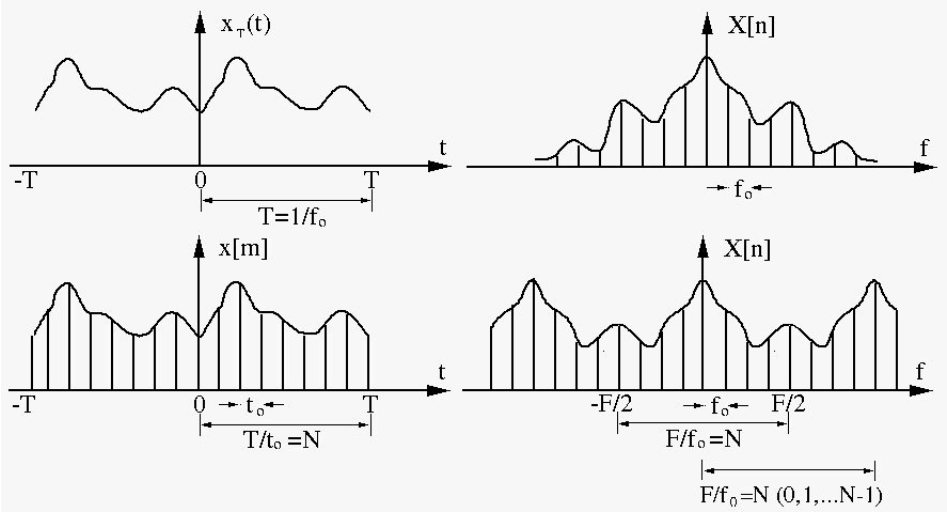

Figure 4.9 From continuous Fourier transform to discrete Fourier transform

To formulate the discrete Fourier transform, let us first recall the two different types of Fourier transform discussed before. First, when the time signal is periodic $x_T(t+T) = x_T(t)$, the coefficients $X[n]$ of its Fourier expansion form its discrete spectrum (Eq. 3.72), where the interval between two neighboring harmonics is the fundamental frequency $f_0 = 1/T$. Second, when the signal $x[m]$ is discrete as a sequence of values of a continuous signal $x(t)$ sampled at a rate of F_s, with $t_0 = 1/F$ between two consecutive samples, its spectrum $X_F(f+F) = X_F(f)$ is periodic (Eq. 4.4). 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 should be both discrete with interval $f_0 = 1/T$ between two frequency components, and periodic with frequency period $F = 1/t_0$.

In time domain, if the number of samples in a period T is

$$N = T/t_0 \tag{4.94}$$

then in frequency domain, the number of frequency components in a period F is:

$$\frac{F}{f_0} = \frac{1/t_0}{1/T} = \frac{T}{t_0} = N \tag{4.95}$$

i.e., the number of independent variables, or *degrees of freedom (DOF)*, in either time or frequency domain is preserved by the DFT. This fact should not be surprising from the view point of information conservation of the transform. Moreover, we also have the following relations that are useful later:

$$TF = \frac{T}{t_0} = N, \qquad f_0 t_0 = \frac{t_0}{T} = \frac{1}{N} \tag{4.96}$$

Now consider a continuous signal which has already been truncated with duration T and assumed to be periodic $x_T(t+T) = x_T(t)$. The sampling of this signal can be represented mathematically by multiplying the signal by the sampling (or comb) function $comb(t)$:

$$x_T(t)\, comb(t) = x_T(t) \sum_{m=-\infty}^{\infty} \delta(t - mt_0) = \sum_{m=-\infty}^{\infty} x[m]\delta(t - mt_0) \tag{4.97}$$

where $x[m] = x_T(mt_0)$ is the kth sample of the signal. Note that $x[m]$ is periodic with period N:

$$x[m+N] = x_T((m+N)t_0) = x_T(mt_0 + T) = x_T(mt_0) = x[m] \tag{4.98}$$

The Fourier coefficient of this sampled version of the signal can be found as:

$$\begin{aligned} X[n] &= \frac{1}{T}\int_T [\sum_{m=-\infty}^{\infty} x[m]\delta(t - mt_0)]\, e^{-j2\pi n f_0 t} dt \\ &= \frac{1}{T}\sum_{m=0}^{N-1} x[m] \int_T \delta(t - mt_0) e^{-j2\pi n f_0 t} dt = \frac{1}{T}\sum_{m=0}^{N-1} x[m] e^{-j2\pi n f_0 m t_0} \\ &= \frac{1}{T}\sum_{m=0}^{N-1} x[m] e^{-j2\pi nm/N}, \qquad (n = 0, 1, \cdots, N-1) \end{aligned} \tag{4.99}$$

The number of terms in the summation is reduced from infinity to N for those inside the integral range of T, as all those terms outside this range make no contribution to the integral. Note that $X[n+N] = X[n]$ is also periodic with period N:

$$\begin{aligned} X[n+N] &= \frac{1}{T}\sum_{m=0}^{N-1} x[m] e^{-j2\pi(n+N)m/N} \\ &= \frac{1}{T}\sum_{m=0}^{N-1} x[m] e^{-j2\pi nm/N} e^{-j2m\pi} = X[n] \end{aligned} \tag{4.100}$$

The inverse transform can be found by multiplying both sides of Eq.4.99 by $e^{j2\pi\mu n/N}/F$, and taking summation with respect to n from 0 to $N-1$:

$$\begin{aligned}
&\frac{1}{F}\sum_{n=0}^{N-1} X[n]e^{j2\pi\mu n/N} = \frac{1}{F}\sum_{n=0}^{N-1}[\frac{1}{T}\sum_{m=0}^{N-1} x[m]e^{-j2\pi mn/N}]e^{j2\pi\mu n/N} \\
&= \sum_{m=0}^{N-1} x[m]\frac{1}{N}\sum_{n=0}^{N-1} e^{j2\pi n[\mu-m]/N} = \sum_{m=0}^{N-1} x[m]\delta[\mu-m] \\
&= x[\mu], \qquad (\mu = 0, 1, \cdots, N-1)
\end{aligned} \tag{4.101}$$

Here we have used Eq.1.29 shown before.

Now we put Equations 4.99 and 4.101 together to form the DFT pair:

$$\begin{aligned}
X[n] &= \frac{1}{T}\sum_{m=0}^{N-1} x[m]e^{-j2\pi mn/N}, \quad (n = 0, 1, \cdots, N-1) \\
x[m] &= \frac{1}{F}\sum_{n=0}^{N-1} X[n]e^{j2\pi mn/N}, \quad (m = 0, 1, \cdots, N-1)
\end{aligned} \tag{4.102}$$

The first equation is the forward DFT while the second one the inverse DFT (note the summation index μ is replaced by m). As both $x[m]$ and $X[n]$ 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 now make a trivial modification of the transform above by redefining the coefficient as $X[n]/F$, then the above DFT pair becomes:

$$\begin{aligned}
X[n] &= \frac{1}{N}\sum_{m=0}^{N-1} x[m]e^{-j2\pi mn/N} \quad (n = 0, 1, \cdots, N-1) \\
x[m] &= \sum_{n=0}^{N-1} X[n]e^{j2\pi mn/N} \quad (m = 0, 1, \cdots, N-1)
\end{aligned} \tag{4.103}$$

Actually the scaling factor has little significance in practice. The factor $1/N$ can be moved from the first equation (forward DFT) to the second (inverse DFT) transform (such as in the Matlab implementation of DFT).

However, we prefer to put a scaling factor $1/\sqrt{N}$ in front of both forward and inverse transforms:

$$\begin{aligned}
X[n] &= \frac{1}{\sqrt{N}}\sum_{m=0}^{N-1} x[m]e^{-j2\pi mn/N} \quad (n = 0, 1, \cdots, N-1) \\
x[m] &= \frac{1}{\sqrt{N}}\sum_{n=0}^{N-1} X[n]e^{j2\pi mn/N} \quad (m = 0, 1, \cdots, N-1)
\end{aligned} \tag{4.104}$$

for the reason that now a set of N orthonormal vectors can be defined as:

$$\boldsymbol{\phi}_n = \frac{1}{\sqrt{N}}[e^{j2\pi 0m/N}, \cdots, e^{j2\pi(N-1)m/N}]^T, \quad (n = 0, \cdots, N-1) \tag{4.105}$$

which satisfy

$$< \boldsymbol{\phi}_n, \boldsymbol{\phi}_{n'} >= \boldsymbol{\phi}_n^T \overline{\boldsymbol{\phi}}_{n'} = \frac{1}{N} \sum_{m=0}^{N-1} e^{j2\pi(n-k)m/N} = \delta[n-n'] \tag{4.106}$$

Similar to the kth basis function $\phi_k(t) = e^{j2\pi f_k t}$ for the Fourier series expansion, which corresponds to a continuous sinusoidal function of frequency $f_k = kf_0 = k/T$ (k cycles per period of T seconds), here the kth basis vector $\boldsymbol{\phi}_k = e^{j2\pi mk/N}$ corresponds to a discrete sinusoidal function of frequency $f_k = k/N$ (k cycles per period of N samples). N such vectors of different frequencies $f_k = k/N$ ($k = 0, \cdots, N-1$) form a complete basis that spans the space $\mathbb{C}^N$. Any signal vector originally given as $\boldsymbol{x} = [x[0], \cdots, x[N-1]]^T$ under the standard basis $\boldsymbol{e}_n$ can now be represented under the DFT basis $\boldsymbol{\phi}_n$ as

$$\boldsymbol{x} = \sum_{n=0}^{N-1} X[n]\boldsymbol{\phi}_n \tag{4.107}$$

which can also be expressed in component form as:

$$x[m] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X[n] e^{j2\pi mn/N}, \quad (m = 0, \cdots, N-1) \tag{4.108}$$

This is the inverse DFT where the coefficient $X[n]$ can be obtained by taking the inner product with $\boldsymbol{\phi}_{n'}$ on both sides of Eq.4.107:

$$\begin{aligned} < \boldsymbol{x}, \boldsymbol{\phi}_{n'} > &= < \sum_{n=0}^{N-1} X[n]\boldsymbol{\phi}_n,\ \boldsymbol{\phi}_{n'} >= \sum_{n=0}^{N-1} X[n] < \boldsymbol{\phi}_n,\ \boldsymbol{\phi}_{n'} > \\ &= \sum_{n=0}^{N-1} X[n]\delta[n-n'] = X[n'] \end{aligned} \tag{4.109}$$

We see that $X[n]$ is simply the projection of the signal vector $\boldsymbol{x}$ onto the nth basis vector $\boldsymbol{\phi}_n$:

$$X[n] =< \boldsymbol{x}, \boldsymbol{\phi}_n >= \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} x[m] e^{-j2\pi mn/N}, \quad (n = 0, \cdots, N-1) \tag{4.110}$$

This is the forward DFT.

As here both the signal and its spectrum are discrete and periodic, and therefore finite as only one period is needed in the computation, the DFT is the only form of Fourier transform that can be actually carried out by a digital computer. Moreover, due to the Fast Fourier Transform (FFT) algorithm to be discussed later,

It is important to know how to obtain meaningful data and properly interpret the result. The Fourier coefficients $X[n]$ obtained by the DFT are obviously related to, but certainly not equal to, the spectrum $X(f)$ of the continuous signal $x(t)$ originally give, as the truncation and sampling process has significantly

modified the signal 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 is most likely to occur at the point between two periods of the duration T. Second, due to the sampling process, aliasing may occur if the sampling rate F is not higher than the Nyquist frequency, twice the highest frequency component in the signal. The spectrum may be contaminated by various artifacts, most likely some faulty high frequency components corresponding to the discontinuities, together with some aliased faulty frequencies caused by sampling. It is therefore important to pay special attention to the truncation and sampling process in order to minimize such artifacts. For example, we can use some windowing technique to smooth the truncated signal, as well as low-pass filtering to reduce the high frequency components before sampling to reduce aliasing. Only then, can the DFT generate meaningful data representative of the actual signal of interest.

Example 4.6: Consider a sinusoid of frequency k/N (k cycles per N points)...

$$x[m] = \cos(m\frac{2\pi}{5}) = \frac{1}{2}[e^{j2\pi m/5} + e^{-j2\pi m/5}] \tag{4.111}$$

with $\omega_0 = 2\pi/5$, or period $N = 5$. Comparing this expression with the DFT expansion:

$$x[m] = \sum_{n=0}^{n=4} X[n]e^{j2\pi mn/5} \tag{4.112}$$

we see that $X[1] = X[-1] = 1/2$. Alternatively, following the DFT we can also get the nth Fourier coefficient as:

$$\begin{aligned} X[n] &= \frac{1}{N}\sum_{m=0}^{N-1} x[m]e^{-j2\pi mn/N} = \frac{1}{10}\sum_{m=-2}^{2}[e^{-j2\pi m/5}e^{-j2\pi mn/5} + e^{-j2\pi m/5}e^{-j2\pi mn/5}] \\ &= \frac{1}{10}\sum_{m=-2}^{2}[e^{-j2\pi m(1-n)/5} + e^{-j2\pi m(1+m)n/5}] = \frac{1}{2}[\delta[n+1] + \delta[n-1]] \end{aligned} \tag{4.113}$$

Example 4.7: Consider a symmetric square wave with a period of N samples:

$$x[m] = \begin{cases} 1 & |m| \le N_1 \\ 0 & N_1 < |m| \le N/2 \end{cases} \tag{4.114}$$

For convenience, we choose the limits of the Fourier transform summation from $-N/2$ to $N/2-1$, instead of from 0 to $N-1$ to get

$$X[n] = \sum_{m=-N/2}^{N/2-1} x[m]e^{-j2\pi mn/N} = \sum_{m=-N_1}^{N_1} e^{-j2\pi mn/N} \tag{4.115}$$

Let $m' = m + N_1$, we have $m = m' - N_1$ and

$$\begin{aligned} X[n] &= \sum_{m'=0}^{2N_1} e^{-j2\pi m'n/N} e^{j2\pi N_1 n/N} \\ &= e^{j2\pi N_1 n/N} \frac{1 - e^{-j2\pi(2N_1+1)n/N}}{1 - e^{-j2\pi n/N}} \\ &= e^{j2\pi N_1 n/N} \frac{e^{-j\pi(2N_1+1)n/N}(e^{j\pi(2N_1+1)n/N} + e^{-j\pi(2N_1+1)n/N})}{e^{-j\pi n/N}(e^{j\pi n/N} - e^{-j\pi n/N})} \\ &= \frac{sin((2N_1+1)n\pi/N)}{sin(n\pi/N)} \end{aligned} \tag{4.116}$$

4.2.2 Four different forms of Fourier transform

Various forms of the Fourier transform for periodic/non-periodic and continuous/discrete signals discussed in this chapter and the previous one can be considered as four different variations of the same Fourier transform as shown below.

- **I. Non-periodic continuous signal, continuous, non-periodic spectrum**
  This is the most general form of the Fourier transform of a continuous and non-periodic signal $x(t)$, considered as a function in a function space spanned by a set of uncountable basis functions $\phi_f(t) = e^{j2\pi ft}$ $(-\infty < f < \infty)$ that are orthonormal according to Eq.1.26:

  $$< \phi_f(t), \phi_{f'}(t) >= \int_{-\infty}^{\infty} e^{j2\pi(f-f')t} dt = \delta(f - f')$$

  The signal $x(t)$ can therefore be expressed as a linear combination (integral) of these uncountable basis functions as:

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

  This is the inverse transform and the weighting coefficient function $X(f)$ can be obtained as the projection of the signal onto each basis function:

  $$X(f) =< x(t), \phi_f(t) >=< x(t), e^{j2\pi ft} >= \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt$$

  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 function in the space of or periodic functions spanned by a set of countable basis functions $\phi_n(t) = e^{j2\pi nt/T}/\sqrt{T}$ (for all integer n) that are orthonormal according to Eq.1.27:

$$< \phi_m(t), \phi_n(t) >= \int_T e^{j2\pi(m-n)t/T} dt = \delta[n-n']$$

The signal $x_T(t)$ can therefore be expressed as a linear combination (summation) of these basis functions as:

$$x_T(t) = \sum_{n=-\infty}^{\infty} X[n]\phi_n(t) = \sum_{n=-\infty}^{\infty} X[f]e^{j2\pi nt/T}$$

This is the inverse transform and the weighting coefficient $X[n]$ can be obtained as the projection of the signal onto the nth basis function:

$$X[n] =< x_T(t), \phi_n(t) >=< x_T(t), \frac{1}{\sqrt{T}} e^{j2\pi nt/T} >= \frac{1}{\sqrt{T}} \int_T x_T(t) e^{-j2\pi nt/T} dt$$

This is the forward transform. As two consecutive frequency components are separated by $f = 1/T$, the spectrum of the periodic signal can be expressed as a continuous function:

$$X(f) = \sum_{n=-\infty}^{\infty} X[n]\delta(f - nf_0)$$

- **III. Non-periodic discrete signal, continuous periodic spectrum**
  This is the discrete-time Fourier transform of a discrete and non-periodic signal

$$x(t) = \sum_{m=-\infty}^{\infty} x[m]\delta(t - mt_0)$$

These signal samples $x[m]$ (for all integer m) form an infinite dimensional vector $\boldsymbol{x} = [\cdots, x[m], \cdots]^T$ in the vector space of all such vectors spanned by an uncountable set of basis vectors $\boldsymbol{\phi}_f = [\cdots, e^{j2\pi nf/F}/\sqrt{F}, \cdots]^T$ (for all $0 < f < F$) that are orthonormal according to Eq.1.28:

$$< \boldsymbol{\phi}_f, \boldsymbol{\phi}_{f'} >= \frac{1}{F} \sum_{m=-\infty}^{\infty} e^{j2\pi m(f-f')} = \sum_{k=-\infty}^{\infty} \delta(f - f' - nF)$$

The signal $\boldsymbol{x}$ can therefore be expressed as a linear combination (integral) of these uncountable basis vectors as:

$$\boldsymbol{x} = \int_F X(f)\boldsymbol{\phi}_f df$$

or in component form:

$$x[m] = \frac{1}{\sqrt{F}} \int_F X(f) e^{j2\pi mf/F} df$$

This is the inverse transform, and the weighting coefficient function $X(f)$ can be obtained as the projection of the signal onto each basis function:

$$X(f) =< \boldsymbol{x}, \boldsymbol{\phi}_f >= \frac{1}{\sqrt{F}} \sum_{m=-\infty}^{\infty} x[m] e^{-j2\pi mf/F}$$

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 $x[m],\ m = 0, \cdots, N-1$. The N samples form an N-D vector $\boldsymbol{x} = [x[0], \cdots, x[N-1]]^T$ in a N-D unitary space spanned by a set of N N-D vectors $\boldsymbol{\phi}_n = [e^{j2\pi 0n/N}, \cdots, e^{j2\pi(N-1)n/N}]^T/\sqrt{N}$ that are orthonormal according to Eq.1.29:

$$< \boldsymbol{\phi}_n, \boldsymbol{\phi}_{n'} >= \frac{1}{N} \sum_{k=0}^{N-1} e^{j2\pi k(n-n')/N} = \delta[n-n']$$

The signal vector can therefore be expressed as a linear combination (summation) of the N basis vectors:

$$\boldsymbol{x} = \sum_{n=0}^{N-1} X[n] \boldsymbol{\phi}_n$$

or in component form:

$$x[m] = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X[n] e^{j2\pi mn/N}, \qquad (m = 0, 1, \cdots, N-1)$$

This is the inverse transform, and the weighting coefficient $X[n]$ can be obtained as the projection of the signal onto each basis function:

$$X[n] =< \boldsymbol{x}, \boldsymbol{\phi}_n >= \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} x[m] e^{-j2\pi mn/N}, \qquad (n = 0, 1, \cdots, N-1)$$

As the previous two cases, the discrete time samples can be considered as a special continuous function:

$$x_T(t) = \sum_{m=0}^{N-1} x[m] \delta(t - mt_0) \tag{4.117}$$

and the discrete frequency coefficients can be considered as a special continuous spectrum:

$$X_F(f) = \sum_{n=0}^{N-1} X[n] \delta(f - nf_0) \tag{4.118}$$

The four forms of Fourier transform can be summarized as below:

	The signal $x(t)$	The 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) $x_T(t) = \sum_{n=-\infty}^{\infty} X[n]e^{j2\pi nf_0t}$	Non-periodic, Discrete $(f_0 = 1/T)$ $X[n] = \int_T x_T(t)e^{-j2\pi nf_0t}dt/T$ $X(f) = \sum_{n=-\infty}^{\infty} X[n]\delta(f - nf_0)$
III	Discrete (t_0), Non-periodic $x(t) = \sum_{m=-\infty}^{\infty} x[m]\delta(t - mt_0)$ $x[m] = \int_F X_F(f)e^{j2\pi fmt_0}df/F$	Periodic $(F = 1/t_0)$, Continuous $X_F(f) = \sum_{m=-\infty}^{\infty} x[m]e^{-j2\pi fmt_0}$
IV	Discrete (t_0), Periodic (T) $x_T[m] = \sum_{n=0}^{N-1} X[n]e^{j2\pi nm/N}$ $x_T(t) = \sum_{m=0}^{N-1} x[m]\delta(t - mt_0)$ $T/t_0 = N$	Periodic $(F = 1/t_0)$, Discrete $(f_0 = 1/T)$ $X_F[n] = \sum_{m=0}^{N-1} x[m]e^{-j2\pi mn/N}$ $X_F(f) = \sum_{n=0}^{N-1} X[n]\delta(f - nf_0)$ $F/f_0 = T/t_0 = N$

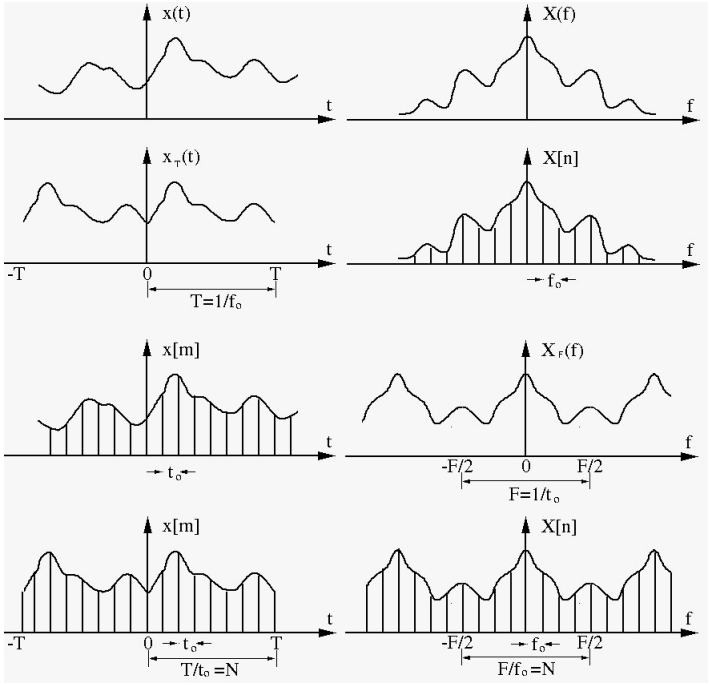

Figure 4.10 Four different forms of Fourier transform

All four forms of the Fourier transform share the same properties, discussed above mostly thoroughly for the continuous and non-periodic case, although they may take different forms for each of the four cases.

4.2.3 Physical Interpretation of DFT

Computationally, given N values $x[m]$ ($m = 0, \cdots, N-1$), any DFT code can generate N complex values $X[n]$ ($n = 0, \cdots, N-1$). But how should these N values interpreted? What does each of them mean specifically? How should they be manipulated to achieve certain data processing goals such as filtering? We will address these issues here.

In reality, a time signal $x[m] = x_r[m]$ is real ($x_j[m] = 0$), and it can be expanded as:

$$x[m] = x_r[m] = Re[\frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X[n] e^{j2\pi mn/N}] \tag{4.119}$$

$$= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} [X_r[n] \cos(2\pi mn/N) - X_j[n] \sin(2\pi mn/N)]$$

$$= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} |X[n]| \cos(2\pi mn/N + \angle X[n]) \tag{4.120}$$

where

$$\begin{cases} |X[n]| = \sqrt{X_r^2[n] + X_j^2[n]} \\ \angle X[n] = \tan^{-1}(X_j[n]/X_r[n]) \end{cases}, \qquad \begin{cases} X_r[n] = |X[n]| \cos \angle X[n] \\ X_j[n] = |X[n]| \sin \angle X[n] \end{cases} \tag{4.121}$$

As both $X[n]$ and $\cos(2\pi mn/N)$ are periodic with period N, the summation above can be changed to be from $-N/2+1$ to $N/2$:

$$x[m] = \frac{1}{\sqrt{N}} \sum_{n=-N/2}^{N/2-1} |X[n]| \cos(2\pi mn/N + \angle X[n]) \tag{4.122}$$

Consider the different types of terms in this summation:

- $n = 0$:
  $X[0]$ is the DC component, which is real (zero phase);
- $n = N/2$:
  $X[N/2]$ is the coefficient for the highest frequency component $\cos(m\pi) = (-1)^m$, which is real without phase shift;
- $n = 1, \cdots, N/2-1$:
  These are $(N-2)/2$ sinusoids $|X[n]| \cos(2\pi mn/N + \angle X[n])$ with frequency n, amplitude $|X[n]|$ and phase shift $\angle X[n]$;
- $n = -N/2+1, \cdots, -1$:
  This range for index n is equivalent to a range $(1, \cdots, N/2-1)$ for index $-n$, and, as $X_r[-n] = X_r[n]$ is even and $X_j[-n] = -X_j[n]$ is odd, we know $|X[-n]| = |X[n]|$ is even and $\angle X[-n] = \angle X[n]$ is odd. Now each term in this range becomes identical to a corresponding term in the previous range $n =$

$1, \cdots, N/2-1$:

$$|X[-n]|\cos(-2\pi mn/N + \angle X[-n]) = |X[n]|\cos(-2\pi mn/N - \angle X[-n])$$
$$= |X[n]|\cos(2\pi mn/N + \angle X[n]), \quad (n = 1, \cdots, N/2-1) \tag{4.123}$$

Now the real signal $x[m]$ can be expanded as:

$$x[m] = \frac{1}{\sqrt{N}}[X[0] + 2\sum_{n=1}^{N/2-1} |X[n]|\cos(2\pi mn/N + \angle X[n]) + X[N/2]\cos(m\pi)] \tag{4.124}$$

This is the discrete version of Eq. 3.70 in the case of the continuous Fourier transform.

In general, the N complex coefficients generated by any DFT code (e.g., Matlab, or C-code) are indexed from 0 to $N-1$ (or sometimes from 1 to N), with the DC component $X[0]$ at the front end and the coefficient $X[N/2]$ for the highest frequency component in the middle. On the other hand, it is conventional conceptually for the DC component to be in the middle, i.e., the coefficients are indexed from $-N/2$ to $N/2-1$. To convert the actual output of a DFT code to fit this convention, one could rearrange the N output data points in frequency domain so that they are shifted by $N/2$. Alternatively, according to the frequency shift property of the Fourier transform, if we multiply each data point $x[m]$ in time domain by $e^{(j2\pi mN/2)/N} = e^{jm\pi} = (-1)^m$, i.e., negate the sign of every other time sample, the corresponding spectrum in frequency domain will be shifted by $N/2$.

Now in frequency domain, various filtering (e.g., low, band or high-pass/stop) can be carried out by modifying (increasing or reducing) the coefficients corresponding to different frequency components, before inverse transforming back to time domain.

4.2.4 Array Representation

A set of N-dimensional vectors can be defined based on the complex exponential functions $e^{\pm j2\pi mn/N}$ appearing in the DFT:

$$\boldsymbol{w}_n = [w_{0n}, \cdots, w_{N-1,n}]^T, \qquad (n = 0, \cdots, N-1) \tag{4.125}$$

where $w_{mn} = e^{j2\pi kn/N}/\sqrt{N}$. (Here we assume a scaling factor $1/\sqrt{N}$ in front of both forward and inverse DFT .) These vectors are orthonormal:

$$< \boldsymbol{w}_m, \boldsymbol{w}_n > = \boldsymbol{w}_m^T \overline{\boldsymbol{w}}_n = \frac{1}{N}\sum_{k=0}^{N-1} e^{j2\pi mk/N} e^{-j2\pi nk/N}$$
$$= \frac{1}{N}\sum_{k=0}^{N-1} e^{j2\pi(m-n)k/N} = \delta[m-n] \tag{4.126}$$

and they therefore form a complete orthogonal system that span an N-dimensional unitary space, of which the vectors containing the discrete signal

samples as well as their Fourier coefficients are members:

$$\boldsymbol{x} = [x[0], \cdots, x[N-1]]^T, \qquad \boldsymbol{X} = [x[0], \cdots, x[N-1]]^T \tag{4.127}$$

Now a signal vector can be expressed by these basis vectors:

$$\boldsymbol{x} = \boldsymbol{W}\boldsymbol{X} = [\boldsymbol{w}_0, \cdots, \boldsymbol{w}_{N-1}] \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \sum_{n=0}^{N-1} X[n]\boldsymbol{w}_n \tag{4.128}$$

where $\boldsymbol{W}$ is an $N \times N$ unitary matrix composed of N orthonormal vectors:

$$\boldsymbol{W} = [\boldsymbol{w}_0, \cdots, \boldsymbol{w}_{N-1}] = \frac{1}{\sqrt{N}} \begin{bmatrix} e^{j2\pi 00/N} & e^{j2\pi 01/N} & \cdots & e^{j2\pi 0(N-1)/N} \\ e^{j2\pi 10/N} & e^{j2\pi 11/N} & \cdots & e^{j2\pi 1(N-1)/N} \\ \vdots & \vdots & \ddots & \vdots \\ e^{j2\pi(N-1)0/N} & e^{j2\pi(N-1)1/N} & \cdots & e^{j2\pi(N-1)(N-1)/N} \end{bmatrix} \tag{4.129}$$

and the element in the mth row and nth column is $w[m,n] = e^{j2\pi mn/N} = w[n,m]$, i.e., $\boldsymbol{W} = \boldsymbol{W}^T$ is a symmetric unitary matrix: $\boldsymbol{W}^{-1} = \overline{\boldsymbol{W}}$. Left multiplying $\boldsymbol{W}^{-1} = \overline{\boldsymbol{W}}$ on both sides of Eq. 4.128, we get

$$\overline{\boldsymbol{W}}\boldsymbol{x} = \overline{\boldsymbol{W}}\boldsymbol{W}\boldsymbol{X} = \boldsymbol{X} \tag{4.130}$$

If we write $\overline{\boldsymbol{W}}$ in terms of its row vectors:

$$\overline{\boldsymbol{W}} = \begin{bmatrix} \overline{\boldsymbol{w}}_0^T \\ \vdots \\ \overline{\boldsymbol{w}}_{N-1}^T \end{bmatrix} \tag{4.131}$$

then Eq.4.130can be written as

$$\boldsymbol{X} = \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \overline{\boldsymbol{W}}\boldsymbol{x} = \begin{bmatrix} \overline{\boldsymbol{w}}_0^T \\ \vdots \\ \overline{\boldsymbol{w}}_{N-1}^T \end{bmatrix} \boldsymbol{x} \tag{4.132}$$

where the Nth coefficient

$$X[n] = \overline{\boldsymbol{w}}_n^T \boldsymbol{x} = < \boldsymbol{x}, \boldsymbol{w}_n > \tag{4.133}$$

can be considered as the projection of the signal vector $\boldsymbol{x}$ onto the nth basis vector $\boldsymbol{w}_n$ of the N-D unitary space.

Equations 4.128 and 4.130 form the DFT pair in matrix form:

$$\begin{cases} \boldsymbol{X} = \overline{\boldsymbol{W}}\boldsymbol{x} & \text{(forward)} \\ \boldsymbol{x} = \boldsymbol{W}\boldsymbol{X} & \text{(inverse)} \end{cases} \tag{4.134}$$

The component form of these two equations are the same as Eq. 4.103 (except now the scaling constant $1/\sqrt{N}$ appears on both equations).

As a unitary transformation in the N-dimensional unitary space, the DFT can be considered as a rotation represented by the unitary matrix $\boldsymbol{W}$. A signal vector

$\boldsymbol{x} = [x[0], \cdots, x[N-1]]^T$ composed of N samples is originally given under the standard basis

$$\boldsymbol{x} = \sum_{m=0}^{N-1} x[m]\boldsymbol{e}_m = \boldsymbol{I}\boldsymbol{x} \tag{4.135}$$

but it can also be expressed in terms of a different set of basis vectors $\boldsymbol{w}_n$ $(n = 0, \cdots, N-1)$, obtained by rotating the standard basis vectors $\boldsymbol{e}_n$ by the rotation matrix $\boldsymbol{W}$, therefore

$$\boldsymbol{x} = \boldsymbol{W}\boldsymbol{X}, \quad \text{or} \quad \boldsymbol{X} = \overline{\boldsymbol{W}}\boldsymbol{x} \tag{4.136}$$

As rotation does not change the norm of a vector (Parseval's equality) the norm of the signal is conserved $||\boldsymbol{x}|| = ||\boldsymbol{X}||$, i.e., either the original signal $\boldsymbol{x}$ in time domain or its Fourier coefficients X in frequency domain contains the same among of energy or information.

Consider the following three examples for N=2, 4 and 8. First when $N = 2$, the element of the mth row and nth column of the 2-point DFT matrix is

$$w[m,n] = w[n,m] = \frac{1}{\sqrt{2}}(e^{j2\pi/N})^{mn} = \frac{1}{\sqrt{2}}(e^{j\pi})^{mn} = \frac{1}{\sqrt{2}}(-1)^{mn}, \quad (m,n = 0,1) \tag{4.137}$$

and the DFT matrix can be found to be:

$$\boldsymbol{W}_{2\times 2} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{4.138}$$

Now the DFT of the given 2-point signal $\boldsymbol{x} = [x[0], x[1]]^T$ can be easily found to be

$$\boldsymbol{X} = \begin{bmatrix} X[0] \\ X[1] \end{bmatrix} = \overline{\boldsymbol{W}}\boldsymbol{x} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} x[0] \\ x[1] \end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix} x[0]+x[1] \\ x[0]-x[1] \end{bmatrix} \tag{4.139}$$

We see that the first component $X[0]$ is proportional to the sum of the two signal samples $x[0] + x[1]$ representing the average or DC component of the signal, and second $X[1]$ is proportional to the difference between the two samples $x[0] - x[1]$.

When $N = 4$, the element of the mth row and nth column of the 4-point DFT matrix is

$$w[m,n] = w[n,m] = \frac{1}{\sqrt{N}}(e^{j2\pi/N})^{mn} = \frac{1}{2}(e^{j\pi/2})^{mn} = j^{mn}, \quad (m,n = 0,\cdots,3) \tag{4.140}$$

The 4 by 4 DFT matrix can be found to be:

$$\boldsymbol{W}_{4\times 4} = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & j & -1 & -j \\ 1 & -1 & 1 & -1 \\ 1 & -j & -1 & j \end{bmatrix} \tag{4.141}$$

We can easily verify that $\boldsymbol{W} = \boldsymbol{W}^T$ and $\boldsymbol{W}\overline{\boldsymbol{W}} = \boldsymbol{I}$.

When $N = 8$, the real and imaginary parts of the DFT matrix $\boldsymbol{W} = \boldsymbol{W}_r + j\boldsymbol{W}_j$ are respectively:

$$\boldsymbol{W}_r = \frac{1}{\sqrt{8}} \begin{bmatrix} 1.0 & 1.0 & 1.0 & 1.0 & 1.0 & 1.0 & 1.0 & 1.0 \\ 1.0 & 0.7 & 0.0 & -0.7 & -1.0 & -0.7 & 0.0 & 0.7 \\ 1.0 & 0.0 & -1.0 & 0.0 & 1.0 & 0.0 & -1.0 & -0.0 \\ 1.0 & -0.7 & 0.1 & 0.7 & -1.0 & 0.7 & 0.0 & -0.7 \\ 1.0 & -1.0 & 1.0 & -1.0 & 1.0 & -1.0 & 1.0 & -1.0 \\ 1.0 & -0.7 & 0.0 & 0.7 & -1.0 & 0.7 & 0.0 & -0.7 \\ 1.0 & 0.0 & -1.0 & 0.0 & 1.0 & 0.0 & -1.0 & -0.0 \\ 1.0 & 0.7 & 0.0 & -0.7 & -1.0 & -0.7 & 0.0 & 0.7 \end{bmatrix} \tag{4.142}$$

and

$$\boldsymbol{W}_j = \frac{1}{\sqrt{8}} \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & -0.7 & -1.0 & -0.7 & 0.0 & 0.7 & 1.0 & 0.7 \\ 0.0 & -1.0 & 0.0 & 1.0 & 0.0 & -1.0 & 0.0 & 1.0 \\ 0.0 & -0.7 & 1.0 & -0.7 & 0.0 & 0.7 & -1.0 & 0.7 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.7 & -1.0 & 0.7 & 0.0 & -0.7 & 1.0 & -0.7 \\ 0.0 & 1.0 & 0.0 & -1.0 & 0.0 & 1.0 & 0.0 & -1.0 \\ 0.0 & 0.7 & 1.0 & 0.7 & 0.0 & -0.7 & -1.0 & -0.7 \end{bmatrix} \tag{4.143}$$

The DFT can be considered as the discrete version of the Fourier series expansion by sampling both the continuous signal and the basis functions at a rate of N samples per period T. However, the physical interpretation of the nth basis vector $\phi_n[m] = e^{j2\pi mn/N}$ in the DFT (Eq. 4.103) is different from that of the nth basis function $\phi_n(t) = e^{j2\pi nf_0 t}$ in the Fourier series expansion (Eq. 3.6). The frequency of $\phi_n(t)$ is nf_0, which grows without limit as n increases, but the frequency of $\phi_n[m]$ does not grow without limit. For example, the frequency corresponding to $\phi_{N-1}[m] = e^{j2\pi(N-1)m/N}$ is not $N - 1/N$ per period T, but $1/N$ one cycle per T. This is because $\phi_n[m]$ is the discrete version of $\phi_n(t)$ obtained by sampling it at a rate of N per period T, and any frequency $n > N/2$ per period T is aliased to appear as n_N.

Now let us consider the real and imaginary parts of the first $N = 8$ basis functions $\phi_n(t) = e^{j2\pi nt/N}$ $(n = 0, \cdots, 7)$ for the Fourier series expansion are shown in Fig.4.11, together with the N discrete samples (the circles) for each of the basis vectors for the DFT, which are also given in Eqs. 4.142 and 4.143. Note that $\phi_1(t)$, $\phi_2(t)$ and $\phi_3(t)$ represent, respectively, frequencies of 1, 2 and 3 cycles per period T, but $\phi_5(t)$, $\phi_6(t)$, and $\phi_7(t)$ actually represent frequencies of 3, 2 and 1 cycles per period (the dashed lines in the figure), instead of 5, 6 and 7 cycles per period, due to aliasing. Also, in particular, the 0th basis function $\phi_0(t) = 1$ is a constant representing the DC component of the signal, while the 4th basis function $\phi_4(t)$ has the highest frequency of $N/2 = 4$ cycles per period T.

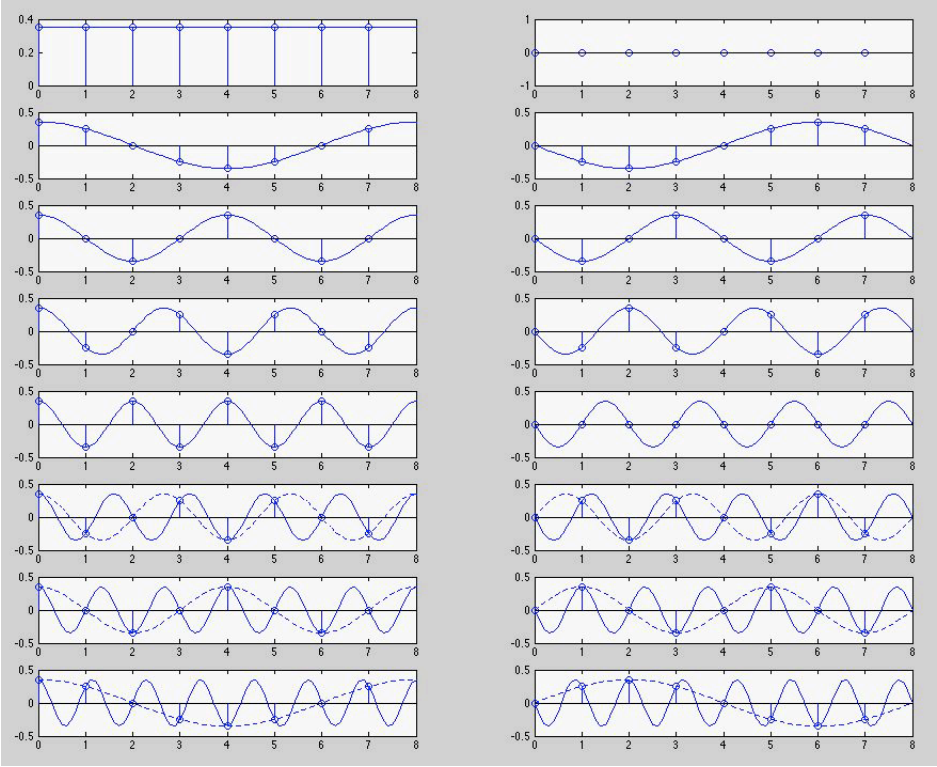

Figure 4.11 The basis vectors of 8-point DFT ($n = 0, \cdots, 7$ top-down, real on the left, imaginary on the right)

The DFT and inverse DFT can be implemented easily in either Matlab or C language.

```
function X=dft(x)
N=length(x);
X=[exp(j*2*pi*[0:N-1]'*[0:N-1]/N)/sqrt(N)]*x;

function x=idft(X)
N=length(x);
x=[exp(j*2*pi*[0:N-1]'*[0:N-1]/N)/sqrt(N)]*X;
```

Here the signal x and its spectrum X are assumed to be column vectors (same as in the text). If they are row vectors, we can simply reverse the order of the two expressions in the multiplication. The C source code for DFT and inverse DFT is listed below:

```
void dft(xr,xi,N,forward)
float *xr,*xi;
```

```
int N,forward;
{ int i,j,k,m,n;
  float arg,s,c,*yr,*yi;
  yr=(float *)malloc(N*sizeof(float));
  yi=(float *)malloc(N*sizeof(float));
  for (n=0; n<N; n++) {
    yr[n]=yi[n]=0;
    for (m=0; m<N; m++) {
      arg=pi*m*n/N;
      if (forward) arg=-arg;
      yr[n]=yr[n]+cos(arg)*xr[m]-sin(arg)*xi[m];
      yi[n]=yi[n]+cos(arg)*xi[m]+sin(arg)*xr[m];
    }
  }
  for (n=0; n<N; n++) {
    xr[n]=yr[n]/sqrt(N); xi[n]=yi[n]/sqrt(N);
  }
  free(yr);
  free(yi);
}
```

This function carries out DFT if forward=1, or inverse DFT otherwise. Note that all these implementations have the same computational complexity of $O(N^2)$.

Example 4.8: Find the DFT of a real signal of $N = 8$ samples: $[0, 0, 2, 3, 4, 0, 0, 0]$, which is represented as a complex vector with zero imaginary part:

$$\boldsymbol{x} = [(0,0),(0,0),(2,0),(3,0),(4,0),(0,0),(0,0),(0,0)]^T \tag{4.144}$$

The element in the mth row and nth column of the 8 by 8 DFT matrix is

$$w[m,n] = \frac{1}{\sqrt{N}} e^{-j2\pi mn/N} = \frac{1}{\sqrt{8}} (e^{-j\pi/4})^{mn} = \frac{1}{\sqrt{8}} [\cos(\frac{\pi mn}{4}) - j\ \sin(\frac{\pi mn}{4})] \tag{4.145}$$

The real and imaginary parts of the 8-point DFT matrix $\boldsymbol{W}$ are given in Eqs. 4.142 and 4.143. Carrying out the DFT matrix multiplication:

$$\boldsymbol{X} = \overline{\boldsymbol{W}}\boldsymbol{x} \tag{4.146}$$

we get the $N = 8$ DFT coefficients $\boldsymbol{X} = \boldsymbol{X}_r + j\boldsymbol{X}_j$:

$$\begin{aligned} \boldsymbol{X}_r &= \big[3.18, -2.16, 0.71, -0.66, 1.06, -0.66, 0.71, -2.16\big]^T \\ \boldsymbol{X}_j &= \big[0.0, -1.46, 1.06, -0.04, 0.0, 0.04, -1.06, 1.46\big]^T \end{aligned} \tag{4.147}$$

The real and imaginary parts of these complex coefficients are shown in Fig.4.12. As the time signal is real, its DFT is symmetric. The real part 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: $X_j[1] = -X_j[7]$, $X_j[2] = -X_j[6]$, $X_j[3] = -X_j[5]$. However, note that $X_r[0] \neq X_r[4]$, and $X_j[0] = X_j[4] = 0$.

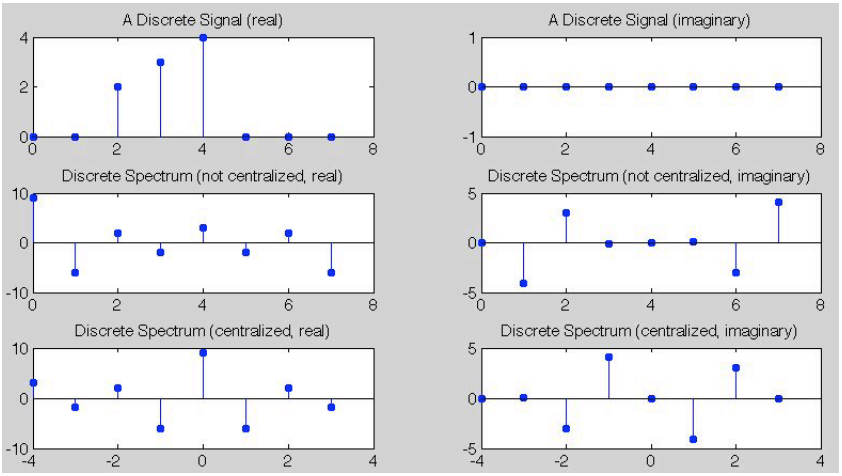

Figure 4.12 A discrete signal $x[m]$ and its DFT spectrum $X[n]$

The signal $x[m]$ can also be reconstructed by the inverse DFT from its DFT coefficients $X[n]$:

$$\boldsymbol{x} = \begin{bmatrix} x[0] \\ \vdots \\ x[7] \end{bmatrix} = \boldsymbol{W}\boldsymbol{X} = [\boldsymbol{w}_0, \cdots, \boldsymbol{w}_7] \begin{bmatrix} X[0] \\ \vdots \\ X[7] \end{bmatrix} = \sum_{n=0}^{7} X[n]\boldsymbol{w}_n \tag{4.148}$$

Here the signal $\boldsymbol{x}$ is expressed as a linear combination of the column vectors of the DFT matrix $\boldsymbol{W}$, which, as a set of 8 orthonormal basis vectors, span an 8-D vector space.

Consider specifically what these 8 complex values $X[n] = X_r[n] + jX_j[n]$ ($n = 0, \cdots, 7$) represent:

- $X_r[0]$ is proportional to the sum of all signal samples $x[m]$, therefore it represents the average of the signal. This is a real value as $X_j[0] = 0$.
- The three pairs of terms corresponding to $n = 1, 7$, $n = 2, 6$ and $n = 3, 5$ represent respectively three sinusoids of frequency $f_n = n/N$ or $\omega_n = 2\pi f_n = 2n\pi/N$, with amplitude $|X[n]| = \sqrt{X_r^2[n] + X_j^2[n]}$ and phase $\angle X[n] = \tan^{-1}(X_j[n]/X_r[n])$:
  - $n = 1, 7$: $f_1 = 1/8$, $\omega_1 = 0.79$, $|X[1]| = 2.61/8$, $\angle X[1] = -2.55$ rad/sec.
  - $n = 2, 6$: $f_2 = 2/8$, $\omega_2 = 1.57$, $|X[2]| = 1.28/8$, $\angle X[2] = 0.98$ rad/sec.
  - $n = 3, 5$: $f_3 = 3/8$, $\omega_3 = 2.36$, $|X[3]| = 0.67/8$, $\angle X[3] = -3.08$ rad/sec.
- $X_r[4] = 3/8$ is the amplitude of the highest frequency component with $f_4 = 4/8$ or $\omega_4 = 3.14$. As $X_j[4] = 0$, the phase shift is zero.

Now the signal can be expanded as (Eq. 4.124):

$$\begin{aligned} x[m] &= \frac{1}{\sqrt{N}}[X[0] + 2\sum_{n=1}^{3}|X[n]|\cos(\frac{2\pi mn}{N} + \angle X[n]) + X[4]\cos(m\pi)] \\ &= \frac{1}{\sqrt{8}}[3.18 + 2(2.61\cos(0.79m - 2.55) + 1.28\cos(1.57m + 0.98) \\ &+ 0.67\cos(2.36m - 3.08)) + 1.06\cos(3.14m)], \quad (m = 0, \cdots, 7) \end{aligned} \tag{4.149}$$

To illustrate the reconstruction of this 8-point discrete signal, we consider it as the discrete version of the corresponding Fourier series expansion of a continuous signal, which can be reconstructed as a linear combination of the its frequency components with progressively more frequency components with higher frequencies, as shown In Fig. 4.13.

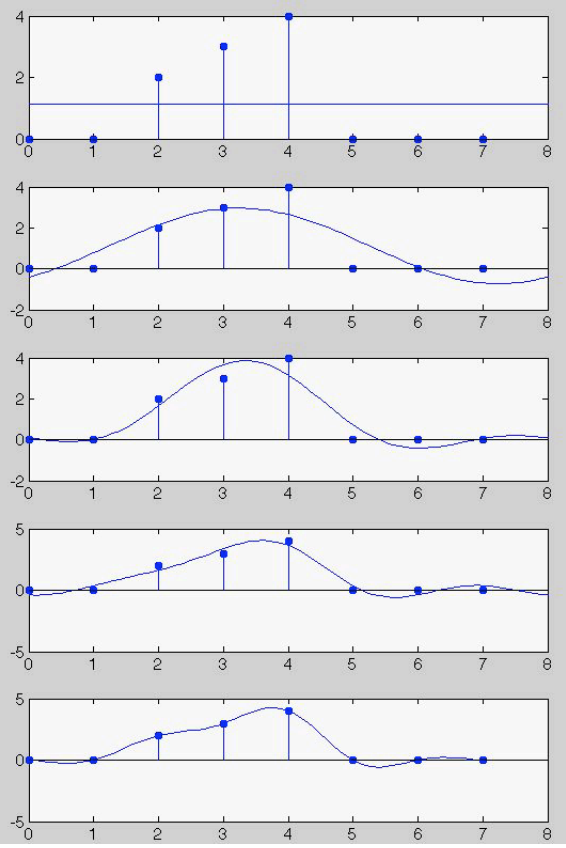

Figure 4.13 Reconstruction of the signal
From the top down, the number of components included in the reconstruction is increased with progressively higher frequencies from $f_0 = 0$ (DC component) to f_4 (highest frequency). The perfect reconstruction is obtained when all frequency components are included.

In the discrete spectrum shown in Eq. 4.147, the DC is on the left ($n = 0$) while the highest frequency component is in the middle ($n = N/2 = 4$). One may want to shift the spectrum by half of its length $N/2$ so that the DC component in the middle while the higher frequencies are farther away from it on both sides. This can also be easily realized in time domain due to the frequency-shift property (Eq. 4.153):

$$\mathcal{F}^{-1}[X[n - N/2]] = x[m]e^{j2\pi mN/2/N} = x[m]e^{m\pi} = x[m](-1)^m \tag{4.150}$$

i.e., if we negate every other sample in time domain, its spectrum is shifted by half of the length to become:

$$\begin{aligned} \boldsymbol{X}_r &= \big[1.06, -0.66, 0.71, -2.61, 3.18, -2.16, 0.71, -0.66\big]^T \\ \boldsymbol{X}_j &= \big[0.0, 0.04, -1.06, 1.46, 0.0, -1.46, 1.06, -0.04,\big]^T \end{aligned} \tag{4.151}$$

Once the signal is decomposed by the DFT into different frequency components in frequency domain, various filtering processing can be carried out as needed for the specific application, for example, low, band and high-pass (or stop), by manipulating the coefficients for different frequency components.

4.2.5 Properties of DFT

As a special case of the Fourier transform, the DFT shares all the properties of the Fourier transform discussed previously, although they are in different forms. We consider only a few of the properties here.

- **Time and frequency shifting**

$$\mathcal{F}[x[m \pm m_0]] = X[n]e^{\pm j2\pi m_0 n/N} \tag{4.152}$$

and

$$\mathcal{F}[x[m]e^{\mp j2\pi mn_0/N}] = X[n \pm n_0] \tag{4.153}$$

These results can be most straightforwardly proven from the definition.

- **Symmetry**

The DFT is complex transform which can be separated into real and imaginary parts:

$$\begin{aligned} X[n] &= \sum_{m=0}^{N-1} x[m]e^{-j2\pi mn/N} = \sum_{m=0}^{N-1} (x_r[m] + jx_i[m])[\cos(\frac{2\pi mn}{N}) - j\ \sin(\frac{2\pi mn}{N})] \\ &= X_r[n] + jX_j[n] \end{aligned} \tag{4.154}$$

where

$$X_r[n] = \sum_{m=0}^{N-1} x_r[m]\cos(\frac{2\pi mn}{N}) + \sum_{m=0}^{N-1} x_j[m]\sin(\frac{2\pi mn}{N})$$
$$X_j[n] = \sum_{m=0}^{N-1} x_j[m]\cos(\frac{2\pi mn}{N}) - \sum_{m=0}^{N-1} x_r[m]\sin(\frac{2\pi mn}{N}) \quad (4.155)$$

In particular, if $x[m] = x_r[m]$ is real ($x_j[m] = 0$), then $X_r[n]$ is even

$$X_r[n] = \sum_{m=0}^{N-1} x_r[m]\cos(\frac{2\pi mn}{N}) = X_r[-n] \quad (4.156)$$

and $X_j[n]$ is odd

$$X_j[n] = -\sum_{m=0}^{N-1} x_r[m]\sin(\frac{2\pi mn}{N}) = -X_j[-n] \quad (4.157)$$

Specially, $X_r[0]$ represents the DC offset of the signal (zero frequency):

$$X_r[0] = \sum_{m=0}^{N-1} x_r[m]\cos(\frac{2\pi m0}{N}) = \sum_{m=0}^{N-1} x_r[m] \quad (4.158)$$

and $X_r[N/2]$ represents the highest frequency component:

$$X_r[N/2] = \sum_{m=0}^{N-1} x_r[m]\cos(\frac{2\pi mN/2}{N}) = \sum_{m=0}^{N-1} x_r[m](-1)^m \quad (4.159)$$

When $n = 0$ and $n = N/2$, the imaginary parts $X_j[0] = X_j[N/2] = 0$ are zero because $\sin(0) = \sin(m\pi) = 0$.

- **Convolution theorem**
  The convolution of two discrete and periodic signals $x[m + N] = x[m]$ and $h[m + N] = h[m]$ $(m = 0, \cdots, N - 1)$ is defined as

$$y[m] = h[m] * x[m] = \sum_{n=0}^{N-1} h[m - n]x[n], \quad (m = 0, \cdots, N - 1) \quad (4.160)$$

  As both $x[m]$ and $h[m]$ are assumed to be periodic with period N, it is obvious that the result $y[m]$ of the convolution is also periodic: $y[m + N] = y[m]$. The convolution is therefore also referred to as a *circular convolution*.
  The convolution theorem states:

$$\mathcal{F}[h[m] * x[m]] = H[n]X[n] \quad (a) \quad (4.161)$$

$$\mathcal{F}[h[m]x[m]] = H[n] * X[n] \quad (b) \quad (4.162)$$

Proof of (a):

$$\mathcal{F}[x[m] * h[m]] = \sum_{m=0}^{N-1}[\sum_{k=0}^{N-1} x[k]h[m-k]]e^{-j2\pi mn/N}$$
$$= \sum_{k=0}^{N-1} x[k] \sum_{m=0}^{N-1} h[m-k]e^{-j2\pi(m-k)n/N}e^{-j2\pi kn/N}$$
$$= \sum_{k=0}^{N-1} x[k]H[n]e^{-j2\pi kn/N} = X[n]H[n] \tag{4.163}$$

Note that due to periodicity of the signal $y[m-k]$, the second summation is still for the same N samples over one period, and therefore is the Fourier transform of signal y. The proof of (b) is very similar to the above.

- **Diagonalization of circulant matrix**

  Based on one of the vectors $h[n]$ of the convolution above, an N by N matrix $\boldsymbol{H}$ can be constructed with its element in the mth row and nth column defined as $h[m,n] = h[m-n]$, so that the circular convolution in Eq.4.160 can be expressed as a matrix multiplication:

$$\boldsymbol{y} = \begin{bmatrix} y[0] \\ y[1] \\ \vdots \\ y[N-2] \\ y[N-1] \end{bmatrix} = \boldsymbol{H}\boldsymbol{x} = \begin{bmatrix} h[0] & h[N-1] & \cdots & h[2] & h[1] \\ h[1] & h[0] & \cdots & h[3] & h[2] \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ h[N-2] & h[N-3] & \cdots & h[0] & h[N-1] \\ h[N-1] & h[N-2] & \cdots & h[1] & h[0] \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ \vdots \\ x[N-2] \\ x[N-1] \end{bmatrix} \tag{4.164}$$

  Such a matrix $\boldsymbol{H}$ is called a *circulant matrix* where each row is rotated one element to the right relative to the previous row.

  We now show that the nth DFT coefficient $H[n]$ of $h[m]$ and the nth column vector $\boldsymbol{w}_n$ of the DFT matrix $\boldsymbol{W}$ are the eigenvalue and eigenvector of the matrix $\boldsymbol{H}$, respectively:

$$\boldsymbol{H}\boldsymbol{w}_n = H[n]\boldsymbol{w}_n, \qquad (n = 0, \cdots, N-1) \tag{4.165}$$

  where $\boldsymbol{w}_n = [w^{j2\pi 0n/N}, \cdots, w^{j2\pi(N-1)n/N}]^T$ is the Nth column vector of $\boldsymbol{W}$ and $H[n]$ is the nth DFT coefficient:

$$H[n] = \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} h[m]e^{-j2\pi mn/N} \tag{4.166}$$

  Consider the lth element of the left hand side of Eq.4.165:

$$\sum_{k=0}^{N-1} h[l,k]w^{j2\pi kn/N} = \sum_{k=0}^{N-1} h[l-k]w^{j2\pi kn/N} \tag{4.167}$$

We let $l-k=m'$, i.e., $k=l-m'$, and the above becomes:

$$\sum_{m'=0}^{N-1} h[m']w^{-j2\pi m'n/N}w^{j2\pi nl/N} = H[n]w^{j2\pi nl/N} \tag{4.168}$$

This turns out to be the lth element of the right hand side of Eq.4.165. We can now see that the circulant matrix $\boldsymbol{H}$ can be diagonalized by the DFT matrix $\boldsymbol{W}=[\boldsymbol{w}_0,\cdots,\boldsymbol{w}_{N-1}]$:

$$\boldsymbol{HW}=\boldsymbol{WD}, \qquad \text{i.e.} \qquad \boldsymbol{W}^{-1}\boldsymbol{HW}=\overline{\boldsymbol{W}}\boldsymbol{HW}=\boldsymbol{D} \tag{4.169}$$

where $\boldsymbol{D}$ is a diagonal matrix composed of all N DFT coefficients along the main diagonal:

$$\boldsymbol{D}=diag(H[0],\cdots,H[N-1]) \tag{4.170}$$

Taking the DFT on both sides of $\boldsymbol{y}=\boldsymbol{Hx}$ in Eq.4.164 by pre-multiplying $\overline{\boldsymbol{W}}$, we get:

$$\overline{\boldsymbol{W}}\boldsymbol{y}=\boldsymbol{Y}=\overline{\boldsymbol{W}}\boldsymbol{Hx}=\overline{\boldsymbol{W}}\boldsymbol{HW}\,\overline{\boldsymbol{W}}\boldsymbol{x}=\boldsymbol{DX} \tag{4.171}$$

i.e.,

$$\begin{bmatrix} Y[0] \\ Y[1] \\ \vdots \\ Y[N-1] \end{bmatrix} = \begin{bmatrix} H[0] & 0 & \cdots & 0 \\ 0 & H[1] & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & H[N-1] \end{bmatrix} \begin{bmatrix} X[0] \\ X[1] \\ \vdots \\ X[N-1] \end{bmatrix} \tag{4.172}$$

The nth element of this vector equation is:

$$Y[n]=H[n]X[n] \tag{4.173}$$

This is of course consistent with the conclusion of the discrete convolution theorem.

Example 4.9:

$$\boldsymbol{x}=[1\ 2\ 3\ 4\ 5\ 6\ 7\ 8]^T, \qquad \boldsymbol{h}=[1\ 2\ 3\ 0\ 0\ 0\ 0\ 0]^T$$

We want to find their convolution:

$$y[m]=h[m]*x[m]=\sum_n h[m-n]x[n], \qquad (m=0,\cdots,N-1)$$

Different from the convolution in Example 4.3, here $y[m]=h[m]*x[m]$ is a *circular convolution* as both $\boldsymbol{x}$ and $\boldsymbol{h}$ are assumed to be periodic with period $N=8$,

and, consequently, their convolution is also periodic, as shown below:

n	$\cdots$	-2	-1	0	1	2	3	4	5	6	7	8	9	$\cdots$
$x[n]$	$\cdots$	7	8	1	2	3	4	5	6	7	8	1	2	$\cdots$
$h[0-n]$	$\cdots$	3	2	1										$\cdots$
$h[1-n]$	$\cdots$		3	2	1									$\cdots$
$h[2-n]$	$\cdots$			3	2	1								$\cdots$
$h[3-n]$	$\cdots$				3	2	1							$\cdots$
$h[4-n]$	$\cdots$					3	2	1						$\cdots$
$h[5-n]$	$\cdots$						3	2	1					$\cdots$
$h[6-n]$	$\cdots$							3	2	1				$\cdots$
$h[7-n]$	$\cdots$								3	2	1			$\cdots$
$h[8-n]$	$\cdots$									3	2	1		$\cdots$
$h[9-n]$	$\cdots$										3	2	1	$\cdots$
$h[10-n]$	$\cdots$											3	2	$\cdots$
$y[m]$	$\cdots$	34	40	38	28	10	16	22	28	34	40	38	28	$\cdots$

For example, when $m = 2$, we have:

$$y[2] = \sum_n h[2-n]x[n] = h[2]x[0] + h[1]x[1] + h[0]x[2] = 3 \times 1 + 2 \times 2 + 1 \times 3 = 10$$

We see that the resulting $y[m+8] = y[m]$ is indeed periodic.

Next, we show that this discrete convolution can also be carried out by DFT. We find the 8-point DFTs $\boldsymbol{X} = DFT[\boldsymbol{x}]$ and $\boldsymbol{H} = DFT[\boldsymbol{h}]$ and also their element-wise product $\boldsymbol{Y} = [Y[0], \cdots, Y[7]]^T$, where $Y[n] = H[n]X[n]$ ($n = 0, \cdots, 7$):

$$\boldsymbol{X} = \begin{bmatrix} 36 \\ -4+9.657j \\ -4+4j \\ -4+1.657j \\ -4 \\ -4-1.657j \\ -4-4j \\ -4-9.657j \end{bmatrix}, \quad \boldsymbol{H} = \begin{bmatrix} 6 \\ 2.414-4.414j \\ -2-2j \\ -0.414+1.586j \\ 2 \\ -0.414-1.586j \\ -2+2j \\ 2.414+4.414j \end{bmatrix}, \quad \boldsymbol{Y} = \begin{bmatrix} 216 \\ 32.971+40.971j \\ 16 \\ -0.971-7.029j \\ -8 \\ -0.971+7.029j \\ 16 \\ 32.971-40.971j \end{bmatrix}$$

The convolution $y[m] = h[m] * x[m]$ can be obtained by inverse DFT to be:

$$\boldsymbol{y} = DFT^{-1}[\boldsymbol{Y}] = [38\ 28\ 10\ 16\ 22\ 28\ 34\ 40]^T$$

4.2.6 DFT Computation and Fast Fourier Transform

The Fourier transform of a signal can be carried out on a digital computer only if the signal is (a) discrete and (b) finite, i.e., out of all different forms of the Fourier transform, only DFT can actually be carried out.

Listed below is the C code for both the forward and inverse DFT based on the simple array multiplications in Eq.4.134. This function dft takes a complex data vector for the time signal as the input and returns its complex DFT coefficients. This is an in-place algorithm, i.e., the input vector $xr[m] + j\,xi[m]$ $(m = 0, \cdots, N-1)$ 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 argument inv=0, or inverse DFT when inv=1.

```
void dft(xr,xi,N,inv)
     float *xr, *xi;      // real and imaginary parts of data
     int N;               // size of data
     int inverse;         // inv=0 for forward DFT, inv=1 for 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 (n=0; n<N; n++) {   // for all N frequency components
    yr[n]=yi[n]=0;
    for (m=0; m<N; m++) { // for all N data samples
      arg=2*Pi*m*n/N;
      if (!inv) arg=-arg; // minus sign needed for forward DFT
      c=cos(arg); s=sin(arg);
      yr[n]+=xr[m]*c-xi[m]*s;
      yi[n]+=xi[m]*c+xr[m]*s;
    }
  }
  arg=1.0/sqrt((float)N);
  for (n=0; n<N; n++)
     { xr[n]=arg*yr[n]; xi[n]=arg*yi[n]; }
  free(yr); free(yi);
}
```

The computational complexity of this algorithm is $O(N^2)$, due obviously to the two nested for loops each of size N, i.e., it takes $O(N)$ operations to obtain each of the N coefficients $X[n]$. If the signal contains $N = 10^3 = 1000$ samples, the computational complexity is $O(N^2) = O(10^6)$. Due to such a high computational complexity, the actual application of the DFT is quite limited in practice.

To speed up the computation, a revolutionary *fast Fourier transform (FFT)* algorithm was developed in 1960's that can reduce the complexity of a DFT from $O(N^2)$ to $O(N \log_2 N)$. 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 $\boldsymbol{W}$. We first define

$$w_N = e^{-j2\pi/N} \tag{4.174}$$

and note the following properties of w_N:

1.

$$w_N^{kN} = e^{-j2k\pi N/N} = e^{-j2k\pi} = 1 \tag{4.175}$$

2.

$$w_{2N}^{2k} = e^{-j2k2\pi/2N} = e^{-jk2\pi/N} = w_N^k \tag{4.176}$$

3.

$$w_{2N}^{N} = e^{-j2N\pi/2N} = e^{-j\pi} = -1 \tag{4.177}$$

Let $N = 2M$, an N-point DFT can be written as

$$\begin{aligned} X[n] &= \sum_{m=0}^{N-1} x[m] e^{j2\pi mn/N} = \sum_{m=0}^{N-1} x[m] w_N^{mn} \\ &= \sum_{m=0}^{M-1} x[2m] w_{2M}^{2mn} + \sum_{m=0}^{M-1} x[2m+1] w_{2M}^{(2m+1)n} \end{aligned} \tag{4.178}$$

This is only for the first half of the coefficients $X[n]$ for $n = 0, \cdots, M-1$. The second half will be considered later. The first summation in this expression includes all the even terms and the second all the odd ones. Due to the 2nd property of w_M, the above can be rewritten as

$$X[n] = \sum_{m=0}^{M-1} x[2m] w_M^{mn} + \sum_{m=0}^{M-1} x[2m+1] w_M^{mn} w_{2M}^{n} = X_{even}[n] + X_{odd}[n] w_{2M}^{n} \tag{4.179}$$

where we have defined:

$$X_{even}[n] = \sum_{m=0}^{N-1} x[2m] w_M^{mn}, \quad \text{and} \quad X_{odd}[n] = \sum_{m=0}^{N-1} x[2m+1] w_M^{mn} \tag{4.180}$$

which are two $N/2$-point DFTs. The coefficients $X[n]$ in the second half can be obtained by replacing n in Eq. 4.179 by $n + N$:

$$X[n+M] = X_{even}[n+M] + X_{odd}[n+M] w_{2M}^{n+M} \tag{4.181}$$

Due to the first property of w_M, we have

$$X_{even}[n+M] = \sum_{m=0}^{M-1} x[2m] w_M^{m(n+M)} = \sum_{m=0}^{M-1} x[2m] w_M^{mn} = X_{even}[n] \tag{4.182}$$

and similarly we have

$$X_{odd}[n+M] = X_{odd}[n] \tag{4.183}$$

Also, due to the 3rd property of w_M, we have

$$w_{2M}^{n+M} = w_{2M}^{n} w_{2M}^{M} = -w_{2M}^{n} \tag{4.184}$$

Now the coefficients in the second half of the DFT can be found as

$$X[n+M] = X_{even}[n] - X_{odd}[n] w_{2M}^{n} \tag{4.185}$$

The N-point DFT can now be obtained from Eqs. 4.179 and 4.185 with complexity of $O(N)$, once $X_{ever}[n]$ and $X_{odd}[n]$ are available. But $X_{even}[n]$ and $X_{odd}[n]$ are themselves two N/2-point DFTs, which can be obtained exactly the same way. Now we can see that an N-point DFT can be carried out recursively in $\log_2 N$ levels, each time the DFT size is reduced by half, until eventually the size becomes 1, the DFT coefficient is the same as the signal sample. As the complexity at each level is $O(N)$, the total complexity for the N-point DFT is $O(N\ log_2 N)$, as symbolically illustrated by the diagram in Fig. 4.14 For the same example of a signal containing 1000 samples, the number of operations needed by the FFT algorithm is in the order of 10^4 ($1000 \times log_2 1000 \approx 1000 \times 10$), instead of 10^6 without using FFT.

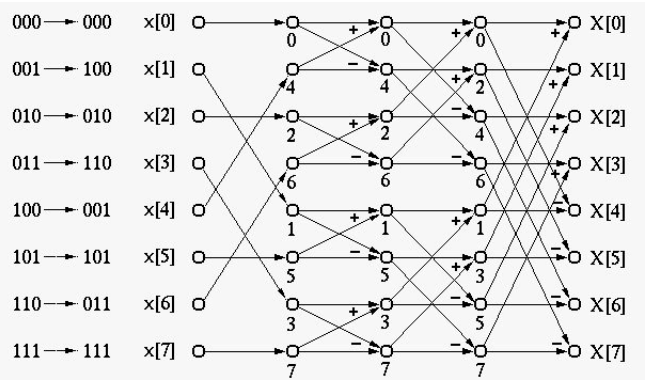

Figure 4.14 The fast Fourier transform algorithm

The C code for the FFT algorithm is given below. The function fft takes a complex data vector for the time signal as the input and returns its complex DFT coefficients. 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 $xr[m] + j\,xi[m]$ $(m = 0, \cdots, N-1)$ for the time signal will be overwritten by the output, its DFT coefficients. This 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 inv=0, or inverse DFT when 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,m,n;
  float arg,s,c,w,tmpr,tmpi;
  m=log2f((float)N);
  for (i=0; i<N; ++i) {      // for all N elements of data
    j=0;
    for (k=0; k<m; ++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<m; i++) {      // for log2(N) stages
    n=pow(2.0,(float)i);     // length of section in current stage
    w=Pi/n;
    if (!inv) w=-w;          // include minus sign needed for forward FFT
    k=0;
    while (k<N-1) {          // for N elements in a stage
      for (j=0; j<n; j++) {  // for all points in each section
        arg=j*w; c=cos(arg); s=sin(arg);
        l=k+j;
        tmpr=xr[l+n]*c-xi[l+n]*s;
        tmpi=xi[l+n]*c+xr[l+n]*s;
        xr[l+n]=xr[l]-tmpr;
        xi[l+n]=xi[l]-tmpi;
        xr[l]=xr[l]+tmpr;
        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 DFT computation can be further cut in half if multiple real signals need to be transformed. In general, the Fourier transform is an operator in a unitary space in which all vectors are complex. On the other hand, all physical signals are real, i.e., the imaginary part of a signal vector need to be filled with zeros,

subsequently wasting half of the computation. This waste of time can be avoided if more than one real signal vector need to be transformed, by the following method based on the symmetry properties of the Fourier transform (table 3.1).

Recall that if $x(t)$ is real, the real part of its spectrum is even $X_r(f) = X_r(-f)$, and the imaginary part is odd $X_j(f) = -X_j(-f)$. But if $x(t)$ is imaginary, the real part of its spectrum is odd $X_r(f) = -X_r(-f)$, and the imaginary part is odd $X_j(f) = -X_j(-f)$.

We also note that an arbitrary function $f(x)$ can be decomposed into the even and odd components $f_e(x)$ and $f_o(x)$:

$$\begin{cases} f_e(x) = [f(x) + f(-x)]/2 = f_e(-x) \\ f_o(x) = [f(x) - f(-x)]/2 = -f_o(-x) \end{cases} \tag{4.186}$$

This result can be verified:

$$f_e(x) + f_o(x) = f(x) \tag{4.187}$$

Two real signals $x_1[m]$ and $x_2[m]$ $(m = 0, \cdots, N-1)$ can be transformed simultaneously in the following steps:

1. Construct a complex vector composed of $x_1[m]$ as its real part and $x_2[m]$ as its imaginary part:

$$x[m] = x_1[m] + jx_2[m], \qquad (m = 0, \cdots, N-1) \tag{4.188}$$

2. Obtain the DFT of $x[m]$:

$$X[n] = X_r[n] + jX_j[n], \qquad (n = 0, \cdots, N-1) \tag{4.189}$$

3. Obtain $\mathcal{F}[x_1] = X_1 = X_{1r} + jX_{1j}$.
   As x_1 is real, the real part of its spectrum X_{1r} is even and the imaginary part X_{1j} is odd, i.e.,

$$X_1[n] = X_{1r}[n] + jX_{1j}[n] = \frac{X_r[n] + X_r[-n]}{2} + j\frac{X_j[n] - X_j[-n]}{2} \tag{4.190}$$

4. Obtain $\mathcal{F}[x_2] = X_2 = X_{2r} + jX_{2j}$.
   As jx_2 is imaginary, the real part of its spectrum jX_{2r} is odd and the imaginary part jX_{2j} is even, i.e.,

$$jX_2[n] = jX_{2r}[n] + j(jX_{2j}[n]) = \frac{X_r[n] - X_r[-n]}{2} + j\frac{X_j[n] + X_j[-n]}{2} \tag{4.191}$$

   Dividing both sides by j, we get the spectrum X_2 of real signal x_2:

$$X_2[n] = X_{2r}[n] + jX_{2j}[n] = \frac{X_j[n] + X_j[-n]}{2} - j\frac{X_r[n] - X_r[-n]}{2} \tag{4.192}$$

4.3 Two-Dimensional Fourier Transform

4.3.1 Four Forms of 2-D Fourier Transform

All signals considered so far are assumed to be one-dimensional time functions. However, a signal could also be a function over a 1D space, with the spatial frequency defined as the number of cycles in unit length, instead of in unit time. Moreover, the concept of frequency analysis can be extended to various signals in two or three-dimensional spaces. In particular, images are also a typical 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 1D situation, the Fourier transform is also a powerful tool in signals processing and analysis in two or higher dimensional space.

Same as in the 1D 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**

$$F(u, v) = \int\int_{-\infty}^{\infty} f(x, y) e^{-j2\pi(ux+vy)} dx\, dy \tag{4.193}$$

$$f(x, y) = \int\int_{-\infty}^{\infty} F(u, v) e^{j2\pi(ux+vy)} du\, dv \tag{4.194}$$

  where u and v represent two spatial frequencies along the directions of x and y in the 2-D space, respectively. In the inverse transform the 2-D signal $f(x, y)$ is represented by a linear combination (the double integral) of infinite number of uncountable 2-D orthogonal basis functions $\phi_{u,v}(x, y) = e^{j2\pi(ux+vy)}$, where u and v are the frequencies in directions x and y, respectively. Each basis function $\phi_{u,v}(x, y)$ is weighted by the Fourier coefficient function $F(u, v)$, the 2-D spectrum of the signal, which is obtained in the forward transform as the projection of the signal $f(x, y)$ onto each of the basis functions $\phi_{u,v}(x, y)$.
- **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.

$$F_{UV}(u, v) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} f[m, n] e^{-j2\pi(umx_o+vny_o)} \tag{4.195}$$

$$f[m, n] = \frac{1}{UV} \int_0^U \int_0^V F(u, v) e^{j2\pi(umx_o+vny_o)} du\, dv \tag{4.196}$$

  The 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.

$$F[k,l] = \frac{1}{XY}\int_0^X \int_0^Y f_{XY}(x,y) e^{j2\pi(kxu_o + lyv_o)} dx\, dy \tag{4.197}$$

$$f_{XY}(x,y) = \sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} F[k,l] e^{-j2\pi(xku_o + ylv_o)} \tag{4.198}$$

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 $x[m,n]$ ($m = 0, \cdots, M-1, n = 0, \cdots, M-1$). The 2-D DFT pair is

$$X[k,l] = \frac{1}{\sqrt{MN}}\sum_{n=0}^{N-1}\sum_{m=0}^{M-1} x[m,n] e^{-j2\pi(\frac{mk}{M}+\frac{nl}{N})}$$

$$x[m,n] = \frac{1}{\sqrt{MN}}\sum_{l=0}^{N-1}\sum_{k=0}^{M-1} X[k,l] e^{j2\pi(\frac{mk}{M}+\frac{nl}{N})}$$

$$(0 \le m,k \le M-1, \quad 0 \le n,l \le N-1) \tag{4.199}$$

  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 it its DFT $X[k+M, l+N] = X[k,l]$.

Note that the 2-D kernel function of the 2-D 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. For example, the continuous kernel function can be written as:

$$\phi_{u,v}(x,y) = e^{j2\pi(ux+vy)} = e^{j2\pi ux} e^{j2\pi vy} = \phi_u(x)\phi_v(y) \tag{4.200}$$

Therefore the 2-D transform can be carried out as:

$$\begin{aligned} F(u,v) &= \int\int_{-\infty}^{\infty} f(x,y) e^{-j2\pi ux} e^{-j2\pi vy} dx\, dy \\ &= \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} f(x,y) e^{-j2\pi ux}\, dx\right] e^{-j2\pi vy}\, dy \\ &= \int_{-\infty}^{\infty} F'(u,y) e^{-j2\pi vy}\, dy \end{aligned} \tag{4.201}$$

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.

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 actually carried out. As in the case of 1D Fourier transform, the scaling factor $1/MN$ is of little significance.

4.3.2 Computation of 2-D DFT

A 2-D discrete signal $x[m, n]$ can be considered as an M by N matrix consisting of N M-dimensional column vectors, or M N-dimensional row vectors. And, as discussed before, as the kernel function is separable, the 2-D DFT for this signal can be carried out in two steps: N 1D DFTs of the N columns, followed by M 1D DFTs of the M rows of the matrix obtained in step 1. The order of the two steps can be reversed. Specifically, we have

$$\begin{aligned} X[k, l] &= \frac{1}{\sqrt{MN}} \sum_{n=0}^{N-1} [\sum_{m=0}^{M-1} x[m, n] e^{-j2\pi \frac{mk}{M}}] e^{-j2\pi \frac{nl}{N}} \\ &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} X'[k, n] e^{-j2\pi \frac{nl}{N}} \quad (k = 0, \cdots, M-1, \;\; l = 0, \cdots, N-1) \end{aligned} \tag{4.202}$$

where X' is an intermediate result obtained by column transforms in the first step:

$$X'[k, n] = \frac{1}{\sqrt{M}} \sum_{m=0}^{M-1} x[m, n] e^{-j2\pi \frac{mk}{M}} \quad (n = 0, 1, \cdots, N-1) \tag{4.203}$$

The summation above is with respect to the row index m and the column index n can be treated as a parameter. This 1D DFT of the nth column of the 2-D signal matrix can be written in column vector (vertical) form as:

$$\boldsymbol{X'}_n = \overline{\boldsymbol{W}}_M \boldsymbol{x}_n, \quad (n = 0, \cdots, N-1) \tag{4.204}$$

where $\boldsymbol{X'}^T = [X[0, n], \cdots, X[N-1, n]]^T$ is an N-D vector and $\boldsymbol{W}_M$ is a $M \times M$ Fourier transform matrix with the mn-th element $w[m, n] = e^{j2\pi mn/M}/\sqrt{M}$, similar to the matrix in Eq. 4.129. Putting all these N columns together, we can

write

$$[\boldsymbol{X'}_0, \cdots, \boldsymbol{X'}_{N-1}] = \overline{\boldsymbol{W}}_M \, [\boldsymbol{x}_0, \cdots, \boldsymbol{x}_{N-1}] \tag{4.205}$$

We define two $M \times N$ matrices $\boldsymbol{X'}_{M\times N} = [\boldsymbol{X'}_0, \cdots, \boldsymbol{X'}_{N-1}]$ and $\boldsymbol{x}_{M\times N} = [\boldsymbol{x}_0, \cdots, \boldsymbol{x}_{N-1}]$, so that the above can be more concisely written as

$$\boldsymbol{X'}_{M\times N} = \overline{\boldsymbol{W}}_M \, \boldsymbol{x}_{M\times N} \tag{4.206}$$

In the second step, a 1D DFT is carried out for each of the M rows of the intermediate matrix $\boldsymbol{X'}$:

$$\boldsymbol{X}_k^T = (\overline{\boldsymbol{W}}_N \boldsymbol{X'}_k)^T = \boldsymbol{X'}_k^T \overline{\boldsymbol{W}}_N^T = \boldsymbol{X'}_k^T \overline{\boldsymbol{W}}_N, \quad (k = 0, \cdots, M-1) \tag{4.207}$$

where $\boldsymbol{X}_k^T$ is the kth row vector of matrix $\boldsymbol{X}$, and $\boldsymbol{W}_N$ is a $N \times N$ Fourier transform matrix with the mn-th element $e^{j2\pi mn/N}/\sqrt{N}$, which is symmetric $\overline{\boldsymbol{W}}_N^T = \overline{\boldsymbol{W}}_N$. Putting all these M rows together, we can write

$$\begin{bmatrix} \boldsymbol{X}_0^T \\ \vdots \\ \boldsymbol{X}_{M-1}^T \end{bmatrix} = \begin{bmatrix} \boldsymbol{X}_0^{'T} \\ \vdots \\ \boldsymbol{X}_{M-1}^{'T} \end{bmatrix} \overline{\boldsymbol{W}}_N \tag{4.208}$$

The equation above can be more concisely expressed as:

$$\boldsymbol{X}_{M\times N} = \boldsymbol{X'}_{M\times N} \overline{\boldsymbol{W}}_N = \overline{\boldsymbol{W}}_M \, \boldsymbol{x}_{M\times N} \overline{\boldsymbol{W}}_N \tag{4.209}$$

This is the 2-D DFT in matrix form.

Similarly, the inverse 2-D DFT can be written as

$$\boldsymbol{x}_{M\times N} = \boldsymbol{W}_M \, \boldsymbol{X}_{M\times N} \, \boldsymbol{W}_N \tag{4.210}$$

We rewrite these two equations as a 2-D DFT pair:

$$\begin{cases} \boldsymbol{X}_{M\times N} = \overline{\boldsymbol{W}}_M \, \boldsymbol{x}_{M\times N} \, \overline{\boldsymbol{W}}_N, & \text{(forward)} \\ \boldsymbol{x}_{M\times N} = \boldsymbol{W}_M \, \boldsymbol{X}_{M\times N} \, \boldsymbol{W}_N, & \text{(inverse)} \end{cases} \tag{4.211}$$

As before, the DFT matrix $\boldsymbol{W}$ can be expressed in terms of its column vectors (same as its row vectors as $\boldsymbol{W}^T = \boldsymbol{W}$), and the inverse transform can be written as:

$$\begin{aligned} \boldsymbol{x} &= [\boldsymbol{w}_0, \cdots, \boldsymbol{w}_{M-1}] \begin{bmatrix} X[0,0] & \cdots & X[0,N-1] \\ \vdots & \ddots & \vdots \\ X[M-1,0] & \cdots & X[M-1,N-1] \end{bmatrix} \begin{bmatrix} \boldsymbol{w}_0^T \\ \vdots \\ \boldsymbol{w}_{N-1}^T \end{bmatrix} \\ &= [\boldsymbol{w}_0, \cdots, \boldsymbol{w}_{M-1}] \begin{bmatrix} \sum_{l=0}^{N-1} X[0,l] \boldsymbol{w}_l^T \\ \vdots \\ \sum_{l=0}^{N-1} X[M-1,l] \boldsymbol{w}_l^T \end{bmatrix} \\ &= \sum_{k=0}^{M-1} \boldsymbol{w}_k \sum_{l=0}^{N-1} X[k,l] \boldsymbol{w}_l^T = \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} X[k,l] \boldsymbol{w}_k \boldsymbol{w}_l^T = \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} X[k,l] \boldsymbol{B}_{kl} \end{aligned} \tag{4.212}$$

Now the 2-D signal $\boldsymbol{x}$ is expressed as a linear combination of a set of MN 2-D (M by N) basis functions $\boldsymbol{B}_{kl} = \boldsymbol{w}_k \boldsymbol{w}_l^T$, each weighted by $X[k,l]$ ($k = 0, \cdots, M-1, l = 0. \cdots, N-1$), which can be obtained by the forward 2-D DFT. The equation above also shows that the kl-th 2-D DFT basis function $\boldsymbol{B}_{kl}$ can be found by this inverse 2-D DFT if all elements of the coefficient array are zero except $X[k,l] = 1$. When $M = N = 8$, the $M \times N = 64$ such 2-D basis functions are shown in Fig.4.15.

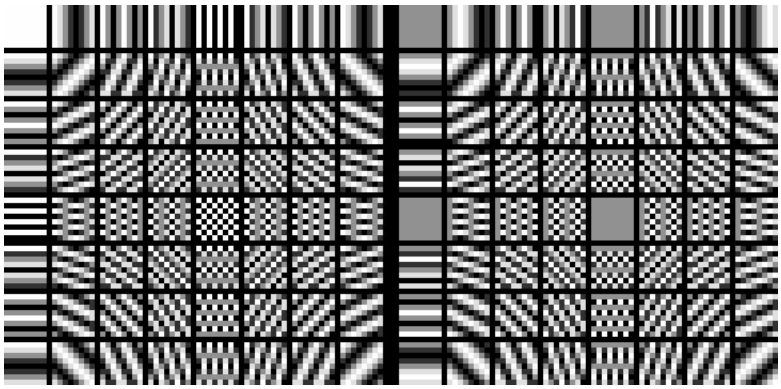

Figure 4.15 The $M \times N = 8 \times 8 = 64$ basis functions of the 2-D DFT
The left half of the image shows the real part of the 8 by 8 2-D basis functions, while the right half shows the corresponding imaginary parts. The DC component is at the top-left corner of the real part, and the highest frequency component in both horizontal and vertical directions is in the middle of the real part.

The coefficients $X[k,l]$ in the above expression for $\boldsymbol{x}$ can be obtained by the forward transform:

$$\boldsymbol{X} = \begin{bmatrix} \overline{\boldsymbol{w}}_0^T \\ \vdots \\ \overline{\boldsymbol{w}}_{M-1}^T \end{bmatrix} \boldsymbol{x}[\overline{\boldsymbol{w}}_0, \cdots, \overline{\boldsymbol{w}}_{N-1}] \tag{4.213}$$

and the kl-th coefficient is

$$\begin{aligned} X[k,l] &= \overline{\boldsymbol{w}}_k^T \begin{bmatrix} x[0,0] & \cdots & x[0,N-1] \\ \vdots & \ddots & \vdots \\ x[M-1,0] & \cdots & x[M-1,N-1] \end{bmatrix} \overline{\boldsymbol{w}}_l \\ &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m,n] \overline{B}_{kl}[m,n] = < \boldsymbol{x}, \boldsymbol{B}_{kl} > \end{aligned} \tag{4.214}$$

This is the inner product of two 2-D arrays $\boldsymbol{x}$ and $\boldsymbol{B}_{kl}$ (Eq.2.24), representing the projection of the signal $\boldsymbol{x}$ onto the kl-th basis function $\boldsymbol{B}_{kl}$.

The C code for both the forward and inverse 2-D DFT is listed below:

```
fft2d(xxr,xxi,m,n,inverse)   // 2D DFT
```

```
    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));
  printf("\nRow xform...\n");
  for (j=0; j<n; j++) {
    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]; }
    }
  printf("\nColumn xform...\n");
  for (i=0; i<m; i++) {
    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 4.10: Consider the 2-D DFT of a real 8×8 2-D signal (imaginary part is zero):

$$\boldsymbol{x}_r = \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 70.0 & 80.0 & 90.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 90.0 & 100.0 & 110.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 110.0 & 120.0 & 130.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 130.0 & 140.0 & 150.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix} \tag{4.215}$$

The 8-point DFT matrix $\boldsymbol{W}_8$ is the same as the one shown in Eqs. 4.142 and 4.143. After carrying out the 2-D DFT $\boldsymbol{X} = \overline{\boldsymbol{W}}_8 \, \boldsymbol{x} \, \overline{\boldsymbol{W}}_8$, we get the coefficient

matrix $\boldsymbol{X} = \boldsymbol{X}_r + j\boldsymbol{X}_j$:

$$\boldsymbol{X}_r = \left[\begin{array}{c|c|c|c|c|c|c|c} 165.0 & -98.9 & 10.0 & -21.1 & 55.0 & -21.1 & 10.0 & -98.9 \\ \hline -63.1 & -11.3 & 27.7 & 13.2 & -21.0 & 1.6 & -32.7 & 85.7 \\ 15.0 & 0.0 & -5.0 & -2.9 & 5.0 & 0.0 & 5.0 & 17.1 \\ -41.9 & 16.8 & 2.7 & 6.3 & -14.0 & 4.3 & -7.7 & 33.4 \\ \hline 15.0 & -8.5 & 0.0 & -1.5 & 5.0 & -1.5 & 0.0 & -8.5 \\ \hline -41.9 & 33.4 & -7.7 & 4.3 & -14.0 & 6.3 & 2.7 & 16.8 \\ 15.0 & -17.1 & 5.0 & 0.0 & 5.0 & -2.9 & -5.0 & 0.0 \\ -63.1 & 85.7 & -32.7 & 1.6 & -21.0 & 13.2 & 27.7 & -11.3 \end{array}\right] \tag{4.216}$$

and

$$\boldsymbol{X}_j = \left[\begin{array}{c|c|c|c|c|c|c|c} 0.0 & -88.9 & 55.0 & 11.1 & 0.0 & -11.1 & -55.0 & 88.9 \\ \hline -90.5 & 89.2 & -27.1 & 6.9 & -30.2 & 16.8 & 15.0 & 19.9 \\ 15.0 & -17.1 & 5.0 & 0.0 & 5.0 & -2.9 & -5.0 & 0.0 \\ -15.5 & 31.9 & -15.0 & -0.8 & -5.2 & 4.9 & 12.9 & -13.2 \\ \hline 0.0 & -8.5 & 5.0 & 1.5 & 0.0 & -1.5 & -5.0 & -8.5 \\ \hline 15.5 & 13.2 & -12.9 & -4.9 & 5.2 & 0.8 & 15.0 & -31.9 \\ -15.0 & 0.0 & 5.0 & 2.9 & -5.0 & 0.0 & -5.0 & 17.1 \\ 90.5 & -19.9 & -15.0 & -16.8 & 30.2 & -6.9 & 27.1 & -89.2 \end{array}\right] \tag{4.217}$$

As the signal $x[m,n]$ is real, the real part of its spectrum is 2-D 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 2-D odd: $X_j[k,l] = -X_j[M-k,N-l]$, $X_j[k,N-l] = -X_j[M-k,l]$, as can be observed. Also note that the real parts of four coefficients $X_r[0,0]$, $X_r[0,N/2]$, $X_r[M/2,0]$ and $X_r[M/2,N/2]$ are not paired with any other coefficients, and their imaginary parts are all zero $X_j[0,0] = X_j[0,N/2] = X_j[M/2,0] = X_j[M/2,N/2] = 0$.

Consider the 2-D sinusoids corresponding to each of the coefficients:

- $X[0,0] = X_r[0,0]$ is the amplitude of the DC offset (average) of the signal, $X_j[0,0] = 0$, the phase is zero;
- $X[M/2,N/2] = X_r[M/2,N/2]$ is the amplitude of the highest frequency component $(-1)^{m+n}$ contained in the signal, $X[M/2,N/2] = 0$, the phase is zero;
- $X[0,N/2] = X_r[0,N/2]$ is the amplitude of the highest frequency component in horizontal direction $(-1)^n$, $X_j[0,N/2] = 0$, the phase is zero. contained in the signal, $X[M/2,N/2] = 0$, the phase is zero;
- $X[M/2,0] = X_r[M/2,0]$ is the amplitude of the highest frequency component in vertical direction $(-1)^m$, $X_j[M/2,0] = 0$, the phase is zero.
- When $k = 0, l = 1, \cdots, N/2-1$, $X[0,l]$ pairs up with $X[0,N-l]$ 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;
- When $k = 1, \cdots, M/2-1, l = 0$, $X[k,0]$ pairs up with $X[M-k,0]$ 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.

Also note that as the signal is real, half of the data points carries no information. In the spatial domain, the imaginary part of the signal is all zero; while in the spatial frequency domain, both the real and imaginary parts are symmetric. (More specifically, the real part has $MN/2+2$ independent variables, the imaginary part has $MN/2-2$ independent variables. This fact indicates that half of the data is redundant and an algorithm can be designed based on the symmetry of the discrete spectrum to cut the computation of a 2-D DFT of a real signal by half.

From the previous example, we see that the high frequency components are around the center $(M/2, N/2)$ of the 2-D spectrum array, while the low frequency components are around the corners, such as the DC component is at the upper-left corner. Sometime it is preferable to centralize the spectrum so that the DC component and the low frequency components are in the middle of the spectrum array, and high frequency components are around the corners. This centralization of the DC component corresponding to shifting the 2-D discrete spectrum in both dimensions by half of the length, which can be realized by negating every other spatial samples, similar to 1D case discussed before:

$$\begin{aligned}&\mathcal{F}^{-1}[X[k-M/2,l-N/2]] = x[m,n]e^{j2\pi(\frac{mM/2}{M}+\frac{nN/2}{N})}\\ &= x[m][n]e^{j\pi(m+n)} = x[m](-1)^{m+n}\end{aligned} \tag{4.218}$$

If we negate the sign of any spatial sample $x[m,n]$ with $m+n$ being odd, i.e.

$$\begin{bmatrix} x[0,0] & -x[0,1] & x[0,2] & \cdots \\ -x[1,0] & x[1,1] & -x[1,2] & \cdots \\ x[2,0] & -x[2,1] & x[2,2] & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{4.219}$$

then the resulting 2-D Fourier spectrum will be centralized with DC component in the middle and high frequency components around the four edges. For the

example above, the centralized spectrum becomes

$$\boldsymbol{X}_r = \left[\begin{array}{rrrr|rrrr} 5.0 & -1.5 & 0.0 & -8.5 & 15.0 & -8.5 & 0.0 & -1.5 \\ \hline -14.0 & 6.3 & 2.7 & 16.8 & -41.9 & 33.4 & -7.7 & 4.3 \\ 5.0 & -2.9 & -5.0 & 0.0 & 5.0 & -17.1 & 5.0 & 0.0 \\ -21.0 & 13.2 & 27.7 & -11.3 & -63.1 & 85.7 & -32.7 & 1.6 \\ \hline 55.0 & -21.1 & 10.0 & -98.9 & 165.0 & -98.9 & 10.0 & -21.1 \\ \hline -21.0 & 1.6 & -32.7 & 85.7 & -63.1 & -11.3 & 27.7 & 13.2 \\ 5.0 & 0.0 & 5.0 & 17.1 & 15.0 & 0.0 & -5.0 & -2.9 \\ -14.0 & 4.3 & -7.7 & 33.4 & -41.9 & 16.8 & 2.7 & 6.3 \end{array}\right] \tag{4.220}$$

and

$$\boldsymbol{X}_j = \left[\begin{array}{rrrr|rrrr} 0.0 & -1.5 & -5.0 & -8.5 & 0.0 & -8.5 & 5.0 & 1.5 \\ \hline 5.2 & 0.8 & 15.0 & -31.9 & 15.5 & 13.2 & -12.9 & -4.9 \\ -5.0 & 0.0 & -5.0 & 17.1 & -15.0 & 0.0 & 5.0 & 2.9 \\ 30.2 & -6.9 & 27.1 & -89.2 & 90.5 & -19.9 & -15.0 & -16.8 \\ \hline 0.0 & -11.1 & -55.0 & 88.9 & 0.0 & -88.9 & 55.0 & 11.1 \\ \hline -30.2 & 16.8 & 15.0 & 19.9 & -90.5 & 89.2 & -27.1 & 6.9 \\ 5.0 & -2.9 & -5.0 & 0.0 & 15.0 & -17.1 & 5.0 & 0.0 \\ -5.2 & 4.9 & 12.9 & -13.2 & -15.5 & 31.9 & -15.0 & -0.8 \end{array}\right] \tag{4.221}$$

4.4 Fourier Filtering

4.4.1 1-D Filtering

A given time signal $x(t)$ or its spectrum $X(f) = \mathcal{F}[x(t)]$ can be filtered by a filter, which can be considered as an LTI system represented by the impulse response function $h(t)$ in time domain or the frequency response function $H(f) = \mathcal{F}[h(t)]$ in frequency domain. Correspondingly the filtering process can be represented as a convolution in time domain or a multiplication in frequency domain:

$$Y(f) = H(f)X(f), \quad \text{or} \quad y(t) = h(t) * x(t) \tag{4.222}$$

As will be seen in the examples below, the impulse response function $h(t)$ of a filter may not be causal, i.e., it may not satisfy the condition of causality $h(t) = 0$ for all $t < 0$. In other words, such a non-causal filter cannot be actually implementable in real time. However, this non-causality does not prevent the filter from being applied to off-line, recorded data in a wide variety of applications.

There exist many different types of filters, such as low-pass (LP), high-pass (HP), band-pass (BP), band-stop (BS). But to start with, let us first consider the low-pass filters of four different shapes and their filtering effects when applied to a specific signal, a square impulse train, shown in the top row of Fig.4.17.

- The *moving average LP-filter* is a filtering process in time domain that replaces each sample in the discrete signal by the average of a sequence of samples in

the neighborhood of the sample in question. Actually this operation of moving average is the convolution of the signal $x(t)$ with a window function $h(t)$. In frequency domain, the moving average filtering is a multiplication of the signal spectrum $X(f)$ and frequency response function $H(f)$, a sinc function (Eq. 3.141). This filter and its filtering effect are shown in the 2nd and 3rd rows of Fig.4.17.

- The *ideal LP-filter* is more conveniently defined in frequency domain as:

$$H(f) = \begin{cases} 1 & |f| < f_c \\ 0 & |f| > f_c \end{cases} \tag{4.223}$$

where f_c is the cut-off frequency. As shown in Eq.3.143, the impulse response of the ideal filter is a sinc function:

$$h(t) = \frac{\sin(2\pi f_c t)}{\pi t} = 2f_c \, sinc(2f_c t) \tag{4.224}$$

After filtering all frequency components outside the passing band are totally removed while those within the passing band are preserved. The ideal filter and its effect are shown respectively in the 4th and 5th rows of Fig.4.17. Note that the ideal LP-filter causes some strong ringing artifacts in the filtered signal, due to the convolution of the signal in time domain with the impulse response function of the filter, a sinc function $h(t) = \mathcal{F}^{-1}[H(f)]$.

- The nth-order *Butterworth LP-filter* is defined as:

$$H(f) = \frac{1}{\sqrt{1 + (f/f_c)^{2n}}} = \begin{cases} 1 & f = 0 \\ 1/\sqrt{2} & f = f_c \\ 0 & f = \infty \end{cases} \tag{4.225}$$

where f_c is the cut-off frequency at which $H(f) = H(f_c) = 0.5$, and n is a positive integer representing the order of the filter. By adjusting the order of the filter one can control the shape of the filter and thereby making a proper tradeoff between the rigging effects and how accurately the passing band can be specified. Specifically, as shown in Fig.4.16, when the order is low, the shape of the filter is smooth (low frequency accuracy) but with little ringing; when the order is high, the filter becomes sharper (higher frequency accuracy) but much stronger ringing effect will be caused. When $n \to \infty$, the Butterworth filter becomes an ideal filter. The Butterworth filter and its effect are shown respectively in the 6th and 7th rows of Fig.4.17.

- The *Gaussian filter* can be defined in either frequency or time domain as (3.148):

$$H(f) = e^{-\pi(f/f_c)^2}, \quad \text{or} \quad h(t) = \frac{1}{c} e^{-\pi(f_c t)^2} \tag{4.226}$$

Here the width of the passing band can be controlled by the cut-off frequency f_c. Obviously the Gaussian filter is smooth in both time and frequency domains without any ringing effect. The Butterworth filter and its effect are shown respectively in the 8th and 9th rows of Fig.4.17.

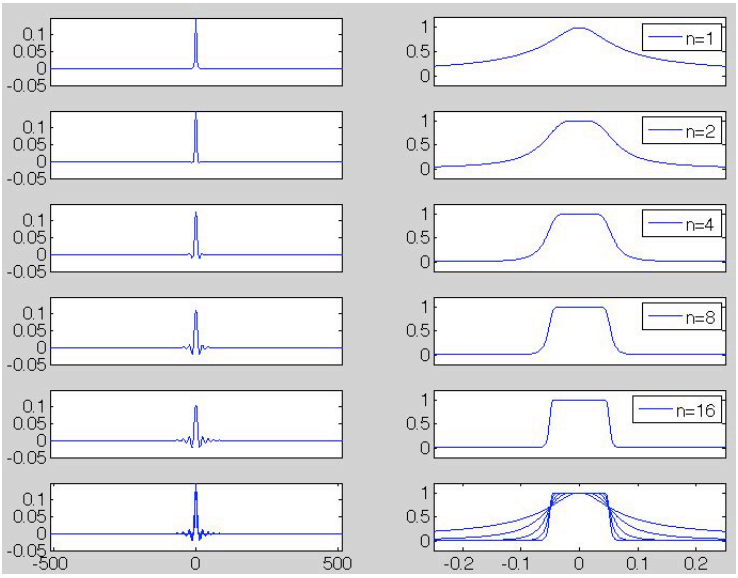

Figure 4.16 Butterworth filters of different orders in both time (left) and frequency right) domains
The plot in the last row compares all five filters of different orders.

Inspecting the filtered signal in time domain, we see that as expected, the sharp corners of the square impulses corresponding to the high frequency components are smoothed by all four low-pass filters. However, these filters each have different filtering effects. Most noticeably, the ringing effect caused by the ideal filter (also high order Butterworth filters) is obviously some undesirable artifact. To prevent such artifact, a filter that is smooth in frequency domain should be used, such as low-order Butterworth, Gaussian and other smooth filters based on cosine functions (e.g., Hamming and Hann filters).

Based on LP-filters, other types of high-pass, band-pass and band-stop filters can also be obtained. Specifically, if $H_{lp}(f)$ is a LP-filter with $H_{lp}(0) = 1$, then a HP-filter can be obtained as:

$$H_{hp}(f) = 1 - H_{lp}(f) \tag{4.227}$$

Also a band-pass filter can be obtained as the difference between two LP-filters $H_{lp1}(f)$ and $H_{lp2}(f)$ with their corresponding cut-off frequencies satisfying $f_1 > f_2$:

$$H_{bp}(f) = H_{lp1}(f) - H_{lp2}(f) \tag{4.228}$$

Finally, a band-stop filter is obtained simply as

$$H_{bs}(f) = 1 - H_{bp}(f) \tag{4.229}$$

Examples of such filters based on a 4th order Butterworth LP-filters are shown in Fig.4.18.

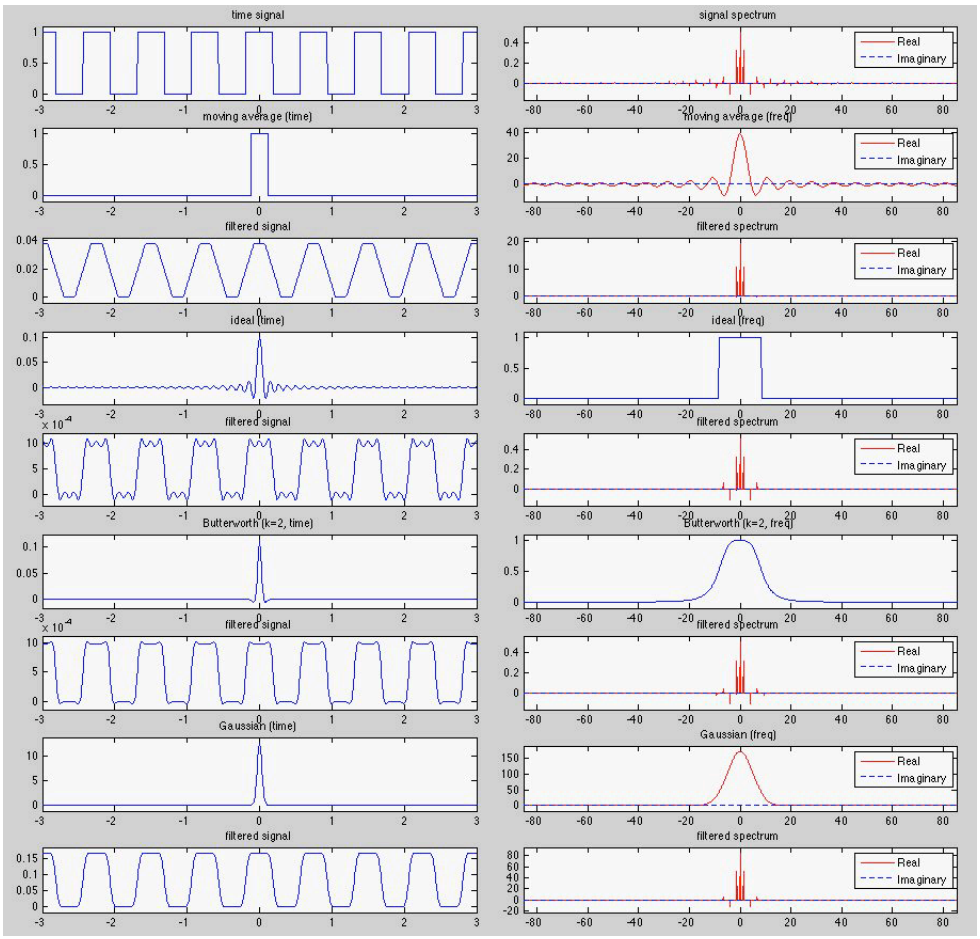

Figure 4.17 1-D low-pass filters in both time (left) and frequency (right) domains
A signal $x(t)$, a square impulse train, and its spectrum $X(f)$ are shown in the top row. Following that there are four pairs of plots showing each of the four types of filters (moving average, ideal, Butterworth and Gaussian) and their filtering effects. The left column shows in time domain the impulse response function $h(t)$ of each of the filters and the filtered time signal $y(t) = h(t) * x(t)$, while the right column shows in frequency domain the corresponding frequency response function $H(f)$ of the filter and the filtered signal spectrum $Y(f) = H(f)X(f)$.

Example 4.11: The annual precipitation in Los Angeles area in the $N = 126$ years from 1878 to 2003 is considered as a discrete time signal $x[m]$, and its spectrum $X[n]$ can be obtained by the DFT, as show in the top row of Fig.4.19. 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.

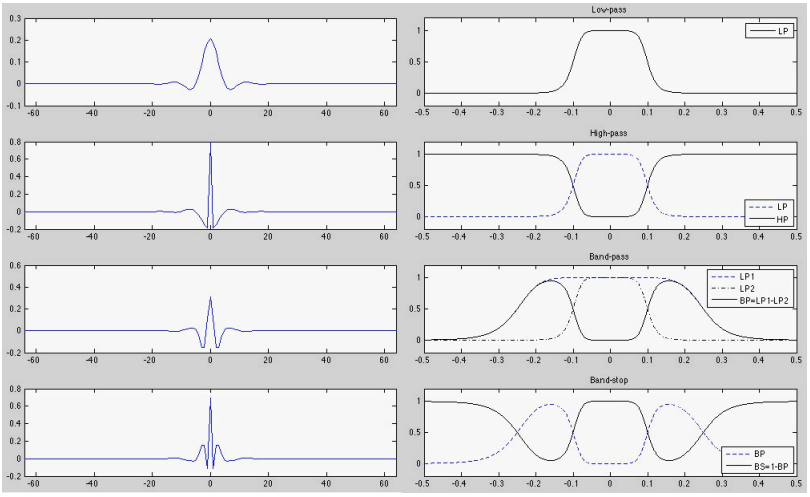

Figure 4.18 The frequency response functions $H(f)$ for the LP, HP, BP and BS filters in frequency domain (right) and their corresponding impulse response functions $h(t)$ in time domain (right)

Also shown in the figure are four Butterworth filters including a LP-filter (2nd row), and three BP-filters with different passing bands (4th, 6th and 8th rows), and the signals filtered by the corresponding filter (3rd, 5th, 7th and 9th rows). For each filter, its frequency response function $H(f)$ and the filtered signal spectrum $Y(f) = H(f)X(f)$ are shown on the right, and its impulse response function $h(f)$ and the filtered time signal $y(t) = h(t) * x(t)$ are shown on the left.

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 component filters $H_k(f)$ $(k = 1, \cdots, 4)$ add up approximately to a constant 1 through out all frequencies, i.e., the combined outputs of the filter bank contain approximately all information in the signal. These result is further confirmed by the last (10th) row in Fig.4.19 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 4.12: A two-dimensional shape in an image can be described by all the pixels along its boundary, in terms of there coordinates $(x[m], y[m])$, $(m = 1, \cdots, N)$, where N is the total number of pixels along the boundary. The coordinates $x[m]$ and $y[m]$ can be treated, respectively, as the real and imaginary components of a complex number $z[m] = x[m] + j\ y[m]$, and the Fourier transform can be carried out to obtain the Fourier coefficients, called the *Fourier*

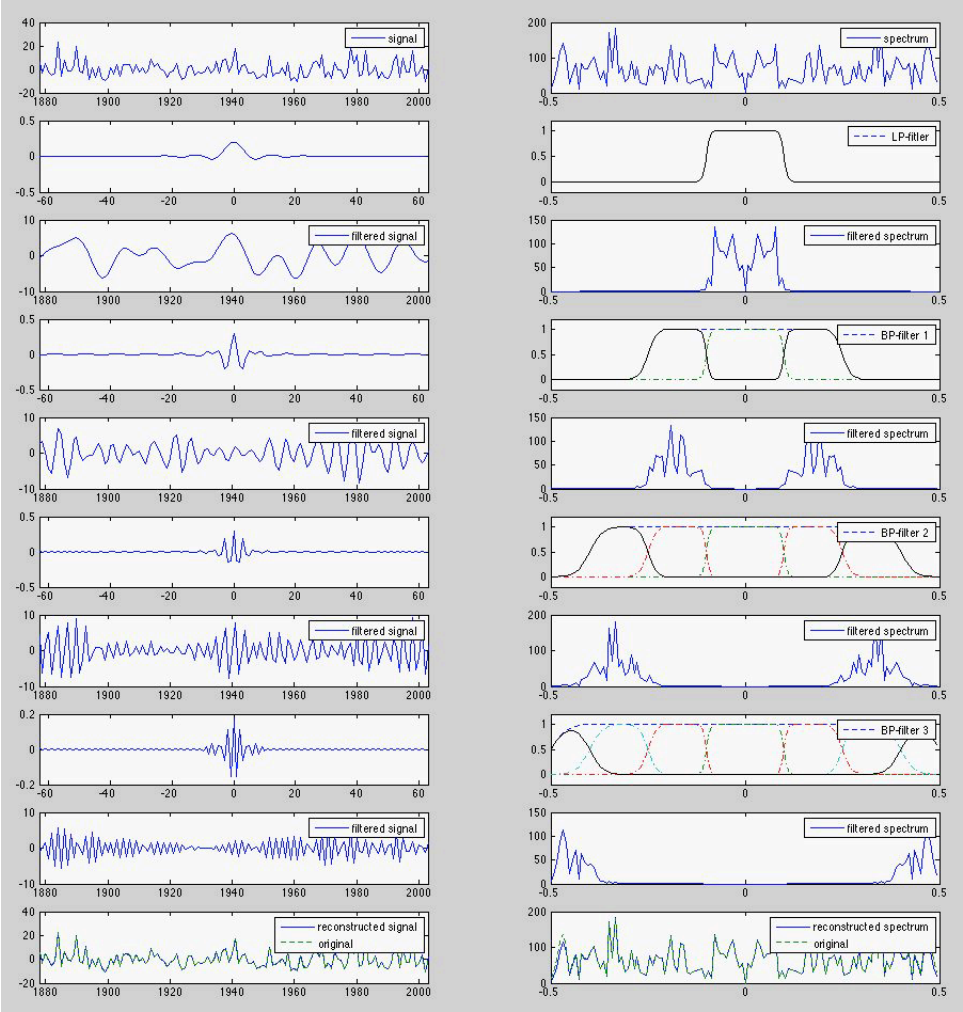

Figure 4.19 Annual precipitation from 1878 to 2003 (left) and its spectrum (right) Here only the magnitude of each spectrum is shown while the phase is neglected.

descriptors in the field of image process: of the shape:

$$Z[n] = \frac{1}{\sqrt{N}} \sum_{m=1}^{N} z[m] e^{-j2\pi mn/N}, \qquad n = 1, \cdots, N \tag{4.230}$$

Based on these coefficients $Z[n]$, the original shape can be reconstructed by inverse Fourier transform:

$$z[m] = \frac{1}{\sqrt{N}} \sum_{n=1}^{N} Z[n] e^{j2\pi mn/N}, \qquad m = 1, \cdots, N \tag{4.231}$$

The inverse Fourier transform using all N coefficients will perfectly reconstruct the original one. While this result is not surprising at all, it is interesting to observe the reconstructed shape using only the first $M < N$ low frequency components. Note that since the Fourier transform is a complex transform with both negative frequencies as well as positive ones in the frequency spectrum, the inverse transform with M components needs to contain both positive and negative terms symmetric to the DC component in the middle:

$$\hat{z}[m] = \sum_{k=-M/2}^{M/2} Z[k] e^{j2\pi mk/N} \quad (m = 1, \cdots, N) \tag{4.232}$$

As an example, the shape of Gumby as shown in Fig.4.20 is represented by a chain of $N = 1,157$ pixels along the boundary, in terms of their coordinates $\{x[m], y[m]\}$ $(m = 0, 1, \cdots, N-1 = 1156)$, which are then Fourier transformed to obtain the same number of Fourier coefficients as the Fourier descriptors of the Gumby figure.

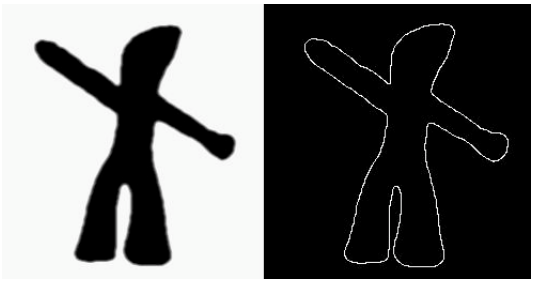

Figure 4.20 Gumby (left) and its boundary pixels (right)

The two different representations of the shape, $z[m]$ in spatial domain and $Z[n]$ in frequency domain are plotted in Fig. 4.21. Note that as the magnitudes of a small number of complex coefficients for to 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 absolute value of the magnitudes of all DFT coefficients, so that those coefficients with small magnitudes do not appear to be zero in the plots. The original shape of Gumby can be reconstructed by the inverse DFT using $M \leq N$ coefficients. As the DFT is a complex transform with negative as well as positive frequency components in the spectrum, the inverse DFT needs to contain both positive and negative terms on both sides of the DC component in the middle:

$$\hat{z}[m] = \sum_{k=-M/2}^{M/2} Z[k] e^{j2\pi mk/N} \quad (m = 1, \cdots, N) \tag{4.233}$$

The reconstructed shapes are shown in Fig. 4.22. The first row shows the reconstructions based on the first 1 to 4 low frequency components, while the second row shows the reconstructions using the first 5 to 8 components. Finally the last

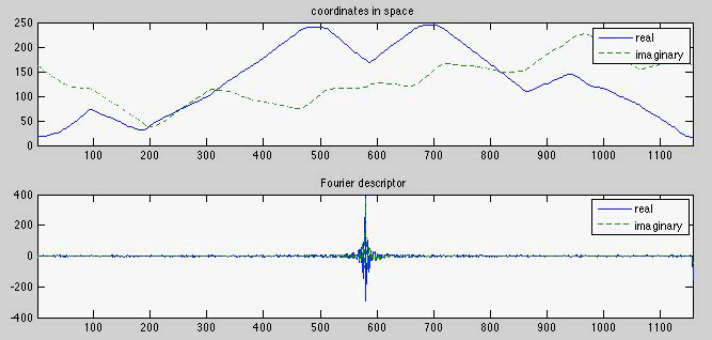

Figure 4.21 The vertical and horizontal components of 2-D shape (top) and its Fourier descriptors (bottom)

row shows reconstructions using the first 10, 20, 30 and all N=1,257 coefficients, respectively. In particular, it is interesting to compare the last figure, perfectly reconstructed using all N=1157 frequency components with the second to the last, reconstructed using only 30 components. They look almost identical, except the last one may have some very minor details of the shape, such as the sharper corners. This result shows that the remaining 1127 frequency components contain little information, and can therefore be ignored (treated as zero) in the inverse DFT with little effect in terms of the quality of the reconstruction. Moreover, it is all likely that some high frequency components may contain random noise. In this sense, the reconstruction without using many of the higher frequency components is actually a low-pass filtering process desirable for removing unwanted noise.

Some observations can be made based on the results discussed above:

- **Fourier transform tends to compact signal energy**
  The values of a few coefficients representing mostly low frequency components have significantly higher values than the rest of the coefficients, indicating that most of the energy contained in a signal is concentrated around the low frequency region. This phenomenon is common in various applications, due to the fact that in most physical signals relatively slow changes over time or space are much more significant compared to rapid and sudden changes. In other words, most natural signals are continuous and smooth.
- **Fourier transform tends to decorrelate a signal**
  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 $x[m]$ at position m, one can estimate the value $x[m+1]$ at the next position with reasonable confidence. However, the same thing can not be said in the spatial frequency domain, as the magnitudes of the DFT coefficients seem random. Given $X[n]$, one has little idea about the next value $X[n+1]$. In other words, the signal is highly correlated in spatial domain but significantly decorrelated in frequency domain after the Fourier transform.

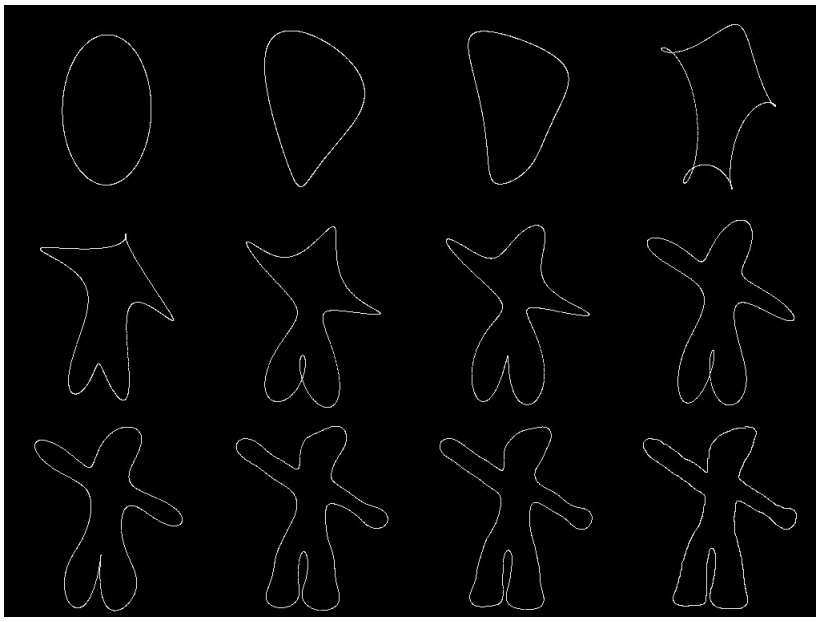

Figure 4.22 Reconstruction of a 2-D shape

- As most of the signal energy is concentrated in a small number of low frequency components, little error will result if only $M < N$ of the coefficients corresponding to low frequencies are used in the inverse DFT for the reconstruction of the figure in space. Such a low-pass filtering may also have the effect of removing unwanted high frequency noise.

This example illustrates the 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 of the others ignored. By doing so, we could process, store and transmit only a small portion of the data without losing much information (30 out of 1,157 coefficients used in this example is a impressive compression ratio).

Moreover, the observations made here for the Fourier transform are also valid in general for all other orthogonal transforms, as they will appear repetitively in our future discussions in the following chapters.

4.4.2 2-D Filtering and Compression

In the spatial frequency domain, the discrete spectrum $F[k, l]$ of a 2-D spatial signal $x[m, n]$ (e.g., an image) can be easily manipulated according to the specific application. Most commonly a filter function $H[k, l]$ is used to modify the discrete

spectrum:

$$F'[k,l] = H[k,l]F[k.l], \quad (k,l = 0, \cdots, N-1) \tag{4.234}$$

and the filtered spectrum can then be inverse transformed back to spatial domain to get the filtered signal:

$$x'[m,n] = \mathcal{F}^{-1}[F'[k,l]] \tag{4.235}$$

Typical filters include various high, band, and low pass or stop filters. We will only consider a few commonly used filters below, which suppress the frequency components around the corners and edges of the array for the 2-D discrete spectrum, while keeping the frequency components around the central area of the array unchanged. Such filters can be used for either low-pass or high-pass filtering, depending on whether the discrete spectrum is centralized with low frequency components in the middle or not.

We assume the 2-D signal is an N by N array $x[m,n]$ and so is its 2-D discrete spectrum $X[k,l]$. We define $d_k = k - N/2, d_l = l - N/2$ as the distances of a point $[k,l]$ of the 2-D spectrum to the center $(N/2, N/2)$ in vertical and horizontal directions, respectively. The spatial frequency represented by a coefficient $F[k,l]$ is $\sqrt{k^2+l^2}/N$ proportional to the distance $\sqrt{d_k^2+d_l^2}$. Here we consider some typical 2-D filters.

- **Ideal filter**

$$H_{ideal}[k,l] = \begin{cases} 1 & \sqrt{d_k^2+d_l^2} < D_0 \\ 0 & \text{otherwise} \end{cases} \tag{4.236}$$

  where $0 < D_0 < N/2$ corresponds to a cut-off frequency. Ideal filter completely removes any frequency components outside the circle determined by the cut-off frequency.
- **Gaussian low-pass filter**

$$H_{gauss}[k,l] = exp[-a(d_k^2+d_l^2)/D_0^2] \tag{4.237}$$

  where D_0 is the cut-off frequency at which $(d_k^2+d_l^2 = D_0^2)$ the magnitude is attenuated to $exp(-a)$, and a and c are two parameters.
- **Butterworth filter**

$$H_{butterworth}[k,l] = \frac{1}{1+((d_k^2+d_l^2)/D_0^2)^n} \tag{4.238}$$

  In particular, when $d_k = d_l = 0$, $H = 1$, when $d_k^2+d_l^2 = D_0^2$, $H = 0.5$. Butterworth filter is also a smooth low-pass filter with a parameter n. When $n \to \infty$, the Butterworth filter becomes an ideal filter.
- **Hamming filter**

$$H_{hamming} = \begin{cases} 0.5(1+\cos(\pi\sqrt{u^2+v^2}/D_0)) & 0 < \sqrt{u^2+v^2} < D_0 \\ 0 & \text{otherwise} \end{cases} \tag{4.239}$$

These filters can be represented in both spatial and frequency domains, as shown in Fig.4.23. When in frequency domain the spectrum of a 2-D signal is multiplied by the filter (left in the figure), correspondingly in time domain the 2-D signal is convolved with the inverse Fourier transform of the filter (right in the figure).

As these filters are all central symmetric (with respect to the center at $(N/2, N/2)$, when both the real and imaginary parts of the signal spectrum are identically multiplied by the filter, the symmetry property of the spectrum is not changed. If the 2-D signal is real, the inverse transform of the filtered spectrum is guaranteed to be also real.

Whether these filters $H[k, l]$ are low-pass or high-pass filters depends on whether or not the discrete 2-D spectrum is centralized with DC component in the center. If so, the low frequency components around the center $(N/2, N/2)$ will be kept, while high frequency components farther away from the center will be reduced. But if the spectrum is not centralized, the high frequency components will be around the middle region and therefore kept, the filters become high-pass. Alternatively, corresponding to each filter H_{lp}, another filter can be obtained by

$$H_{hp} = 1 - H_{lp} \tag{4.240}$$

which will turn a low-pass filter to high pass, and vice versa.

Example 4.13: We first consider the 2-D Fourier transform of an image, a typical 2-D signal, as show in Fig.4.24. An image of a panda (left) is treated as the real part of a 2-D signal and the imaginary part is set to zero, and then the even real part (middle) and odd imaginary part (right) of its Fourier spectrum are obtained. As the signal energy is in general always highly concentrated in a small number of low frequency components around DC, which show up as a bright spot in the center area of the spectrum image, while the rest of the image looks dark. In order to visualize other frequency components with small magnitudes away from the center, the pixel values of the image is transformed by a nonlinear function $y = x^{\alpha}$, where α is a fraction such as 0.2, 0.3, etc. so that the low pixel values representing the frequency components with low magnitudes are relatively enhanced and become visible in the image.

Alternatively, the spectrum can also be represented in terms of its magnitude and phase, as shown in Fig.4.25. The bottom row shows the magnitude and phase of the spectrum of another image of cat. Both images can be perfectly reconstructed by the inverse Fourier transform based on the real and imaginary parts, or equivalently, the magnitude and phase of its spectrum.

While it is obvious that the real and imaginary parts of the spectrum are equally important in terms of the amount of signal information they each carry, are the magnitude and phase components of the frequency components also equally important? To answer this question, two images are reconstructed based on the magnitude of the spectrum of one image but the phase of the other, the

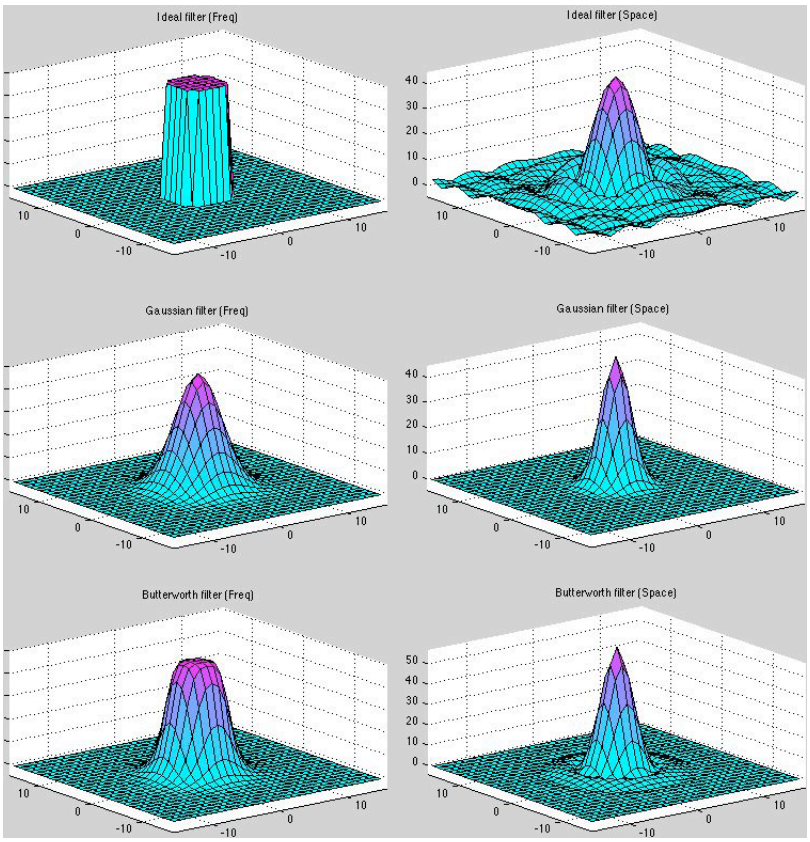

Figure 4.23 2-D filters in both spatial and frequency domains

results are shown on the right in Fig.4.25. It is obvious that the phases play a more significant and dominant role than the magnitudes, as the reconstructed image always looks similar to whichever image whose phase is used in the reconstruction.

Based on this simple observation, we see that during the filtering process in frequency domain, the real $Re[X]$ and imaginary $Im[X]$ parts of the spectrum need to be treated identically so that the phase angle $\phi = \tan^{-1} Im[X]/Re[X]$ of each frequency component remains the same after filtering, so that the relative positions of different frequency components also remain the same.

Example 4.14: In this example we illustrate the effect of the Fourier filtering an image shown in the previous example in Fig. 4.24. Different types of filtering of this image can be carried out in the frequency domain. First, the effects of ideal filtering are shown in Fig. 4.26, including the filter (left), and the two resulting images after low-pass (middle) and high-pass (right) filtering. The top row shows

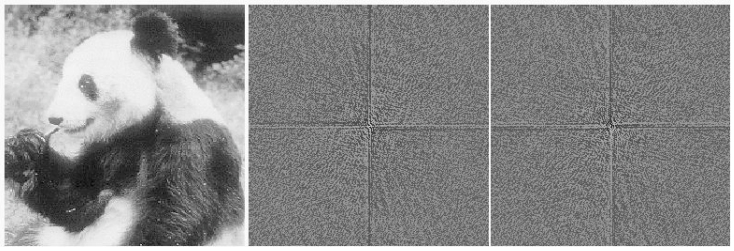

Figure 4.24 An image (left) and its 2-D Fourier transform

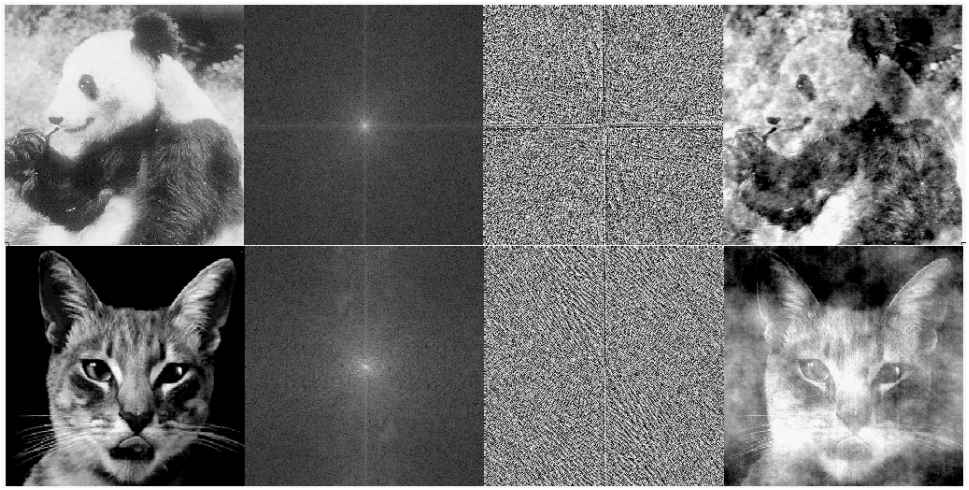

Figure 4.25 Magnitude and phase of Fourier spectra
Given the magnitude and phase components (middle two) of the spectra of two images, two images are reconstructed (right) based on the phase of one but the magnitude of the other (phase of panda but magnitude of cat on top, phase of cat but magnitude of panda at the bottom).

the spectrum in frequency domain while the bottom row shows the corresponding image in spatial domain. In the case of ideal low-pass filtering, all frequency components are suppressed to zero except those low frequency components inside the circle corresponding to a cut-off frequency, which remain unchanged. Inversely, in the case of high-pass filtering, all low frequency components inside the circle are suppressed to zero while other components outside the circle remain unchanged

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.3.211), as shown in the bottom row. Note that the resulting low-pass and high-pass filtered images have some obvious ringing effect, due to the shape of the sinc function.

To avoid this artifacts caused by the sharp edge of the ideal low-pass filter, we can instead use a Butterworth filter without sharp edges, shown in Fig. 4.27,

so that the ringing effect in spatial domain can be significantly reduced. As seen in the bottom row of the figure, the low-pass and high-pass filtered images no longer suffer from the artifacts seen before in Fig. 4.26.

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. For example, in Fig. 4.28, 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 row). In the corresponding images reconstructed by the inverse transform of each directionally low-passed spectrum (bottom row), the image features in the orientation favored by the directional filtering are emphasized.

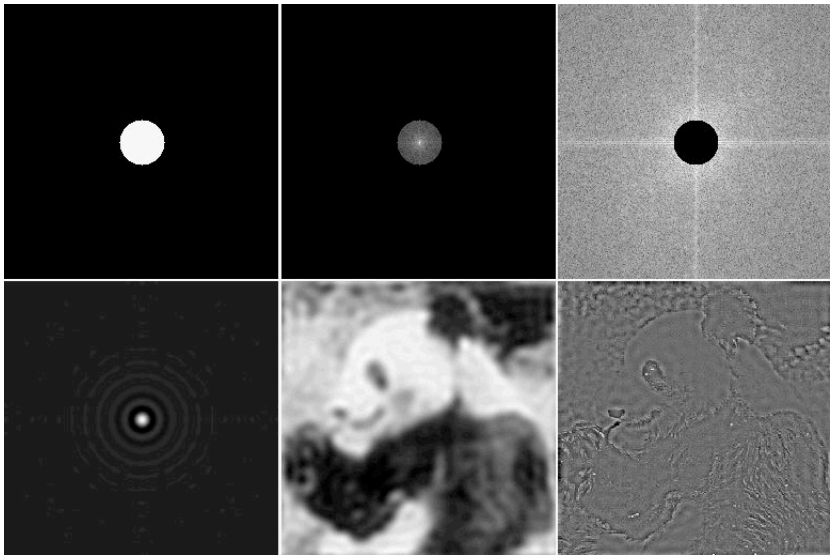

Figure 4.26 Ideal filtering (from left to right, ideal filter, low-pass, high-pass)

Example 4.15: The simple example shown in Fig. 4.29 illustrates why the Fourier transform can also be used for data compression. The image of panda and its 2-D Fourier spectrum are shown in the lower and upper left panels, respectively. Then 80% of the DFT coefficients (corresponding mostly to some high frequency components) with magnitudes less than a certain threshold value were surprised to zero as shown in the upper right panel (black in the image). The image is then reconstructed by the inverse transform based only on the remaining 20% of the DFT coefficients but containing over 99% of the signal energy. As can be seen in the lower right panel, the reconstructed 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 which were suppressed.

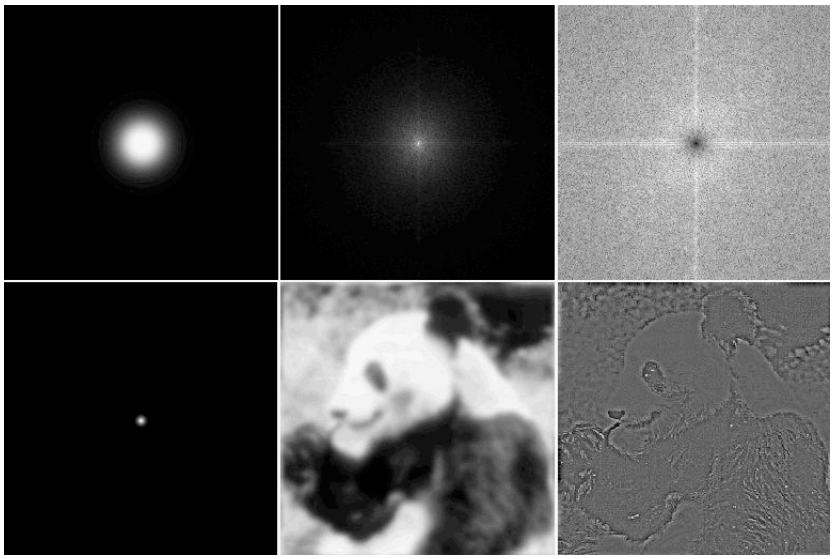

Figure 4.27 Butterworth filtering (from left to right, ideal filter, low-pass, high-pass)

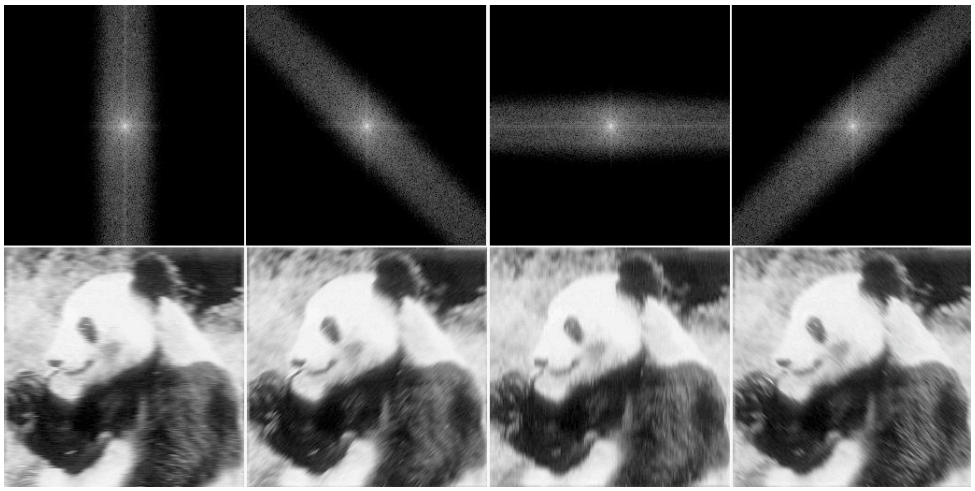

Figure 4.28 Directional low-pass filtering

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 for us to do so in spatial domain? This is obviously due to the two general properties of the Fourier transform (as well as 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 in transform based data compression. In practice, the compression process also includes many other components such

as the quantization and encoding of frequency components. Interested reader can do some further reading about image compression standards, such as JPEG.

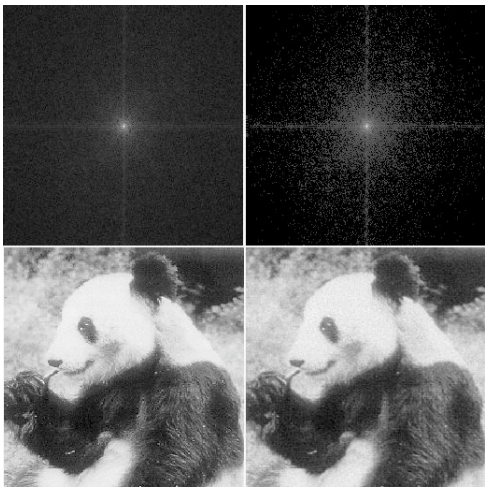

Figure 4.29 Image Compression based on DFT
An image (lower left) and its 2-D DFT spectrum (upper left) and the reconstructed image (lower right) based on 20% of its DFT coefficients containing 99% of the total energy (upper right).

5 The Laplace and Z Transforms

The Laplace and Z transforms are the natural generalization of the continuous and discrete-time Fourier transforms, respectively, 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 s-domain or z-domain. While different forms of the Fourier transform are used mostly for continuous and discrete signal processing and filtering, the Laplace and Z-transforms are particularly useful for the analysis and design of various linear and time-invariant (LTI) systems.

5.1 The Laplace Transform

5.1.1 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:

$$\mathcal{F}[x(t)] = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}dt = X(j\omega) \tag{5.1}$$

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 above transform is based on the underlying assumption that the signal $x(t)$ is square integrable so that the integral converges and the spectrum 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 grow without a bound when $|t| \to \infty$ and are not square integrable. 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 $t \to \infty$, therefore its Fourier spec-

trum does not exist. However, if we choose $Re[s] = \sigma > a$, the modified version $e^{-(\sigma-a)t}u(t)$ will converge as $t \to \infty$.

In general, the Fourier transform of the modified signal is:

$$\mathcal{F}[x'(t)] = \int_{-\infty}^{\infty} x(t)e^{-\sigma t}e^{-j\omega t}dt = \int_{-\infty}^{\infty} x(t)e^{-(\sigma+j\omega)t}dt = \int_{-\infty}^{\infty} x(t)e^{-st}dt \quad (5.2)$$

where s is a complex variable defined as $s = \sigma + j\omega$. If the integral above converges, it results in a complex function $X(s)$, which is called the *bilateral Laplace transform* of $x(t)$, formally defined as:

$$X(s) = \mathcal{L}[x(t)] = \int_{-\infty}^{\infty} x(t)\phi(t,s)dt = \int_{-\infty}^{\infty} x(t)e^{-st}dt \quad (5.3)$$

Same as the continuous-time Fourier transform, the Laplace transform can also be considered as an integral transform with a kernel function:

$$\phi(t,s) = e^{-st} = e^{-(\sigma+j\omega)t} = e^{-\sigma t}e^{-j\omega t} \quad (5.4)$$

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 real and imaginary parts $Re[s] = \sigma$ and $Im[s] = \omega$, and the transform $X(s)$ is a complex function defined in a 2-D complex plane, called s-plane, represented by its 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. 5.3. 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 with respect to ω from $\omega = -\infty$ to $\omega = \infty$ to obtain the Fourier transform $X(j\omega)$ of $x(t)$. We see that the 1-D Fourier spectrum of the signal can be found as the cross section of the 2-D complex function $X(s) = X(\sigma + j\omega)$ along the imaginary axis $s = j\omega$. In other words, the continuous-time Fourier transform is just a special case of the Laplace transform when $\sigma = 0$ and $s = j\omega$:

$$\mathcal{F}[x(t)] = \mathcal{L}[x(t)]\big|_{s=j\omega} = X(s)\big|_{s=j\omega} = X(j\omega) \quad (5.5)$$

This is the reason why sometimes the Fourier spectrum is also 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:

$$\mathcal{L}[x(t)] = X(s) = X(\sigma + j\omega) = \int_{-\infty}^{\infty} x(t)e^{-(\sigma+j\omega)t}dt = \mathcal{F}[x(t)e^{-\sigma t}] \quad (5.6)$$

Taking the inverse Fourier transform of the above, we get

$$x(t)e^{-\sigma t} = \mathcal{F}^{-1}[X(\sigma + j\omega)] = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\sigma + j\omega)e^{j\omega t}d\omega \tag{5.7}$$

Multiplying both sides by $e^{\sigma t}$, we get:

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\sigma + j\omega)e^{(\sigma+j\omega)t}d\omega \tag{5.8}$$

To further represent this inverse transform in terms of s (instead of ω), we note

$$ds = d(\sigma + j\omega) = j\ d\omega, \quad i.e., \quad d\omega = ds/j \tag{5.9}$$

The integral over $-\infty < \omega < \infty$ with respect to ω corresponds to the integral with respect to s over $\sigma - j\infty < s < \sigma + j\infty$:

$$x(t) = \mathcal{L}^{-1}[X(s)] = \frac{1}{j2\pi}\int_{\sigma-j\infty}^{\sigma+j\infty} X(s)e^{st}ds \tag{5.10}$$

Now we get the forward and inverse Laplace transform pair:

$$\begin{aligned} X(s) &= \mathcal{L}[x(t)] = \int_{-\infty}^{\infty} x(t)e^{-st}dt \\ x(t) &= \mathcal{L}^{-1}[X(s)] = \frac{1}{j2\pi}\int_{\sigma-j\infty}^{\sigma+j\infty} X(s)e^{st}ds \end{aligned} \tag{5.11}$$

which can also be more concisely represented as

$$x(t) \overset{\mathcal{L}}{\longleftrightarrow} X(s) \tag{5.12}$$

In practice, we hardly need to carry out the integral in the inverse transform with respective to the complex variable s, as the Laplace transform pairs of most of the signals of interest can be obtained in some other ways and made available in table form.

As we will see later, in some applications the Laplace transform is a rational function as a ratio of two polynomials:

$$X(s) = \frac{N(s)}{D(s)} = \frac{\sum_{k=0}^{m} b_k s^k}{\sum_{k=0}^{n} a_k s^k} = \frac{b_m}{a_n}\frac{\prod_{k=1}^{m}(s - z_k)}{\prod_{k=1}^{n}(s - p_k)} \tag{5.13}$$

where the roots $z_k, (k = 1, 2, \cdots, m)$ of the numerator polynomial $N(s)$ of order m are called the *zeros* of $X(s)$, and the roots $p_k, (k = 1, 2, \cdots, n)$ of the denominator polynomial $D(s)$ of order n are called the *poles* of $X(s)$. The last equal sign in the equation above is due to the fundamental theorem of algebra, which states that an nth order polynomial has n roots (some of which may be repeated with multiplicity greater than 1). Obviously we have:

$$X(z_k) = 0, \quad \text{and} \quad X(p_k) = \infty \tag{5.14}$$

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 the long division for $N(s)/D(s)$ to expand

$X(s)$ into multiple terms so that $m < n$ is true for each term. 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.

5.1.2 The Region of Convergence

The region of convergence plays an important role in the Laplace transform. A Laplace transform $X(s)$ 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 5.1:

1. A right-sided signal $x(t) = e^{-at}u(t)$:

$$X(s) = \int_0^\infty e^{-at}e^{-st}dt = \int_0^\infty e^{-at}e^{-(\sigma+j\omega)t}dt = \int_0^\infty e^{-(a+\sigma)t}e^{-j\omega t}dt \quad (5.15)$$

where a is a real constant. For this integral to converge, it is necessary to have $a + \sigma > 0$, i.e. the ROC is $Re[s] = \sigma > -a$, inside which the above becomes:

$$X(s) = \frac{1}{-(a+\sigma+j\omega)}\, e^{-(a+\sigma+j\omega)t}\Big|_0^\infty = \frac{1}{(\sigma+a)+j\omega} = \frac{1}{s+a} \quad (5.16)$$

In particular, if $a = 0$, $x(t) = u(t)$ and we have

$$U(s) = \mathcal{L}[u(t)] = \frac{1}{s}, \qquad \sigma > 0 \quad (5.17)$$

If we let $\sigma \to 0$, 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 $R[s] > 0$. Comparing this result with the real Fourier transform of $u(t)$ in Eq.3.65:

$$\mathcal{F}[u(t)] = \frac{1}{2}\delta(f) + \frac{1}{j\omega} \quad (5.18)$$

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 $x(t) = -e^{-at}u(-t)$:

$$X(s) = -\int_{-\infty}^0 e^{-at}e^{-st}dt = -\int_{-\infty}^0 e^{-(a+\sigma+j\omega)t}dt \quad (5.19)$$

where a is a real constant. For this integral to converge, it is necessary that $a + \sigma < 0$, i.e., the ROC is $Re[s] = \sigma < -a$, inside which the above becomes:

$$X(s) = \frac{1}{a+\sigma+j\omega}\, e^{-(a+\sigma+j\omega)t}\Big|_{-\infty}^0 = \frac{1}{a+\sigma+j\omega} = \frac{1}{s+a}, \qquad \sigma < -a \quad (5.20)$$

When $a = 0$, $x(t) = -u(-t)$ we have

$$\mathcal{L}[-u(-t)] = \frac{1}{s}, \quad \sigma < 0 \tag{5.21}$$

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.

3. A two-sided signal $x(t) = e^{-a|t|} = e^{-at}u(t) + e^{at}u(-t)$:
   As the Laplace transform is linear, the transform of this signal is the sum of the transforms of the two individual terms. According to the results in the previous two cases, we get:

$$X(s) = \frac{1}{s+a} - \frac{1}{s-a} = \frac{-2a}{s^2 - a^2}, \qquad \sigma > -a, \;\; \sigma < a \tag{5.22}$$

   provided the intersection of the two individual ROCs is non-empty, i.e., $-a < \sigma < a$, which is possible only if $a > 0$, i.e., $x(t)$ decays when $|t| \to \infty$. However, if $a < 0$, the intersection of the two ROCs is an empty set, and the Laplace transform does not exist, reflecting the fact that $x(t)$ grows without bound when $|t| \to \infty$.

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 because by definition $X(s)$ 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 real part $Re[s] = \sigma$ of s determines the convergence of the integral in the Laplace transform and thereby the ROC. The imaginary part $Im[s]$ 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 $Re[p_1] < Re[p_2]$. It is possible that $Re[p_1] = -\infty$ and/or $Re[p_2] = \infty$.
- 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 $Re[s] = 0$, i.e., $s = j\omega$.

The zeros and poles of the Laplace transform $X(s) = \mathcal{L}[x(t)]$ of a signal dictate the most essential properties such as whether it is right or left-sided, whether it

grows or decays over time. Moreover, the zeros and poles of the transfer function $H(s) = \mathcal{L}[h(t)]$ of an LTI system dictate its stability and filtering effects. All such properties and behaviors can be qualitatively characterized based on the locations of the zeros and poles of in the s-plane, as we will see in the later discussions.

5.1.3 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 ommitted 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:

$$\mathcal{L}[x(t)] = X(s), \qquad \mathcal{L}[y(t)] = Y(s) \tag{5.23}$$

with ROCs R_x and R_y, respectively.

- **Linearity**

$$\mathcal{L}[ax(t) + by(t)] = aX(s) + bY(s), \quad ROC \supseteq (R_x \cap R_y) \tag{5.24}$$

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

$$X(s) = \mathcal{L}[x(t)] = \frac{1}{s+1}, \quad Re[s] > -1 \tag{5.25}$$

and

$$Y(s) = \mathcal{L}[y(t)] = \frac{1}{(s+1)(s+2)}, \quad Re[s] > -1 \tag{5.26}$$

then

$$\mathcal{L}[x(t) - y(t)] = \frac{1}{s+1} - \frac{1}{(s+1)(s+2)} = \frac{s+1}{(s+1)(s+2)} = \frac{1}{s+2}, \quad Re[s] > -2 \tag{5.27}$$

- **Time shifting**

$$\mathcal{L}[x(t-t_0)] = e^{-t_0 s}X(s), \quad ROC = R_x \tag{5.28}$$

- **Time reversal**

$$\mathcal{L}[x(-t)] = X(-s), \qquad ROC = -R_x \tag{5.29}$$

- **s-Domain shifting**

$$\mathcal{L}[e^{-s_0 t}x(t)] = X(s+s_0), \quad ROC = R_x + Re[s_0] \tag{5.30}$$

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**

$$\mathcal{L}[x(at)] = \frac{1}{|a|}X(\frac{s}{a}), \quad ROC = \frac{R_x}{a} \tag{5.31}$$

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**

$$\mathcal{L}[x^*(t)] = X^*(s^*), \quad ROC = R_x \tag{5.32}$$

- **Convolution**

$$\mathcal{L}[x(t) * y(t)] = X(s)Y(s), \quad ROC \supseteq (R_x \cap R_y) \tag{5.33}$$

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

$$X(s) = \mathcal{L}[x(t)] = \frac{s+1}{s+2}, \quad Re[s] > -2 \tag{5.34}$$

$$Y(s) = \mathcal{L}[y(t)] = \frac{s+2}{s+1}, \quad Re[s] > -1 \tag{5.35}$$

then

$$\mathcal{L}[x(t) * y(t)] = X(s)Y(s) = 1 \tag{5.36}$$

with an ROC of the entire s-plane.

- **Differentiation in time domain**

$$\mathcal{L}[\frac{d}{dt}x(t)] = sX(s), \quad ROC \supseteq R_x \tag{5.37}$$

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:

$$\frac{d}{dt}x(t) = \frac{1}{j2\pi}\int_{\sigma-j\infty}^{\sigma+j\infty} X(s)\frac{d}{dt}e^{st}ds = \frac{1}{j2\pi}\int_{\sigma-j\infty}^{\sigma+j\infty} sX(s)e^{st}ds \tag{5.38}$$

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:

$$\mathcal{L}[\frac{d^n}{dt^n}x(t)] = s^n X(s) \tag{5.39}$$

In particular, when $x(t) = \delta(t)$, we have

$$\mathcal{L}[\frac{d^n}{dt^n}\delta(t)] = s^n, \quad ROC = \text{entire s-plane} \tag{5.40}$$

- **Differentiation in s-Domain**

$$\mathcal{L}[tx(t)] = -\frac{d}{ds}X(s), \quad ROC = R_x \tag{5.41}$$

This can be proven by differentiating the Laplace transform:

$$\frac{d}{ds}X(s) = \int_{-\infty}^{\infty} x(t)\frac{d}{ds}e^{-st}dt = \int_{-\infty}^{\infty}(-t)x(t)e^{-st}dt \tag{5.42}$$

Repeat this process we get

$$\mathcal{L}[t^n x(t)] = (-1)^n \frac{d^n}{ds^n}X(s), \quad ROC = R_x \tag{5.43}$$

- **Integration in time domain**

$$\mathcal{L}[\int_{-\infty}^{t} x(\tau)d\tau] = \frac{X(s)}{s}, \qquad ROC \supseteq (R_x \cap \{Re[s] > 0\}) \tag{5.44}$$

This can be proven by realizing that

$$x(t) * u(t) = \int_{-\infty}^{\infty} x(\tau)u(t-\tau)d\tau = \int_{-\infty}^{t} x(\tau)d\tau \tag{5.45}$$

and therefore by convolution property we have

$$\mathcal{L}[x(t) * u(t)] = X(s)\frac{1}{s} \tag{5.46}$$

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-pane.

5.1.4 Laplace Transform of Typical Signals

- $\delta(t)$, $\delta(t-\tau)$

$$\mathcal{L}[\delta(t)] = \int_{-\infty}^{\infty}\delta(t)e^{-st}dt = e^0 = 1, \quad \text{ROC: entire } s \text{ plane} \tag{5.47}$$

Moreover, due to time shifting property, we have

$$\mathcal{L}[\delta(t-\tau)] = e^{-s\tau}, \quad \text{ROC: entire } s \text{ plane} \tag{5.48}$$

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

$$\mathcal{L}[u(t)] = \mathcal{L}\left[\int_{-\infty}^{t} \delta(\tau)d\tau\right] = \frac{1}{s}, \qquad Re[s] > 0 \tag{5.49}$$

  Applying the s-domain differentiation property to the above, we have

$$\mathcal{L}[tu(t)] = -\frac{d}{ds}\left[\frac{1}{s}\right] = \frac{1}{s^2}, \qquad Re[s] > 0 \tag{5.50}$$

  and in general

$$\mathcal{L}[t^n\,u(t)] = \frac{n!}{s^{n+1}}, \quad Re[s] > 0 \tag{5.51}$$

- $e^{-at}\,u(t)$, $te^{-at}\,u(t)$
  Applying the s-domain shifting property to

$$\mathcal{L}[u(t)] = \frac{1}{s}, \quad Re[s] > 0 \tag{5.52}$$

  we have

$$\mathcal{L}[e^{-at}u(t)] = \frac{1}{s+a}, \quad Re[s] > -a \tag{5.53}$$

  Applying the same property to

$$\mathcal{L}[t^n u(t)] = \frac{n!}{s^{n+1}}, \quad Re[s] > 0 \tag{5.54}$$

  we have

$$\mathcal{L}[t^n e^{-at}u(t)] = \frac{n!}{(s+a)^{n+1}}, \quad Re[s] > -a \tag{5.55}$$

- $e^{-j\omega_0 t}u(t)$, $sin(\omega_0 t)u(t)$, $cos(\omega_0 t)u(t)$
  Letting $a = \pm j\omega_0$ in

$$\mathcal{L}[e^{-at}u(t)] = \frac{1}{s+a}, \quad Re[s] > -Re[a] \tag{5.56}$$

  we get

$$\mathcal{L}[e^{-j\omega_0 t}u(t)] = \frac{1}{s+j\omega_0} \quad \text{and} \quad \mathcal{L}[e^{j\omega_0 t}u(t)] = \frac{1}{s-j\omega_0} \quad Re[s] > 0 \tag{5.57}$$

  and therefore

$$\mathcal{L}[cos(\omega_0 t)u(t)] = \frac{1}{2}\mathcal{L}[e^{j\omega_0 t} + e^{-j\omega_0 t}] = \frac{1}{2}[\frac{1}{s-j\omega_0} + \frac{1}{s+j\omega_0}] = \frac{s}{s^2+\omega_0^2} \tag{5.58}$$

  and

$$\mathcal{L}[sin(\omega_0 t)u(t)] = \frac{1}{2j}\mathcal{L}[e^{j\omega_0 t} - e^{-j\omega_0 t}] = \frac{1}{2j}[\frac{1}{s-j\omega_0} - \frac{1}{s+j\omega_0}] = \frac{\omega_0}{s^2+\omega_0^2} \tag{5.59}$$

- $t\,cos(\omega_0 t)u(t)$, $t\,sin(\omega_0 t)u(t)$
  Letting $a = \pm j\omega_0$ in

$$\mathcal{L}[te^{-at}u(t)] = \frac{1}{(s+a)^2}, \quad Re[s] > -a \tag{5.60}$$

we get

$$\mathcal{L}[te^{-j\omega_0 t}u(t)] = \frac{1}{(s+j\omega_0)^2}, \qquad \mathcal{L}[te^{j\omega_0 t}u(t)] = \frac{1}{(s-j\omega_0)^2}, \quad Re[s] > -a \tag{5.61}$$

Based on these we have:

$$\begin{aligned}\mathcal{L}[t\ cos(\omega_0 t)u(t)] &= \frac{1}{2}\mathcal{L}[t\ (e^{j\omega_0 t} + e^{-j\omega_0 t})] = \frac{1}{2}[\frac{1}{(s-j\omega_0)^2} + \frac{1}{(s+j\omega_0)^2}] \\ &= \frac{s^2 - \omega_0^2}{(s^2+\omega_0^2)^2}\end{aligned} \tag{5.62}$$

and

$$\begin{aligned}\mathcal{L}[t\ sin(\omega_0 t)u(t)] &= \frac{1}{2j}\mathcal{L}[t\ (e^{j\omega_0 t} - e^{-j\omega_0 t})] = \frac{1}{2j}[\frac{1}{(s-j\omega_0)^2} - \frac{1}{(s+j\omega_0)^2}] \\ &= \frac{2s\omega_0}{(s^2+\omega_0^2)^2}\end{aligned} \tag{5.63}$$

- $e^{-at}cos(\omega_0 t)u(t)$, $e^{-at}sin(\omega_0 t)u(t)$
  Applying s-domain shifting property to

$$\mathcal{L}[cos(\omega_0 t)u(t)] = \frac{s}{s^2+\omega_0^2}, \quad \text{and} \quad \mathcal{L}[sin(\omega_0 t)u(t)] = \frac{\omega_0}{s^2+\omega_0^2} \tag{5.64}$$

we get, respectively

$$\mathcal{L}[e^{-at}cos(\omega_0 t)u(t)] = \frac{s+a}{(s+a)^2+\omega_0^2} \tag{5.65}$$

and

$$\mathcal{L}[e^{-at}sin(\omega_0 t)u(t)] = \frac{\omega_0}{(s+a)^2+\omega_0^2} \tag{5.66}$$

Below we give a few more examples:

Example 5.2: The Laplace transform of the following function

$$x(t) = [e^{-2t} + e^t cos(3t)]u(t) = [e^{-2t} + \frac{1}{2}e^{-(1-j3)t} + \frac{1}{2}e^{-(1+j3)t}]u(0) \tag{5.67}$$

can be found as

$$\begin{aligned} X(s) &= \int_0^\infty [e^{-2t} + \frac{1}{2}e^{-(1-j3)t} + \frac{1}{2}e^{-(1+j3)t}]e^{-st}dt \\ &= \int_0^\infty e^{-2t}e^{-st}dt + \frac{1}{2}\int_0^\infty e^{-(1-j3)t}e^{-st}dt + \frac{1}{2}\int_0^\infty e^{-(1+j3)t}e^{-st}dt \\ &= \frac{1}{s+2} + \frac{1/2}{s+(1-j3)} + \frac{1/2}{s+(1+j3)} \end{aligned}$$

Following the examples above, we see that the conditions for the three integrals to converge are, respectively,

$$Re[s] > -2, \quad Re[s] > -1, \quad Re[s] > -1 \tag{5.68}$$

i.e., the ROC corresponding to this transform $X(s)$ is $Re[s] > -1$ that satisfies all three conditions. This $X(s)$ can be further written as a rational function, a ratio of two polynomials:

$$X(s) = \frac{1}{s+2} + \frac{1/2}{s+(1-j3)} + \frac{1/2}{s+(1+j3)} = \frac{2s^2+5s+12}{(s^2+2s+10)(s+2)}, \quad Re[s] > -1 \tag{5.69}$$

Example 5.3:

$$X(s) = \frac{s^2-3}{s+2} \tag{5.70}$$

As the order of the numerator $M = 2$ is higher than that of the denominator $N = 1$, we expand it into the following terms by partial fraction expansion:

$$X(s) = \frac{s^2-3}{s+2} = A + Bs + \frac{C}{s+2} \tag{5.71}$$

and get

$$s^2 - 3 = (A + Bs)(s + 2) + C = Bs^2 + (A + 2B)s + (2A + C) \tag{5.72}$$

Equating the coefficients for terms s^k $(k = 0, 1, \cdots, M)$ on both sides, we get

$$B = 1, \quad A + 2B = 0, \quad 2A + C = -3 \tag{5.73}$$

Solving this equation system, we get coefficients

$$A = -2; \quad B = 1, \quad C = 1 \tag{5.74}$$

and

$$X(s) = s - 2 + \frac{1}{s+2} \tag{5.75}$$

Alternatively, the same result can be obtained by carrying out a long division $(s^2 - 3) \div (s + 2)$.

Example 5.4: Let $x(t) = u(-1) - u(1)$, then

$$X(s) = \int_{-1}^{1} e^{st} dt = \frac{1}{s}(e^{s} - e^{-s}) \tag{5.76}$$

As $X(s) = \infty$ when $s = \infty$ or $s = -\infty$, neither of these two s values is included in the ROC. However, note that $s = 0$ is inside the ROC as $X(0) = 2$.

Example 5.5: Let $x(t)$ be a two-sided function $x(t) = e^{-a|t|} = e^{-at}u(t) + e^{at}u(-t)$ The transform of the first term is:

$$X_1(s) = \int_{0}^{\infty} e^{-at} e^{-st} dt = \frac{1}{s+a}, \quad Re[s] > -a \tag{5.77}$$

The transform of the second term is:

$$X_2(s) = \int_{-\infty}^{0} e^{at} e^{-st} dt = -\frac{1}{s-a}, \quad Re[s] < a \tag{5.78}$$

The Laplace transform of the two components is:

$$X(s) = X_1(s) + X_2(s) = \frac{1}{s+a} - \frac{1}{s-a} = \frac{-2a}{s^2 - a^2}, \quad -a < Re[s] < a \tag{5.79}$$

Whether $X(s)$ exists or not depends on a. If $a > 0$, i.e., $x(t)$ decays exponentially as $|t| \to \infty$, then the ROC is the strip between $-a$ and a and $X(s)$ exists. But if $a < 0$, i.e., $x(t)$ grows exponentially as $|t| \to \infty$, then the ROC is an empty set and $X(s)$ does not exist.

Example 5.6: Given the following Laplace transform, find the corresponding function:

$$X(s) = \frac{1}{(s+1)(s+2)} = \frac{1}{s+1} - \frac{1}{s+2} \tag{5.80}$$

Given the two poles $p_1 = -1$ and p_2 of the expression, there are three possible associated ROCs:

- The half plane to the right of the rightmost pole $p_2 = -1$, with the corresponding right-sided time function

$$x(t) = e^{-t}u(t) - e^{-2t}u(t) \tag{5.81}$$

- The half plane to the left of the leftmost pole $p_1 = -2$, with the corresponding left-sided time function

$$x(t) = -e^{-t}u(-t) + e^{-2t}u(-t) \tag{5.82}$$

- The vertical strip between the two poles $-1 < Re[s] < -2$, with the corresponding two sided time function

$$x(t) = -e^{-t}u(-t) - e^{-2t}u(t) \tag{5.83}$$

In particular, note that only the first ROC includes the $j\omega$-axis and the corresponding time function has a Fourier transform. Fourier transform of the other two functions do not exist.

5.1.5 Analysis of LTI Systems by Laplace Transform

The Laplace transform is a convenient tool for the analysis and design of continuous LTI systems whose output $y(t)$ is the convolution of the input $x(t)$ and its impulse response function $h(t)$:

$$y(t) = \mathcal{O}[x(t)] = h(t) * x(t) = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau \tag{5.84}$$

In particular, if the input is an impulse $x(t) = \delta(t)$, then the out is the impulse response function:

$$y(t) = \mathcal{O}[\delta(t)] = h(t) * \delta(t) = \int_{-\infty}^{\infty} h(\tau)\delta(t-\tau)d\tau = h(t) \tag{5.85}$$

If the input is a complex exponential $x(t) = e^{st} = e^{\sigma+j\omega}$, then the output can be found to be

$$y(t) = \mathcal{O}[e^{st}] = \int_{-\infty}^{\infty} h(\tau)e^{s(t-\tau)}d\tau = e^{st} \int_{-\infty}^{\infty} h(\tau)e^{-s\tau}d\tau = H(s)e^{st} \tag{5.86}$$

where $H(s)$ is the *transfer function* of the system, first defined in Eq.1.70 in Chapter 1, which is actually the Laplace transform of the impulse response $h(t)$ of the system:

$$H(s) = \int_{-\infty}^{\infty} h(t)e^{-st}dt \tag{5.87}$$

Note that Eq.5.86 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 $\sigma = 0$, i.e., $s = j\omega$, then the transfer function $H(s)$ becomes the Fourier transform of the impulse response $h(t)$ of the system:

$$H(s)\big|_{s=j\omega} = H(j\omega) = \int_{-\infty}^{\infty} h(t)e^{-j\omega t}dt = \mathcal{F}[h(t)] \tag{5.88}$$

This is the frequency response function of the LTI system first defined in Eq.3.217 of Chapter 3. If the input $x(t) = e^{j\omega_0 t}$ has a certain frequency ω_0, the response can be obtained by multiplying the input by the frequency response $H(j\omega)$ eval-

uated at $\omega = \omega_0$:

$$y(t) = H(j\omega_0)e^{j\omega_0 t} \tag{5.89}$$

Moreover, due to its convolution property of the Laplace transform, the convolution in Eq.5.84 can be converted to a multiplication in s-domain:

$$y(t) = h(t) * x(t) \xrightarrow{\mathcal{L}} Y(s) = H(s)X(s) \tag{5.90}$$

Based on this relationship the transfer function $H(s)$ can also be found in s-domain as the ratio of the output $Y(s)$ and input $X(s)$:

$$H(s) = \frac{Y(s)}{X(s)} \tag{5.91}$$

which can be used as an alternative 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.1.78):

  $$\int_{-\infty}^{\infty} \big|h(\tau)\big| d\tau < \infty \tag{5.92}$$

  i.e., the frequency response function $\mathcal{F}[h(t)] = H(j\omega) = H(s)\big|_{s=j\omega}$ 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 $\delta(t)$:

  $$h(t) = h(t)u(t) = \begin{cases} h(t) & t \geq 0 \\ 0 & t < 0 \end{cases} \tag{5.93}$$

  and its output is (Eq.1.79):

  $$y(t) = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau = \int_{0}^{\infty} h(\tau)x(t-\tau)d\tau \tag{5.94}$$

  We see that 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: $Re[p_k] < 0$.

Example 5.7: The transfer function of an LTI is

$$H(s) = \frac{1}{a+s} \tag{5.95}$$

As shown before, without specifying the ROC, this $H(s)$ could be the Laplace transform of one of the two possible impulse response functions:

- If ROC is $Re[s] > -a$, the system $h(t) = e^{-at}u(t)$ is causal.
  - If $a < 0$, i.e., imaginary axis $Re[s] = 0$ can be included in the ROC, the system is stable;
  - If $a > 0$, i.e., imaginary axis $Re[s] = 0$ cannot be included in the ROC, system is unstable;
- If ROC is $Re[s] < -a$, the system $h(t) = -e^{-at}u(-t)$ is anti-causal.
  - If $a < 0$, i.e., imaginary axis $Re[s] = 0$ cannot be included in the ROC, the system is unstable;
  - If $a > 0$, i.e., imaginary axis $Re[s] = 0$ can be included in the ROC, system is stable;

Example 5.8: The transfer function of an LTI is

$$H(s) = \frac{e^{s\tau}}{s+1} \quad Re[s] > -1 \tag{5.96}$$

Realizing that this is a time-shifted version of $\mathcal{L}[e^{-t}u(t)] = 1/(s+1)$, we can get the corresponding impulse response

$$h(t) = e^{-(t+\tau)}u(t+\tau) \tag{5.97}$$

As this $h(t)$ is not zero in time interval $-\tau < t < 0$, the system is not causal, although its ROC is a right half plane. This example serves as a counter example that not all right half plane ROC corresponds to causal system, while all causal systems' ROCs are right half planes. However, if $X(s)$ is rational, then the system is causal if and only if its ROC is a right half plane.

Many LTI systems can be characterized by a linear constant-coefficient differential equation (LCCDE):

$$\sum_{k=0}^{n} a_k \frac{d^k}{dt^k} y(t) = \sum_{k=0}^{m} b_k \frac{d^k}{dt^k} x(t) \tag{5.98}$$

Taking the Laplace transform of this equation, we get an algebraic equation:

$$Y(s)[\sum_{k=0}^{n} a_k s^k] = X(s)[\sum_{k=0}^{m} b_k s^k] \tag{5.99}$$

The transfer function of such a system is rational:

$$H(s) = \frac{Y(s)}{X(s)} = \frac{\sum_{k=0}^{m} b_k s^k}{\sum_{k=0}^{n} a_k s^k} = \frac{b_m}{a_n} \frac{\prod_{k=0}^{m}(s - z_k)}{\prod_{k=0}^{n}(s - p_k)} = \frac{N(s)}{D(s)} \tag{5.100}$$

where z_k, $(k = 1, 2, \cdots, m)$ are the roots of the numerator polynomial $N(s) = Y(s)$, and p_k, $(k = 1, 2, \cdots, n)$ are the roots of the denominator polynomial $D(s) = X(s)$, they are also respectively the zeros and poles of $H(s)$.

The output $y(t)$ of the LTI system can be found by solving the differential equation Eq.5.98. Alternatively, it can also be found first in s-domain as $Y(s) = H(s)X(s)$, and then in time domain by an inverse Laplace transform:

$$y(t) = \mathcal{L}^{-1}Y(s) = \mathcal{L}^{-1}[H(s)X(s)] \tag{5.101}$$

Obviously this approach is more convenient and therefore preferred as it is carried out algebraically without solving the original differential equation. This is one of the main applications of the Laplace transform. However, note that $y(t)$ obtained this way is only the particular solution due to the input $x(t)$, but the homogeneous solution due to initial conditions $y^{(k)}(0)$ cannot be obtained as the non-zero initial conditions are not represented by the bilateral Laplace transform. This problem will be addressed by the unilateral Laplace transform to be discussed later, by which the initial conditions will be taken into consideration.

The rational transfer function $H(s)$ in Eq.5.100 can be converted to a summation by partial fraction expansion:

$$H(s) = \frac{N(s)}{D(s)} = \frac{\sum_{k=0}^{m} b_k s^k}{\sum_{k=0}^{n} a_k s^k} = \sum_{k=1}^{n} \frac{c_k}{s - p_k} \tag{5.102}$$

The impulse response function can be found by inverse transform (the LTI system described by Eq.5.98 is causal):

$$h(t) = \mathcal{L}^{-1}[H(s)] = \mathcal{L}^{-1}\left[\sum_{k=1}^{n} \frac{c_k}{s - p_k}\right] = \sum_{k=1}^{n} c_k \mathcal{L}^{-1}\left[\frac{1}{s - p_k}\right] = \sum_{k=1}^{n} c_k e^{p_k t} u(t) \tag{5.103}$$

According to the fundamental theorem of algebra, if all coefficients a_k of the polynomial $D(s)$ are real, its solutions $p_k's$ are either real or complex conjugate pairs, corresponding to the following system behaviors in time domain:

- If at least one of the poles is on the right half s-plane, e.g., $Re[p_k] > 0$, then the corresponding term $c_k e^{p_k t} u(t)$ grows exponentially without bounds, and the system is unstable. In other words, for the system to be stable, all poles should be on the left half s-plane.
- If all poles are on the left half s-plane, $Re[p_k] < 0$ for all $1 < k < n$, then all terms in the summation above decay to zero exponentially so that $h(t)$ is absolutely integrable and the system is stable.
- If the system has two complex conjugate poles $p_{1,2} = \sigma \pm j\omega$, then we have:

$$e^{p_1 t} + e^{p_2 t} = e^{\sigma}[e^{j\omega t} + e^{-j\omega t}] = \frac{1}{2} e^{\sigma} \cos(\omega t) \tag{5.104}$$

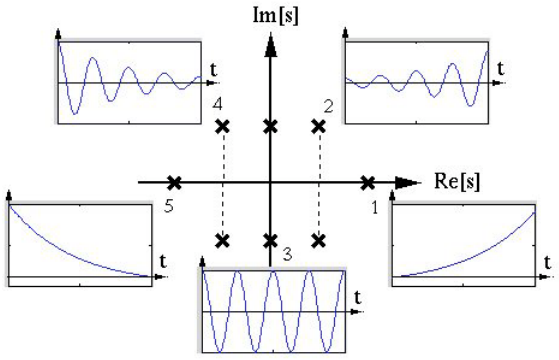

Figure 5.1 Different pole locations and the corresponding waveforms in time domain

This is sinusoid with frequency ω, which either decays exponentially if $\sigma < 0$, or grows exponentially if $\sigma > 0$, corresponding to either a stable or unstable system. When $\sigma = 0$, $p_{1,2}$ are on the imaginary axis and the system is marginally stable.

- If $0 < Re[p_2] \ll Re[p_1]$, then $e^{p_1 t}$ grows much more rapidly then $e^{p_2 t}$, i.e., the pole p_1 farther away from the origin dominates the behavior of the system more than the pole p_2 which is closer. On the other hand, if $Re[p_2] \ll Re[p_1] < 0$, then $e^{p_2 t} = e^{-|p_2|t}$ decays to zero much more rapidly then $e^{p_1 t}$, i.e., the pole p_1 closer to the origin dominates the behavior of the system more than the p_2 which is farther away. Based on this observation, it may be possible to estimate the behavior of the system based on only its dominant poles.

These different pole locations in s-plane and the corresponding waveforms in time domain are further illustrated in Fig.5.1 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 $H(j\omega) = H(s)\big|_{s=j\omega}$. This filter can also be determined based on the locations of the zeros and poles of its transfer function $H(s)$. The magnitude and phase of the frequency response function can

be written as:

$$|H(j\omega)| = \frac{\prod_{k=1}^{m}|j\omega - z_k|}{\prod_{k=1}^{n}|j\omega - p_k|} = \frac{\prod_{k=1}^{m}|\boldsymbol{u}_k|}{\prod_{k=1}^{n}|\boldsymbol{v}_k|}$$
$$\angle H(j\omega) = \frac{\sum_{k=1}^{m}\angle(j\omega - z_k)}{\sum_{k=1}^{n}\angle(j\omega - p_k)} = \frac{\sum_{k=1}^{m}\angle\boldsymbol{u}_k}{\sum_{k=1}^{n}\angle\boldsymbol{v}_k} \tag{5.105}$$

where each factor $\boldsymbol{u}_k = j\omega - z_k$ or $\boldsymbol{v}_k = j\omega - p_k$ is a vector in s-plane connecting a point $j\omega$ on the imaginary axis and each of the poles and zeros. Then the The filtering effects can be qualitatively determined by observing how $|H(j\omega)|$ and $\angle H(j\omega)$ change when ω increase along the imaginary axis from 0 toward ∞.

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.

5.1.6 First order system

In the transfer function $H(s)$ of a first order LTI system, the denominator polynomial $D(s)$ has an order $n = 1$, and $H(s)$ is conventionally written in the following *canonic form*:

$$H(s) = \frac{N(s)}{D(s)} = \frac{N(s)}{s - p} = \frac{N(s)}{s + 1/\tau} \tag{5.106}$$

where $p = -1/\tau$ is the root of denominator $D(s)$, and the pole of $H(s)$, and τ is the system parameter called the *time constant*. 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. Depending on the order m of the numerator $N(s)$, the system is expressed in either of the following two canonical forms:

1. $m = 0$: $N(s) = 1/\tau$ is a constant:

$$H(s) = \frac{1/\tau}{s + 1/\tau} = \frac{1}{s\tau + 1} \tag{5.107}$$

2. $m = 1$: $N(s) = s$:

$$H(s) = \frac{s}{s + 1/\tau} = \frac{s\tau}{s\tau + 1} \tag{5.108}$$

To illustrate the essential properties of the first order system, we reconsider the RC circuit in Example 3.4 in Chapter 3. 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:

$$RC\dot{y}(t) + y(t) = x(t), \qquad \text{i.e.,} \qquad \dot{y}(t) + \frac{1}{\tau}y(t) = \frac{1}{\tau}x(t) \tag{5.109}$$

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:

$$\left[s + \frac{1}{\tau}\right] Y(s) = \frac{1}{\tau} X(s), \qquad \text{i.e.,} \qquad H_C(s) = \frac{Y(s)}{X(s)} = \frac{1/\tau}{s + 1/\tau} \tag{5.110}$$

We can also get the impulse response $h_R(t)$ when $v_R(t)$ is treated as output based on Kirchhoff's voltage law:

$$h_R(t) = \delta(t) - h_C(t) \tag{5.111}$$

Taking the Laplace transform on both sides, we get:

$$H_R(s) = 1 - H_C(s) = 1 - \frac{1/\tau}{s + 1/\tau} = \frac{s}{s + 1/\tau} \tag{5.112}$$

We now consider both the impulse and step responses as well as the filtering effects of these first order systems.

1. **Impulse response function:**
   Taking the inverse Laplace transform on both sides of Eqs.5.110 and 5.112, we get:

   $$h_C(t) = \mathcal{L}^{-1}[H_C(s)] = \frac{1}{\tau} e^{-t/\tau} u(t) \tag{5.113}$$

   and

   $$h_R(t) = \mathcal{L}^{-1}[H_R(s)] = \delta(t) - \frac{1}{\tau} e^{-t/\tau} u(t) \tag{5.114}$$

2. **Step response:**
   The response of this systems to a step input $x(t) = u(t)$ or $X(s) = 1/s$ can also be found in s-domain as:

   $$Y(s) = H_C(s)X(s) = \frac{1/\tau}{s(s + 1/\tau)} = \frac{1}{s} - \frac{1}{s + 1/\tau} \tag{5.115}$$

   Taking inverse transform we get the step response:

   $$v_C(t) = y(t) = (1 - e^{-t/\tau})u(t) \tag{5.116}$$

   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:

   $$v_R(t) = u(t) - v_C(t) = u(t) - (1 - e^{-t/\tau})u(t) = e^{-t/\tau} u(t) \tag{5.117}$$

   The impulse and step response functions for the two first order systems are shown in Fig.5.2.
3. **First order systems as filters:**
   The filtering effects of the two first order systems are characterized by the magnitudes and phases of their frequency response functions $H(j\omega) = H(s)\big|_{s=j\omega}$:

   $$|H_C(j\omega)| = \left|\frac{1/\tau}{j\omega + 1/\tau}\right| = \frac{1}{\sqrt{(\omega\tau)^2 + 1}}, \qquad \angle H_C(j\omega) = -\angle(j\omega + 1/\tau) = -\tan^{-1}\omega\tau \tag{5.118}$$

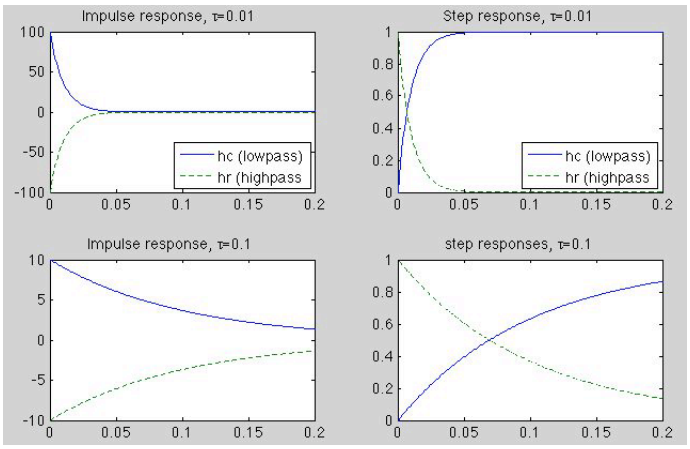

Figure 5.2 Impulse (left) and step (right) responses of first order systems

and

$$|H_R(j\omega)| = \left|\frac{j\omega}{j\omega + 1/\tau}\right| = \frac{\omega\tau}{\sqrt{(\omega\tau)^2+1}}, \qquad \angle H_R(j\omega) = \angle(j\omega\tau) - \angle(j\omega\tau + 1) = \frac{\pi}{2} - \tan^{-1}(\omega\tau) \tag{5.119}$$

Both the linear and Bode plots of the two systems are given in Fig.5.3, 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 respectively attenuate high and low frequencies, and are therefore correspondingly low and high-pass filters.

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

$$\frac{|H_C(j\omega_c)|^2}{|H_C(0)|^2} = \frac{1}{(\omega_c\tau)^2+1} = \frac{1}{2} \tag{5.120}$$

Solving for ω_c, we get the cutoff frequency $\omega_c = 1/\tau$. Equivalently we also have $|H(j\omega_c)| = 1/\sqrt{2} = 0.707$ and $Lm\, H(j\omega_c) = 20\log_{10}|H_C(j\omega_c)| \approx -3\, dB$.

The filtering effects can also be determined based on the locations of the zero and pole of the system. For each point $j\omega$ along the imaginary axis, we define two vectors corresponding to the zero $s_z = 0$ of $H_R(s)$ and the common pole of both $H_C(s)$ and $H_R(s)$ as shown in Fig.5.4:

$$\boldsymbol{u} = j\omega, \qquad \boldsymbol{v} = j\omega + 1/\tau \tag{5.121}$$

Now the magnitudes of $H_C(j\omega)$ and $H_R(j\omega)$ can be expressed as:

$$|H_C(j\omega)| = 1/\tau|\boldsymbol{v}|, \qquad |H_R(j\omega)| = |\boldsymbol{u}|/|\boldsymbol{v}| \tag{5.122}$$

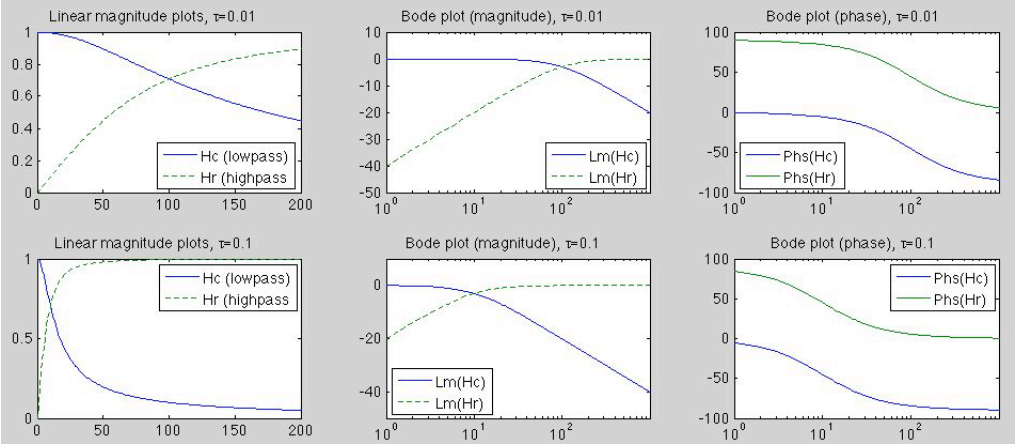

Figure 5.3 Filtering effects of first order systems

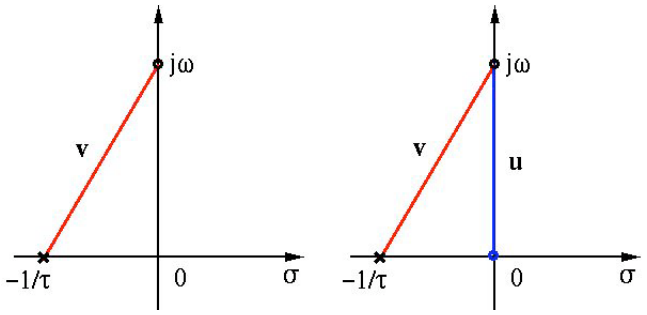

Figure 5.4 Qualitative determination of filtering behavior of first order systems

Based on the following two extreme cases:
– When $\omega = 0$, $|\boldsymbol{u}| = 0$ and $|\boldsymbol{v}| = 1/\tau$, we have $H_C(0) = 1$ and $H_R(0) = 0$;
– When $\omega = \infty$, $|\boldsymbol{v}| = |\boldsymbol{u}| = \infty$, we have $H_C(j\infty) = 0$ and $H_R(j\infty) = 1$.
we conclude that $H_C(j\omega)$ and $H_R(j\omega)$ are low and highpass filters, respectively.

5.1.7 Second order system

In the transfer function $H(s)$ of a second order LTI system, the denominator polynomial $D(s)$ has an order $n = 2$, and $H(s)$ is conventionally written in the following *canonic form*:

$$H(s) = \frac{N(s)}{D(s)} = \frac{N(s)}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{N(s)}{(s - p_1)(s - p_2)} \tag{5.123}$$

The order of the numerator $N(s)$ can be $m = 0$, $m = 1$, or $m = 2$, and ω_n and ζ in $D(s)$ are two system parameters called respectively *natural frequency*, which is always positive, and *damping coefficient*. The two poles p_1 and p_2 of $H(s)$ are

the two roots of the denominator quadratic function $D(s) = s^2 + 2\zeta\omega_n s + \omega_n^2$:

$$\begin{cases} p_1 = (-\zeta + \sqrt{\zeta^2 - 1})\omega_n = (-\zeta + j\sqrt{1-\zeta^2})\omega_n \\ p_2 = (-\zeta - \sqrt{\zeta^2 - 1})\omega_n = (-\zeta - j\sqrt{1-\zeta^2})\omega_n \end{cases} \tag{5.124}$$

If $|\zeta| \geq 1$, both poles are real, else when $|\zeta| < 1$, they form a complex conjugate pair located on a circle in the s-plane with radius ω_n:

$$p_{1,2} = (-\zeta \pm j\sqrt{1-\zeta^2})\omega_n = |p|\, e^{\pm j\angle p} = \omega_n e^{\pm j\phi} \tag{5.125}$$

where

$$|p| = \omega_n, \qquad \angle p = \tan^{-1}\left(\frac{\sqrt{1-\zeta^2}}{\zeta}\right) \tag{5.126}$$

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.5.5 from which we see that each of the two poles follows its own root locus when ζ moves from $-\infty$ to ∞:

- Locus of p_1: $\infty \Longrightarrow \omega_n \Longrightarrow j\omega_n \Longrightarrow -\omega_n \Longrightarrow 0$
- Locus of p_2: $0 \Longrightarrow \omega_n \Longrightarrow -j\omega_n \Longrightarrow -\omega_n \Longrightarrow -\infty$

The root locus is further summarized in the table below:

ζ	p_1, p_2	comments
$\zeta = -\infty$	∞, 0	
$-\infty < \zeta < -1$	$(-\zeta \pm \sqrt{\zeta^2 - 1})\omega_n$	real poles $0 < p_2 < p_1$
$\zeta = -1$	ω_n	repeated real poles $0 < p_1 = p_2 = \omega_n$
$-1 < \zeta < 0$	$(-\zeta \pm j\sqrt{1-\zeta^2})\omega_n$	complex conjugate pair in quadrants I, VI
$\zeta = 0$	$\pm j\omega_n$	imaginary poles
$0 < \zeta < 1$	$(-\zeta \pm j\sqrt{1-\zeta^2})\omega_n$	complex conjugate pair in quadrants II, III
$\zeta = 1$	$-\omega_n$	repeated real poles $p_1 = p_2 = -\omega_n < 0$
$1 < \zeta < \infty$	$(-\zeta \pm \sqrt{\zeta^2 - 1})\omega_n$	real poles $p_2 < p_1 < 0$
$\zeta = \infty$	0, $-\infty$	

We see that only when $\zeta > 0$, will the two poles 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 $N(s) = 1$

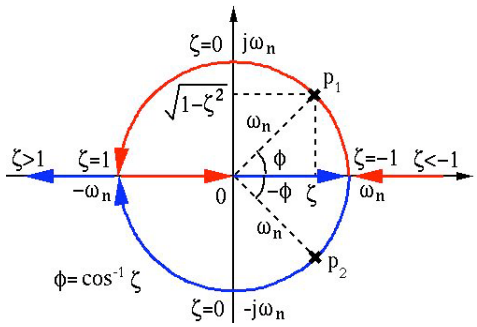

Figure 5.5 Root locus of the poles of a second order system

so that the transfer function is:

$$H(s) = \frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{1}{(s - p_1)(s - p_2)} \tag{5.127}$$

If $\zeta = \pm 1$, we have

$$H(s) = \frac{1}{s^2 \pm 2\omega_n s + \omega_n^2} = \frac{1}{(s \pm \omega_n)^2} \tag{5.128}$$

where $\pm\omega_n$ are repeated poles of $H(s)$, then we have:

$$H(s) = \frac{1}{(s \pm \omega_n)^2}, \quad \text{and} \quad h(t) = \mathcal{L}^{-1}[Y(s)] = t\, e^{\pm\omega_n t} u(t) \tag{5.129}$$

If $|\zeta| \neq 1$, then $p_1 \neq p_2$, and $H(s)$ can be written the following by partial fraction expansion:

$$H(s) = \frac{1}{p_1 - p_2}\left[\frac{1}{s - p_1} - \frac{1}{s - p_2}\right] \tag{5.130}$$

and the impulse response can be found by inverse Laplace transform:

$$h(t) = \mathcal{L}^{-1}[H(s)] = \frac{1}{p_1 - p_2}\left[e^{p_1 t} - e^{p_2 t}\right] u(t) = C\left[e^{p_1 t} - e^{p_2 t}\right] u(t) \tag{5.131}$$

where

$$C = \frac{1}{p_1 - p_2} = \frac{1}{2\omega_n\sqrt{\zeta^2 - 1}} = \frac{1}{2j\omega_n\sqrt{1 - \zeta^2}} \tag{5.132}$$

In the following we consider specifically how $h(t)$ varies when the value of ζ changes from $-\infty$ to ∞.

- $-\infty < \zeta < -1$, both poles $p_1 > 0$ and $p_2 > 0$ 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 $t \to \infty$, so does their difference:

$$h(t) = C\left[e^{p_1 t} - e^{p_2 t}\right] u(t), \quad (p_1 > p_2) \tag{5.133}$$

  The system is unstable.

- $\zeta = -1$, $p_1 = p_2 = -\zeta\omega_n = \omega_n$ are two repeated poles still on the right side of the s-plane. We have:

$$h(t) = t\, e^{\omega_n t} u(t) \tag{5.134}$$

which grows without bound when $t \to \infty$, the system stays unstable.

- $-1 < \zeta < 0$, now the two poles become a conjugate pair in quadrants I and IV, respectively:

$$p_{1,2} = (-\zeta \pm j\sqrt{1-\zeta^2})\omega_n = -\omega_n\zeta \pm j\omega_d \tag{5.135}$$

where

$$\omega_d = \omega_n\sqrt{1-\zeta^2} < \omega_n \tag{5.136}$$

is called the *damped natural frequency*. Now we have:

$$\begin{aligned} h(t) &= \frac{1}{2j\omega_n\sqrt{1-\zeta^2}} e^{-\zeta\omega_n t}[e^{j\omega_d t} - e^{-j\omega_d t}]u(t) \\ &= \frac{e^{-\zeta\omega_n t}}{\omega_d} sin(\omega_d t)u(t) \end{aligned} \tag{5.137}$$

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:

$$h(t) = \frac{1}{2j\omega_n}[e^{j\omega_n t} - e^{-j\omega_n t}]u(t) = \frac{1}{\omega_n}\ sin(\omega_n t)\ u(t) \tag{5.138}$$

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.5.86):

$$y(t) = H(j\omega)\big|_{\omega=\omega_n} e^{j\omega_n t} = \frac{1}{(j\omega)^2 + \omega_n^2}\bigg|_{\omega=\omega_n} e^{j\omega_n t} = \frac{e^{j\omega_n t}}{\omega_n^2 - \omega_n^2} = \infty \tag{5.139}$$

The response of the system becomes infinity, i.e., *resonance* occurs.

- $0 < \zeta < 1$, $p_{1,2} = (-\zeta \pm j\sqrt{1-\zeta^2})\omega_n$ 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 $h(t)$:

$$h(t) = \frac{e^{-\zeta\omega_n t}}{\omega_d} sin(\omega_d t)u(t) \tag{5.140}$$

However, as now $\zeta > 0$, 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.

$$h(t) = t\, e^{-\omega_n t} u(t) \tag{5.141}$$

- $1 < \zeta < \infty$, both poles $p_1 < 0$ and $p_2 < 0$ are on the real axis on the left of the s-plane, the impulse response is the difference of two exponentially decaying functions:

$$h(t) = C(e^{-|p_1|t} - e^{-|p_2|t})u(t), \quad (|p_1| < |p_2|) \tag{5.142}$$

which decays to zero in time. The system is *overdamped* and stable.

All seven cases considered above are summarized in the table below:

ζ	$H(s) = \frac{1}{s^2+2\zeta\omega_n s+\omega_n^2}$	$h(t) = C(e^{p_1 t} - e^{p_2 t})$	Comments on $h(t)$
$\zeta < -1$		$C(e^{p_1 t} - e^{p_2 t})u(t)$	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
$\zeta = 0$	$1/(s^2+\omega_n^2)$	$\frac{1}{\omega_n}\sin(\omega_n t)\ u(t)$	constant 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
$\zeta > 1$		$C(e^{-\lvert p_1\rvert t} - e^{-\lvert p_2\rvert t})u(t)$	exponential decay

These different impulse response functions $h(t)$ corresponding to different values of ζ are plotted in Fig.5.6. Note in particular the following two cases:

- $\zeta \ll -1$, we have $0 < p_2 \ll p_1$ and

$$h(t) = C(e^{p_1 t} - e^{p_2 t})u(t) \approx Ce^{p_1 t} \tag{5.143}$$

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

$$h(t) = C(e^{-|p_1|t} - e^{-|p_2|t})u(t) \approx Ce^{-|p_1|t} \tag{5.144}$$

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. A voltage $x(t)$ is applied as the input to the series combination of the three elements and the output $y(t)$ is the voltage across one of the three elements. The system can be described by a differential equation in time domain:

$$x(t) = v_L(t) + v_R(t) + v_C(t) = L\frac{d}{dt}\, i(t) + R\, i(t) + \frac{1}{C}\int_{-\infty}^{t} i(\tau)d\tau \tag{5.145}$$

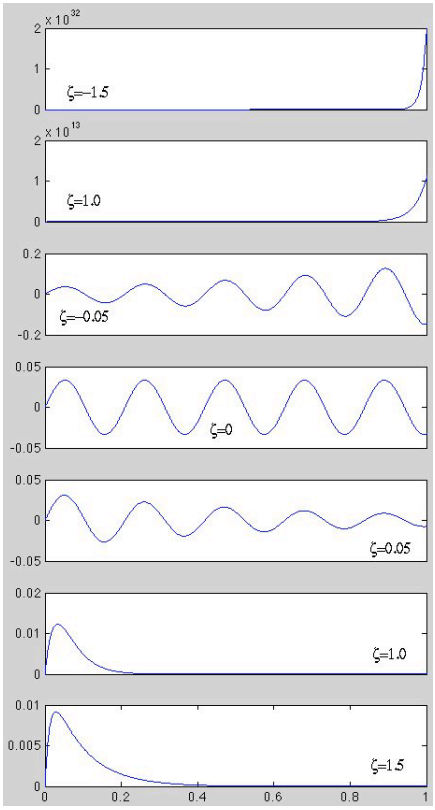

Figure 5.6 Impulse response of 2nd order system for different ζ

Taking Laplace transform on both sides, we get an algebraic equation in s-domain:

$$\begin{aligned} X(s) = V_L(s) + V_R(s) + V_C(s) &= \left[sL + R + \frac{1}{sC}\right] I(s) \\ &= [Z_L + Z_R + Z_C]I(s) = Z(s)I(s) \end{aligned}$$

Here Z_L, Z_R and Z_C are the *impedances* of the circuit elements L, R and C, respectively, defined as:

$$Z_L(s) = sL = \frac{V_L(s)}{I(s)}, \qquad Z_R = R = \frac{V_R(s)}{I(s)}, \qquad Z_C(s) = 1/sC = \frac{V_C(s)}{I(s)} \quad (5.146)$$

In general, the impedance of an element is the ratio $Z(s) = V(s)/I(s)$ of the voltage across and current through the element in s-domain (sometimes $s = j\omega$ with $\sigma = 0$), similar to the resistant $R = v(t)/i(t)$ defined by Ohm's law as the ratio between the voltage across and current through a resistor R in time domain. The total impedance $Z(s)$ of the three elements in series is the sum of the

individual impedances:

$$Z(s) = \frac{V(s)}{I(s)} = sL + R + \frac{1}{sC} = Z_L + Z_R + Z_C \tag{5.147}$$

	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	$V_C = I/Cs$	$V_R = RI$	$V_L = IsL$
impedance $Z = V/I$	$1/sC$	R	sL

The transfer function $H(s)$ is the ratio of the output and input voltages, where the output voltage across any one of the three elements (V_L, V_R, or V_C) can be found by treating the series circuit as a voltage divider:

- Output is voltage across the capacitor $v_C(t)$

$$H_C(s) = \frac{V_C(s)}{V(s)} = \frac{Z_C(s)}{Z(s)} = \frac{1/sC}{Ls + R + 1/sC} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{5.148}$$

- Output is voltage across the resistor $v_R(t)$

$$H_R(s) = \frac{V_R(s)}{V(s)} = \frac{Z_R(s)}{Z(s)} = \frac{R}{Ls + R + 1/sC} = \frac{2\zeta\omega_n s}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{5.149}$$

- Output is voltage across the inductor $v_L(t)$

$$H_L(s) = \frac{V_L(s)}{V(s)} = \frac{Z_L(s)}{Z(s)} = \frac{sL}{Ls + R + 1/sC} = \frac{s^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{5.150}$$

Here we have converted the denominator $D(s)$ into the canonical second order form:

$$D(s) = s^2 + (R/L)s + (1/LC) = s^2 + 2\zeta\omega_n s + \omega_n^2 \tag{5.151}$$

where the damping coefficient ζ and natural frequency ω_n are defined as:

$$\zeta = \frac{R}{2}\sqrt{\frac{C}{L}} > 0, \quad \omega_n = \frac{1}{\sqrt{LC}} > 0 \tag{5.152}$$

In the following, we further consider some important characteristics of the second order systems in both time and frequency domains.

1. **Impulse response function:**

$$h_C(t) = \mathcal{L}^{-1}[H_C(s)] = \omega_n^2 \mathcal{L}^{-1}\left[\frac{1}{(s - p_1)(s - p_2)}\right] \tag{5.153}$$

   Depending on the specific value of ζ ($0 < \zeta < 1$, $\zeta = 1$, or $\zeta > 1$), $h(t)$ takes one of the three forms in Eqs.5.140, 5.141, 5.142.

2. **Step response:**

In s-domain, the response to a step input $X(s) = \mathcal{L}[u(t)] = 1/s$ is:

$$\begin{aligned} Y(s) = H_C(s)X(s) &= \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}\,\frac{1}{s} \\ &= \frac{1}{p_1 p_2}\left[\frac{1}{s} - \frac{p_2}{p_2 - p_1}\frac{1}{s - p_1} + \frac{p_1}{p_2 - p_1}\frac{1}{s - p_2}\right] \end{aligned} \tag{5.154}$$

Specifically, if we assume $0 < \zeta < 1$, the two poles are:

$$p_{1,2} = (-\zeta \pm j\sqrt{1-\zeta^2})\omega_n = -\zeta\omega_n \pm j\omega_d \tag{5.155}$$

We also have:

$$p_1 p_2 = \omega_n^2, \qquad p_2 - p_1 = -2j\omega_n\sqrt{1-\zeta^2} \tag{5.156}$$

and the step response in time domain can be obtained by inverse transform:

$$\begin{aligned} y(t) = \mathcal{L}^{-1}[Y(s)] &= \frac{1}{\omega_n^2}\left[1 - \left(\frac{p_2}{p_2 - p_1}e^{p_1 t} - \frac{p_1}{p_2 - p_1}e^{p_2 t}\right)\right] \\ &= \frac{1}{\omega_n^2}\left[1 - \left(\frac{\zeta + j\sqrt{1-\zeta^2}}{2j\sqrt{1-\zeta^2}}e^{(-\zeta\omega_n + j\omega_d)t} - \frac{\zeta - j\sqrt{1-\zeta^2}}{2j\sqrt{1-\zeta^2}}e^{(-\zeta\omega_n - j\omega_d)t}\right)\right] \\ &= \frac{1}{\omega_n^2}\left[1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}}\left(\frac{\zeta + j\sqrt{1-\zeta^2}}{2j}e^{j\omega_d t} - \frac{\zeta - j\sqrt{1-\zeta^2}}{2j}e^{-j\omega_d t}\right)\right] \\ &= \frac{1}{\omega_n^2}\left[1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}}\left(\frac{e^{j\phi}e^{j\omega_d t} - e^{-j\phi}e^{-j\omega_d t}}{2j}\right)\right] \\ &= \frac{1}{\omega_n^2}\left[1 - \frac{e^{-\zeta\omega_n t}}{\sqrt{1-\zeta^2}}\sin(\omega_d t + \phi)\right] \end{aligned} \tag{5.157}$$

where angle $\phi = \tan^{-1}(\sqrt{1-\zeta^2}/\zeta)$ as defined in Eq.5.126. This step response function is plotted in Fig.5.7.

3. **Second order systems as filters:**
   The filtering effects of the three second order systems $H_C(s)$, $H_R(s)$ and $H_L(s)$ 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.5.8, based on assumed parameters $\omega_n = 2\pi 1000$ and $\zeta = 0.1$ (top) and $\zeta = 1/\sqrt{2} = 0.707$ (bottom). Here the magnitudes of the three frequency response functions are plotted in linear scale (left), together with their Bode plots of both magnitudes (middle) and phases (right). We see that when $\zeta = 0.1 < 0.707$, both $H_C(j\omega)$ and $H_L(j\omega)$ behave like a bandpass filter similar to $H_R(j\omega)$ (top row), but when $\zeta \geq 0.707$, they behave respectively as low-pass and high-pass filters without any peak (bottom row).
   Also, 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 the following vector one for each of the poles and

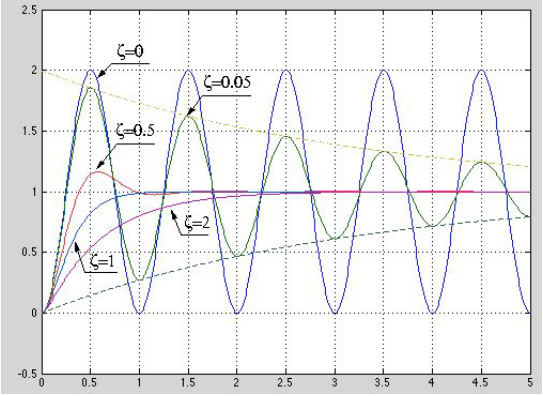

Figure 5.7 Step response of 2nd order system for different ζ

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.

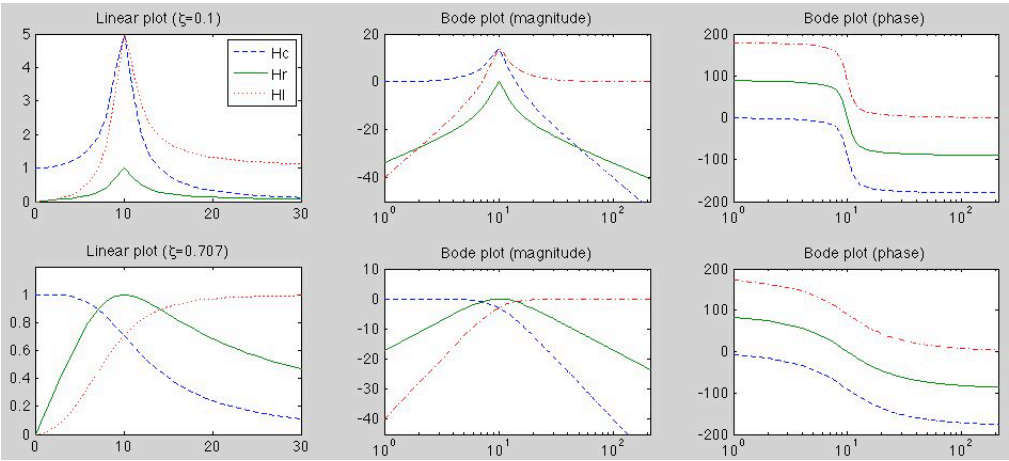

Figure 5.8 Linear and Bode plots of frequency response functions $H_C(j\omega)$, $H_R(j\omega)$, and $H_L(\omega)$

zeros in s-plane:

$$\boldsymbol{u} = j\omega, \quad \boldsymbol{v}_1 = j\omega - p_1, \quad \boldsymbol{v}_2 = j\omega - p_2 \tag{5.158}$$

and then observe how each of the three frequency response functions below changes when ω increase from 0 toward ∞, as illustrated in Fig.5.9:

– $H_C(s) = \omega_n^2 / D(s)$ with two poles but no zero:

$$|H_C(j\omega)| = \frac{\omega_n^2}{|j\omega - p_1||j\omega - p_2|} = \frac{\omega_n^2}{|\boldsymbol{v}_1||\boldsymbol{v}_2|} \tag{5.159}$$

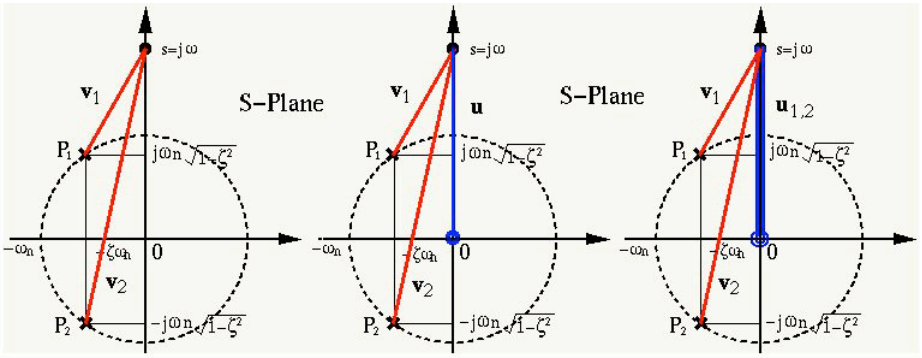

Figure 5.9 Qualitative determination of filtering behavior of second order systems

which is some constant when $\omega = 0$ but approaches 0 when $\omega \to \infty$ causing both $|\boldsymbol{v}_1| \to \infty$ and $|\boldsymbol{v}_2| \to \infty$, 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:

$$|H_R(j\omega)| = \frac{2\zeta\omega_n\,|j\omega|}{|j\omega - p_1||j\omega - p_2|} = \frac{2\zeta\omega_n\,|\boldsymbol{u}|}{|\boldsymbol{v}_1||\boldsymbol{v}_2|} \tag{5.160}$$

which is zero when $\omega = 0$ or $\omega \to \infty$, 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 $\boldsymbol{u}_1 = \boldsymbol{u}_2 = \boldsymbol{u}$):

$$|H_L(j\omega)| = \frac{|j\omega|^2}{|j\omega - p_1||j\omega - p_2|} = \frac{|\boldsymbol{u}|^2}{|\boldsymbol{v}_1||\boldsymbol{v}_2|} \tag{5.161}$$

which is zero when $\omega = 0$, but approaches constant 1 when $\omega \to \infty$, 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 $u = (\omega/\omega_n)^2$ (where ω/ω_n can be considered as the frequency normalized by the natural frequency ω_n) so that the squared magnitudes of the frequency response functions can be expressed as:

$$\begin{aligned}
|H_C(j\omega)|^2 &= \left|\frac{\omega_n^2}{(j\omega)^2 + 2\zeta\omega_n j\omega + \omega_n^2}\right|^2 = \frac{1}{(u-1)^2 + 4\zeta^2 u} \\
|H_R(j\omega)|^2 &= \left|\frac{2\zeta\omega_n j\omega}{(j\omega)^2 + 2\zeta\omega_n j\omega + \omega_n^2}\right|^2 = \frac{4\zeta^2 u}{(u-1)^2 + 4\zeta^2 u} \\
|H_L(j\omega)|^2 &= \left|\frac{(j\omega)^2}{(j\omega)^2 + 2\zeta\omega_n j\omega + \omega_n^2}\right|^2 = \frac{u^2}{(u-1)^2 + 4\zeta^2 u}
\end{aligned} \tag{5.162}$$

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:

$$\begin{cases} u_{p_C} = 1 - 2\zeta^2 \\ u_{p_R} = 1 \\ u_{p_L} = 1/(1-2\zeta^2) \end{cases} \quad \text{i.e.} \quad \begin{cases} \omega_{p_C} = \omega_n\sqrt{u_{p_C}} = \omega_n\sqrt{1-2\zeta^2} \\ \omega_{p_R} = \omega_n\sqrt{u_{p_R}} = \omega_n \\ \omega_{p_L} = \omega_n\sqrt{u_{p_L}} = \omega_n/\sqrt{1-2\zeta^2} \end{cases} \tag{5.163}$$

We see that the three peak frequencies are different:

$$\omega_{C_p} \le \omega_{R_p} \le \omega_{L_p} \tag{5.164}$$

Substituting these peak frequencies into Eq.5.162, we get the peak values of the three filters:

$$\begin{aligned} |H_{max_R}| &= |H(j\omega_{p_R})| = 1 \\ |H_{max_C}| &= |H(j\omega_{p_C})| = |H_{max_L}| = |H(j\omega_{p_L})| = \frac{1}{2\zeta\sqrt{1-\zeta^2}} \end{aligned} \tag{5.165}$$

Also note that for $H_C(j\omega)$ and $H_L(j\omega)$ to behave as a band-pass filter that reaches a peak value at respectively ω_{p_C} and ω_{p_L} given in Eq.5.163, ζ has to satisfy the following (for the peak frequency to be real):

$$1 - 2\zeta^2 > 0, \qquad \text{i.e.,} \quad \zeta < 1/\sqrt{2} = 0.707 \tag{5.166}$$

If $\zeta \ge 1/\sqrt{2}$ this condition is not satisfied, then $|H_C(j\omega)|$ is a low-pass filter that reaches its maximum of 1 at $\omega = 0$, and $|H_L(j\omega)|$ is a high-pass filter that reaches its maximum of 1 at $\omega = \infty$. These two cases have already been illustrated in Fig.5.8.

5. **Bandwidth of second order bandpass filter:**
In general, 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:

$$|H(j\omega_1)|^2 = |H(j\omega_2)|^2 = \frac{1}{2}|H(j\omega_p)|^2 = \frac{1}{2}|H_{max}|^2 \tag{5.167}$$

Specifically, for the bandpass filter $H_R(j\omega)$, $H_{max}(0) = 1$, and at the two cutoff frequencies we have:

$$\frac{|H_R(j\omega)|^2}{|H_{max_R}|^2} = |H_R(j\omega)|^2 = \frac{4\zeta^2 u}{(u-1)^2 + 4\zeta^2 u} = \frac{1}{2} \tag{5.168}$$

Solving this quadratic equation we get two solutions:

$$u_{1,2} = 1 + 2\zeta^2 \pm 2\zeta\sqrt{1+\zeta^2} \tag{5.169}$$

the corresponding cutoff frequencies are:

$$\omega_{1,2} = \omega_n\sqrt{1 + 2\zeta^2 \pm 2\zeta\sqrt{1+\zeta^2}} \tag{5.170}$$

and the bandwidth is:

$$\Delta\omega_R = \omega_1 - \omega_2 = 2\zeta\omega_n \tag{5.171}$$

Based on this result, the denominator of the second order transfer function can also be written as:

$$D(s) = s^2 + \Delta\omega_n s + \omega_n^2 \tag{5.172}$$

5.1.8 The Unilateral Laplace Transform

The bilateral Laplace transform can be applied to left-sided signals (or anti-causal systems) as well as right-sided ones (or causal systems). Also, we have seen that it is a convenient tool for solving differential equations (LCCDEs). However, the bilateral Laplace transform also has some drawbacks, e.g., it can only find the particular solutions of a differential equations, as the initial conditions 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 solution due to non-zero initial conditions as well as the particular solution due to the input on the right-hand side of the equation.

The unilateral Laplace transform of a given signal $x(t)$ is defined as

$$\mathcal{U}L[x(t)] = X(s) = \int_0^\infty x(t)e^{-st}dt \tag{5.173}$$

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, the all poles have to be on the left half s-plane, i.e., the ROC is always in the right half s-plane. Obviously, the unilateral Laplace transform of any signal $x(t) = x(t)u(t)$ is identical to its bilateral Laplace transform. However, when $x(t) \neq x(t)u(t)$, the two Laplace transforms are different.

The unilateral Laplace Transform shares all of the properties of the bilateral Laplace transform, although some of the properties 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**

$$\mathcal{U}L[\frac{d}{dt}x(t)] = \mathcal{U}L[\dot{x}(t)] = sX(s) - x(0) \tag{5.174}$$

Proof:

$$\begin{aligned}\mathcal{U}L[\frac{d}{dt}x(t)] &= \int_0^\infty \dot{x}(t)e^{-st}dt = \int_0^\infty e^{-st}d[x(t)] = x(t)e^{-st}\big|_0^\infty - \int_0^\infty x(t)d(e^{-st}) \\ &= -x(0) + s\int_0^\infty x(t)e^{-st}dt = sX(s) - x(0)\end{aligned} \tag{5.175}$$

We can further get the transform of the 2nd derivative of $x(t)$:

$$\mathcal{U}L[\ddot{x}(t)] = s\mathcal{U}L[\dot{x}(t)] - \dot{x}(0) = s^2X(s) - sx(0) - \dot{x}(0) \tag{5.176}$$

and in general we have:

$$\mathcal{U}L[x^{(n)}(t)] = s^n X(s) - \sum_{k=0}^{n-1} s^k x^{(n-1-k)}(0) \tag{5.177}$$

- **The initial value theorem:**
  If a right-sided signal $x(t)$ containing no impulse or higher order singularities at $t = 0$, its initial value $x(0^+)$ ($t \to 0$ from $t > 0$) can be found to be:

$$x(0^+) = \lim_{t\to 0} x(t) = \lim_{s\to\infty} sX(s) \tag{5.178}$$

  Proof: Consider the Laplace transform of a time derivative:

$$\begin{aligned}\mathcal{L}[\frac{d}{dt}x(t)] &= \int_0^\infty [\frac{d}{dt}x(t)]e^{-st}dt = x(t)e^{-st}|_0^\infty - \int_0^\infty x(t)d(e^{-st}) \\ &= -x(0) + s\int_0^\infty x(t)e^{-st}dt = sX(s) - x(0)\end{aligned} \tag{5.179}$$

  At the limit $s \to 0$, the equation becomes:

$$\lim_{s\to 0}\int_0^\infty \frac{d}{dt}x(t)e^{-st}dt = \int_0^\infty dx(t) = x(\infty) - x(0) = \lim_{s\to 0}[sX(s) - x(0)] \tag{5.180}$$

  i.e.,

$$\lim_{s\to 0} sX(s) = x(\infty) \tag{5.181}$$

- **The final value theorem:**
  If a right-sided signal $x(t)$ has a finite limit as $t \to \infty$, this final value can also be found to be:

$$x(\infty) = \lim_{t\to\infty} x(t) = \lim_{s\to 0} sX(s) \tag{5.182}$$

  Proof: At the limit $s \to \infty$, Eq.5.179 becomes:

$$\lim_{s\to\infty}\int_0^\infty \frac{d}{dt}x(t)e^{-st}dt = 0 = \lim_{s\to\infty}[sX(s) - x(0)] \tag{5.183}$$

  i.e.,

$$\lim_{s\to\infty} sX(s) = x(0) \tag{5.184}$$

Due to these properties, the unilateral Laplace transform is a powerful tool for solving LCCDEs with non-zero initial conditions.

Example 5.9: We consider Example 3.4 in Chapter 3 one more time, where the LCCDE of the first order system is:

$$\tau\dot{y}(t) + y = x \tag{5.185}$$

Taking the unilateral Laplace transform on both sides, we get:

$$\tau(s-y_0)Y(s)=X(s),\quad \text{i.e.}\quad Y(s)=\frac{X(s)}{s\tau+1}+\frac{\tau y_0}{s\tau+1} \tag{5.186}$$

where $y_0=y(t)$ is the initial condition. Now we consider two cases:

- When $x(t)=u(t)$, i.e., $X(s)=1/s$, the output is:

$$Y(s)=\frac{1}{s}\frac{1}{s\tau+1}+\frac{\tau y_0}{s\tau+1}=\frac{1}{s}-\frac{\tau}{s\tau+1}+\frac{\tau y_0}{s\tau+1} \tag{5.187}$$

  Taking inverse transform we get:

$$y(t)=[1+(y_0-1)e^{-t/\tau}]u(t)=(1-e^{-t/\tau})u(t)+y_0e^{-t/\tau}u(t) \tag{5.188}$$

  These two terms represent respectively the charge of the capacitor C by the input $x(t)=u(t)$ and the discharge of the initial voltage $y(0)=y_0$ (superposition property of linear systems).
- When $x(t)=\delta(t)$, i.e., $X(s)=1$, the output is:

$$Y(s)=\frac{1}{s\tau+1}+\frac{\tau y_0}{s\tau+1} \tag{5.189}$$

  Taking inverse transform we get:

$$y(t)=\left[\frac{1}{\tau}+y_0\right]e^{-t/\tau}u(t) \tag{5.190}$$

  This result represents the discharge of the total voltage across the capacitor, including the instantly charged voltage $u(t)/\tau$ and the initial voltage $y(0)=y_0$. In particular, under zero initial condition $y(0)=y_0=0$, we get the impulse response:

$$y(t)\big|_{y_0=0}=h(0)=\frac{1}{\tau}e^{-t/\tau}u(t) \tag{5.191}$$

All these results are consistent with Example 3.4.

Example 5.10: A 2nd-order system is described by this LCCDE:

$$\frac{d^2}{dt^2}y(t)+3\frac{d}{dt}y(t)+2y(t)=x(t)=\alpha u(t) \tag{5.192}$$

with initial conditions

$$y(0)=\beta,\quad \dot{y}(0)=\gamma \tag{5.193}$$

To find $y(t)$, we first apply unilateral Laplace transform to the differential equation to get

$$s^2Y(s)-\beta s-\gamma+3sY(s)-3\beta+2Y(s)=(s^2+3s+2)Y(s)-\beta s-\gamma-3\beta=\alpha/s \tag{5.194}$$

Solving this algebraically for $Y(s)$ we get:

$$Y(s) = \frac{\alpha}{s(s+1)(s+2)} + \frac{\beta(s+3)}{(s+1)(s+2)} + \frac{\gamma}{(s+1)(s+2)} = Y_p(s) + Y_h(s) \tag{5.195}$$

This is the general solution of the LCCDE which is composed of two parts:

- **The homogeneous (zero-input) solution** due to the nonzero initial conditions $y(0) \neq 0$ and $\dot{y}(0) \neq 0$ with zero input $x(t) = 0$:

$$Y_h(s) = \frac{\beta(s+3)}{(s+1)(s+2)} + \frac{\gamma}{(s+1)(s+2)} \tag{5.196}$$

- **The particular (zero-state) solution** due to the nonzero input $x(t) \neq 0$ but with zero initial conditions $y(0) = \dot{y}(0) = 0$:

$$Y_p(s) = \frac{\alpha}{s(s+1)(s+2)} \tag{5.197}$$

Given specific values for α, β and γ such as $\alpha = 2$, $\beta = 3$ 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 Laplace transform on both sides we get the solution in time domain solution:

$$y_p(t) = \mathcal{U}L^{-1}[Y_p(s)] = \mathcal{U}L^{-1}\left[\frac{1}{s} - \frac{2}{s+1} + \frac{1}{s+2}\right] = [1 - 2e^{-t} + e^{-2t}]u(t)$$

$$y_h(t) = \mathcal{U}L^{-1}[Y_h(s)] = \mathcal{U}L^{-1}\left[\frac{1}{s+1} + \frac{2}{s+2}\right] = [e^{-t} + 2e^{-2t}]u(t)$$

and

$$y(t) = y_h(t) + y_p(t) = [1 - e^{-t} + 3e^{-2t}]u(t) \tag{5.198}$$

If bilateral Laplace transform is applied to the same LCCDE, we get

$$s^2Y(s) + 3sY(s) + 2Y(s) = (s^2 + 3s + 2)Y(s) = \frac{\alpha}{s} = \frac{2}{s} \tag{5.199}$$

Solving this for $Y(s)$ and taking inverse transform, we get:

$$Y(s) = \frac{2}{s(s+1)(s+2)}, \qquad y(t) = [e^{-t} + 2e^{-2t}]u(t) \tag{5.200}$$

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.

5.2 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 similar to and in parallel with the previous discussions for the Laplace transform, with the only essential difference that all signals and systems considered here are discrete in time.

5.2.1 From Discrete Time Fourier Transform to Z-Transform

The Z-transform of a discrete signal $x[m]$ can be considered as the generalization of the discrete-time Fourier transform (DTFT) of the signal:

$$\mathcal{F}[x[m]] = \sum_{m=-\infty}^{\infty} x[m]e^{-jm\omega} = X(e^{j\omega}) \tag{5.201}$$

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. Note that this transform is based on the underlying assumption that the signal $x[m]$ is square summable so that the summation converges and $X(e^{j\omega})$ exists. However, this assumption is not true for signals such as $x[m] = m$, $x[m] = m^2$, and $x[m] = e^{am}$, all of which grow without a bound when $|m| \to \infty$, and are not square summaable. In such cases, we could still consider the Fourier transform of a modified version of the signal $x'[m] = x[m]e^{-\sigma m}$, where $e^{-\sigma m}$ is an exponential factor with a real parameter σ, which can force the given signal $x[m]$ to decay exponentially for some properly chosen value of σ (either positive or negative). For example, $x[m] = e^{am}u[m]$ $(a > 0)$ does not converge when $m \to \infty$, therefore its Fourier spectrum does not exist. However, if we choose $Re[s] = \sigma > a$, the modified version $e^{-(\sigma-a)m}u[m]$ will converge as $m \to \infty$.

In general, the Fourier transform of the modified signal is:

$$\mathcal{F}[x'[m]] = \sum_{m=-\infty}^{\infty} x[m]e^{-\sigma m}e^{jm\omega} = \sum_{m=-\infty}^{\infty} x[m]e^{-m(\sigma+j\omega)} = \sum_{m=-\infty}^{\infty} x[m]z^{-m} \tag{5.202}$$

where z is a complex variable defined as

$$z = e^s = e^{\sigma+j\omega} = e^{\sigma}\, e^{j\omega} = |z|\angle z \tag{5.203}$$

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 $x[m]$, formally defined as:

$$X(z) = \mathcal{Z}[x[m]] = \sum_{m=-\infty}^{\infty} x[m]z^{-m} \tag{5.204}$$

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 in the z-plane, composed of all z values that guarantee the convergence of the summation in Eq. 5.204. Due to the introduction of the exponential decay factor $e^{-\sigma m}$, 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 $x[m]$. We see that the 1-D Fourier spectrum $X(e^{j\omega})$ of the discrete signal $x[m]$ 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 $z = e^{j\omega}$:

$$\mathcal{F}[x[m]] = \mathcal{Z}[x[m]]\big|_{z=e^{j\omega}} = X(z)\big|_{z=e^{j\omega}} = X(e^{j\omega}) \tag{5.205}$$

This is the reason why sometimes the discrete-time Fourier spectrum is also denoted by $X(e^{j\omega})$.

Given the Z-transform $X(z) = \mathcal{Z}[x[m]]$, the time signal $x[m]$ can be found by the inverse Z-transform, which can be derived from the corresponding Fourier transform of discrete signals:

$$\mathcal{Z}[x[m]] = X(z) = X(e^{\sigma+j\omega}) = \sum_{m=-\infty}^{\infty} x[m]e^{-(\sigma+j\omega)m} = \mathcal{F}[x[m]e^{-\sigma m}] \tag{5.206}$$

Taking the inverse Fourier transform of the above, we get

$$x[m]e^{-m\sigma} = \mathcal{F}^{-1}[X(e^{\sigma+j\omega})] = \frac{1}{2\pi}\int_0^{2\pi} X(e^{\sigma+j\omega})e^{jm\omega}d\omega \tag{5.207}$$

Multiplying both sides by $e^{m\sigma}$, we get:

$$x[m] = \frac{1}{2\pi}\int_0^{2\pi} X(e^{\sigma+j\omega})e^{(\sigma+j\omega)m}d\omega \tag{5.208}$$

To further represent the inverse Z-transform in terms of z (instead of ω), we note

$$dz = d(e^{\sigma+j\omega}) = e^{\sigma}je^{j\omega}d\omega = jzd\omega, \quad \text{i.e.,} \quad d\omega = z^{-1}dz/j \tag{5.209}$$

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}:

$$x[m] = \frac{1}{2\pi}\oint X(z)z^m z^{-1}dz/j = \frac{1}{2\pi j}\oint X(z)z^{m-1}dz \tag{5.210}$$

Now we get the forward and inverse Z-transform pair:

$$X(z) = \mathcal{Z}[x[m]] = \sum_{m=-\infty}^{\infty} x[m]z^{-m}$$
$$x[m] = \mathcal{Z}^{-1}[X(z)] = \frac{1}{2\pi j}\oint X(z)z^{m-1}dz \tag{5.211}$$

which can also be more concisely represented as

$$x[m] \overset{\mathcal{Z}}{\longleftrightarrow} X(z) \tag{5.212}$$

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.5.203, 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 $Re[s] = \sigma = 0$ 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 $Re[s] = \sigma = -\infty$ in the s-plane is mapped to the origin $|z| = e^{\sigma} = 0$ in the z-plane;
- The vertical line corresponding to $Re[s] = \sigma = \infty$ 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 $x[m]$, the corresponding discrete-time Fourier spectrum $X(e^{j\omega}) = \mathcal{F}[x[m]]$ 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 some applications the Z-transform takes the form of a rational function as a ratio of two polynomials:

$$X(z) = \frac{N(z)}{D(z)} = \frac{\sum_{k=0}^{m} b_k z^k}{\sum_{k=0}^{n} a_k z^k} = \frac{b_m}{a_n} \frac{\prod_{k=1}^{m}(z - z_k)}{\prod_{k=1}^{n}(z - p_k)} \tag{5.213}$$

where the roots $z_k, (k = 1, 2, \cdots, m)$ of the numerator polynomial $N(z)$ of order m are the zeros of $X(z)$, and the roots $p_k, (k = 1, 2, \cdots, n)$ of the denominator polynomial $D(z)$ of order n are the poles of $X(z)$. Some of these roots may be repeated. Obviously we have:

$$X(z_k) = 0, \quad \text{and} \quad X(p_k) = \infty \tag{5.214}$$

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 expand $X(z)$ into multiple terms so that $m < n$ is true for each term. Same as in the case of the Laplace transform, the locations of the zeros and poles of $X(z)$ characterize some essential properties of a signal $x[m]$.

5.2.2 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 5.1 of the Laplace transform:

Example 5.11:

1. A right-sided discrete signal $x[m] = a^{-m}u[m]$:

$$X(z) = \sum_{m=-\infty}^{\infty} x[m]z^{-m} = \sum_{m=0}^{\infty} (az)^{-m} \tag{5.215}$$

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 summation of the Z-transform above can be further written as:

$$X(z) = \sum_{m=0}^{\infty} (az^{-1})^m = \frac{1}{1 - (az)^{-1}} \quad \text{if } |z| > 1/|a| \tag{5.216}$$

Specially when $a = 1$, we have $x[m] = u[m]$ and

$$U(z) = \mathcal{Z}[u[m]] = \frac{1}{1 - z^{-1}}, \quad \text{if } |z| > 1 \tag{5.217}$$

If we let $Re[s] = \sigma \to 0$, i.e., $|z| = 1$, $U(z)$ will be evaluated along the unit circle $z = e^{j\omega}$ and become $\mathcal{Z}[u[m]] = 1/(1 - e^{-j\omega})$, which is seemingly the

Fourier spectrum of $u(t)$. 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 $u[m]$ in Eq.4.18:

$$\mathcal{F}[u[m]] = \frac{1}{1 - e^{-j2\pi f}} + \frac{1}{2} \sum_{n=-\infty}^{\infty} \delta(f - n) \tag{5.218}$$

we see that an extra term $\sum_{n=-\infty}^{\infty} \delta(f - n)/2$ in the Fourier spectrum which reflects the fact that the summation is only marginally convergent when $|z| = 1$.

2. A left-sided signal $x[m] = -a^{-m}u[-m-1]$:

$$X(z) = -\sum_{n=-\infty}^{\infty} a^{-m}u[-m-1]z^{-m} = -\sum_{m=-\infty}^{-1} (az)^{-m} = 1 - \sum_{n=0}^{\infty}(az)^n \tag{5.219}$$

where we have assumed $n = -m$. We see that only when $|az| < 1$, i.e., z is inside the circle $|z| < 1/|a|$ will this summation converge and $X(z)$ exist:

$$X(z) = 1 - \frac{1}{1 - az} = \frac{1}{1 - (az)^{-1}}, \quad \text{if } |z| < 1/|a| \tag{5.220}$$

3. A two-sided signal $x[m] = a^{-|m|} = a^{-m}u[m] + a^m u[-m-1]$:
The transform of this signal is the sum of the transforms of the two individual terms. According to the results in the previous two cases, we get:

$$X(z) = \frac{1}{1 - (az)^{-1}} - \frac{1}{1 - az^{-1}}, \quad |z| > 1/|a|, \;\; |z| < |a| \tag{5.221}$$

provided the intersection of the two individual ROCs is non-empty, i.e., $1/|a| < |z| < |a|$, which is possible only if $|a| > 1$, i.e., $x[m]$ decays when $|m| \to \infty$. However, if $|a| < 1$, the intersection of the two ROCs is an empty set, and the Z-transform does not exist, reflecting the fact that $x[m]$ grows without bound when $|m| \to \infty$.

Based on the examples above we summarize a set of properties of the ROC:

- If a signal $x[m]$ 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 concen-

tric 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 $x[m]$ 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 $X(z) = \mathcal{Z}[x[m]]$ dictate the ROC and thereby the most essential properties of the corresponding signal $x[m]$, such as whether it is right or left-sided, whether it grows or decays over time. Moreover, the zeros and poles of the transfer function $H(z) = \mathcal{Z}[h[m]]$ of an LTI system dictate its stability and filtering effects. All such properties and behaviors can be qualitatively characterized based on the locations of the zeros and poles of in the z-plane, as we will see in the later discussions.

Example 5.12: Find the time signal corresponding to the following Z-transform:

$$X(z) = \frac{1}{(1 - \frac{1}{3}z^{-1})(1 - 2z^{-1})} = -\frac{1/5}{1 - \frac{1}{3}z^{-1}} + \frac{6/5}{1 - 2z^{-1}} \tag{5.222}$$

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:

$$x[m] = -\frac{1}{5}(\frac{1}{3})^m u[m] + \frac{1}{5}(\frac{1}{3})^m u[m] \tag{5.223}$$

- $|z| < 1/3$: The ROC is inside the innermost pole $p_1 = 1/3$, both terms of $X(z)$ correspond to left-sided time functions:

$$x[m] = \frac{1}{5}(\frac{1}{3})^m u[-m-1] - \frac{1}{5}(\frac{1}{3})^m u[-m-1] \tag{5.224}$$

- $1/3 < |z| < 2$: 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:

$$x[m] = -\frac{1}{5}(\frac{1}{3})^m u[m] - \frac{1}{5}(\frac{1}{3})^m u[-m-1] \tag{5.225}$$

In particular, note that only the last ROC includes the circle $|z| = 1$ and the corresponding time function $x[m]$ has a discrete Fourier transform. Fourier transform of the other two functions do not exist.

Example 5.13:

- Let $x[m] = 0$ for all m except $m = -1, 0, 1$, then

$$X(z) = \sum_{m=-1}^{1} z^{-m} = \frac{1}{z} + 1 + z \tag{5.226}$$

As $z^{-m} = \infty$ when $z = \infty$ or $z = 0$, neither of these two z values is included in the ROC.

-

$$x[m] = b^{|m|} = b^m u[m] + b^{-m} u[-m-1] \tag{5.227}$$

For the right-sided part:

$$\mathcal{Z}[b^m u[m]] = \sum_{m=0}^{\infty} b^m z^{-m} = \sum_{m=0}^{\infty} (bz^{-1})^m = \frac{1}{1 - bz^{-1}} \quad |z| > b \tag{5.228}$$

For the left-sided part:

$$\begin{aligned} \mathcal{Z}[b^{-m} u[-m-1]] &= \sum_{m=-\infty}^{-1} b^{-m} z^{-m} = \sum_{m=0}^{\infty} (bz)^m - 1 \\ &= \frac{1}{1 - bz} - 1 = \frac{-1}{1 - (bz)^{-1}} \quad |z| < 1/b \end{aligned}$$

The ROC for both parts combined is the intersection of the individual ROCs:

$$b < |z| < 1/b \tag{5.229}$$

When $b < 1$, $x[m]$ decays on both sides as $m \to \infty$ and its ROC is a ring. But when $b > 1$, $x[m]$ 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.

5.2.3 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 ommitted 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:

$$\mathcal{Z}[x[m]] = X(z), \qquad \mathcal{Z}[y[m]] = Y(z) \tag{5.230}$$

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**

$$\mathcal{Z}[ax[m] + by[m]] = aX(z) + bY(z) \quad ROC \supseteq (R_x \cap R_y) \tag{5.231}$$

Similar to the case of the Laplace transform, the ROC of the linear combination of $x[m]$ and $y[m]$ may be larger than the intersection of their individual ROCs $R_x \cap R_y$, due to reasons such as zero-pole cancellation.

- **Time shifting**

$$\mathcal{Z}[x[m-m_0]] = z^{-m_0}X(z), \quad ROC = R_x \tag{5.232}$$

- **Time reversal**

$$\mathcal{Z}[x[-m]] = X(1/z), \qquad ROC = 1/R_x \tag{5.233}$$

Proof:

$$\mathcal{Z}[x[-m]] = \sum_{m=-\infty}^{\infty} x[-m]z^{-m} = \sum_{n=-\infty}^{\infty} x[n](\frac{1}{z})^{-n} = X(1/z) \tag{5.234}$$

where $n = -m$.

- **Modulation**

$$\mathcal{Z}[(-1)^m x[m]] = X(-z) \tag{5.235}$$

Here modulation means every other sample of the signal is negated.
Proof:

$$\mathcal{Z}[(-1)^m x[m]] = \sum_{m=-\infty}^{\infty} x[m](-1)^m z^{-m} = \sum_{m=-\infty}^{\infty} x[m](-z)^{-m} = X(-z) \tag{5.236}$$

- **Down-sampling**

$$\mathcal{Z}[x_{(2)}[m]] = \frac{1}{2}[X(z^{1/2}) + X(-z^{1/2})] \tag{5.237}$$

Here the down-sampled version $x_{(2)}[m]$ of a signal $x[m]$ is composed of all the even terms of the signal with all odd terms dropped, i.e., $x_{(2)}[m] = x[2m]$.
Proof:

$$\begin{aligned}\mathcal{Z}[x_{(2)}[m]] &= \sum_{m=-\infty}^{\infty} x[2m]z^{-m} = \sum_{n=\cdots,-2,0,,2,\cdots} x[n](z^{1/2})^{-n} \\ &= \frac{1}{2}[\sum_{n=-\infty}^{\infty} x[n](z^{1/2})^{-n} + \sum_{n=-\infty}^{\infty} x[n](-z^{1/2})^{-n}] \\ &= \frac{1}{2}[X(z^{1/2}) + X(-z^{1/2})]\end{aligned} \tag{5.238}$$

where we have assumed $n = 2m$. The second equal sign is due to the fact that the sum of the two terms is zero when $n = \cdots, -3, -1, 1, 3, \cdots$ is odd.

- **Up-sampling**

$$\mathcal{Z}[x^{(k)}[m]] = X(z^k) \tag{5.239}$$

Here $x^{(k)}[m]$ is defined as:

$$x^{(k)}[m] = \begin{cases} x[m/k] & \text{if } m \text{ is a multiple of } k \\ 0 & \text{else} \end{cases} \tag{5.240}$$

i.e. $x^{(k)}[m]$ is obtained by inserting $k-1$ zeros between every two consecutive samples of $x[m]$.

Proof:

$$\mathcal{Z}[x^{(k)}[m]] = \sum_{m=-\infty}^{\infty} x[m/k]z^{-m} = \sum_{n=-\infty}^{\infty} x[n]z^{-kn} = X(z^k) \tag{5.241}$$

Note that the change of the summation index from m to $n = m/k$ has no effect as the terms skipped are all zeros.

- **Convolution**

$$\mathcal{Z}\left[x[m] * y[m]\right] = X(z)Y(z), \quad ROC \supseteq (R_x \cap R_y) \tag{5.242}$$

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**

$$\mathcal{Z}\left[\sum_k x[k]x[k-n]\right] = X(z)X(z^{-1}) \tag{5.243}$$

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**

$$\mathcal{Z}\left[x[m] - x[m-1]\right] = (1 - z^{-1})X(z), \quad ROC = R_x \tag{5.244}$$

Proof:

$$\mathcal{Z}[x[m] - x[m-1]] = X(z) - z^{-1}X(z) = (1 - z^{-1})X(z) \tag{5.245}$$

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**

$$\mathcal{Z}\left[\sum_{k=-\infty}^{n} x[k]\right] = \frac{1}{1 - z^{-1}}X(z) \tag{5.246}$$

Proof: First we realize that the accumulation of $x[m]$ can be written as its convolution with $u[m]$:

$$u[m] * x[m] = \sum_{k=-\infty}^{\infty} u[m-k]x[k] = \sum_{k=-\infty}^{m} x[k] \tag{5.247}$$

Applying the convolution property, we get

$$\mathcal{Z}\left[\sum_{k=-\infty}^{m} x[k]\right] = \mathcal{Z}[u[m] * x[m]] = \frac{1}{1-z^{-1}}X(z) \tag{5.248}$$

as $\mathcal{Z}[u[m]] = 1/(1-z^{-1})$.

- **Scaling in Z-domain**

$$\mathcal{Z}\left[z_0^m x[m]\right] = X\left(\frac{z}{z_0}\right), \quad ROC = |z_0|R_x \tag{5.249}$$

Proof:

$$\mathcal{Z}\left[z_0^m x[m]\right] = \sum_{m=-\infty}^{\infty} x[m]\left(\frac{z}{z_0}\right)^{-1} = X\left(\frac{z}{z_0}\right) \tag{5.250}$$

In particular, if $z_0 = e^{j\omega_0}$, the above becomes

$$\mathcal{Z}\left[e^{jm\omega_0}x[m]\right] = X(e^{-j\omega_0}z), \quad ROC = R_x \tag{5.251}$$

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**

$$\mathcal{Z}\left[x^*[m]\right] = X^*(z^*), \quad ROC = R_x \tag{5.252}$$

Proof: Complex conjugate of the Z-transform of $x[m]$ is

$$X^*(z) = \left[\sum_{m=-\infty}^{\infty} x[m]z^{-m}\right]^* = \sum_{m=-\infty}^{\infty} x^*[m](z^*)^{-m} \tag{5.253}$$

Replacing z by z^*, we get the desired result.

- **Differentiation in z-Domain**

$$\mathcal{Z}\left[mx[m]\right] = -\frac{d}{dz}X(z), \quad ROC = R_x \tag{5.254}$$

Proof:

$$\frac{d}{dz}X(z) = \sum_{m=-\infty}^{\infty} x[m]\frac{d}{dz}(z^{-m}) = \sum_{n=-\infty}^{\infty}(-m)x[m]z^{-m-1} \tag{5.255}$$

i.e.,

$$\mathcal{Z}\left[mx[m]\right] = -z\frac{d}{dz}X(z) \tag{5.256}$$

Example 5.14: Consider the following examples:

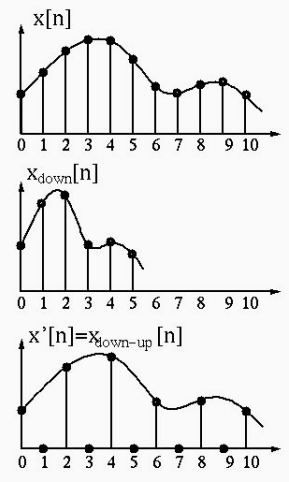

Figure 5.10 Down and up sampling

- The Z-transform of a modulated, time-reversed and shifted signal $(-1)^n x[k-n]$ is

$$\mathcal{Z}[(-1)^n x[k-n]] = \sum_{n=-\infty}^{\infty} (-1)^n x[k-n] z^{-n} = \sum_{n=-\infty}^{\infty} x[k-n](-z)^{-n}$$
$$= \sum_{m=-\infty}^{\infty} x[m](-z)^{m-k} = (-z)^{-k} \sum_{m=-\infty}^{\infty} x[m](-z^{-1})^{-m} = (-z)^{-k} X(-z^{-1}) \tag{5.257}$$

  where $m = k - n$.
- The Z-transform of a signal $x[n]$ first down-sampled then up-sampled is

$$X'(z) = \frac{1}{2}[X(z) + X(-z)] \tag{5.258}$$

  which can be obtained by applying the properties of down-sampling and up-sampling in Eqs.5.237 and 5.239. To verify this result, we apply the property of modulation in Eq.5.235 to the second term and get:

$$x'[n] = \mathcal{Z}^{-1}[X'(z)] = \frac{1}{2}[\mathcal{Z}^{-1}[X(z)] + \mathcal{Z}^{-1}[X(-z)]]$$
$$= \frac{1}{2}[x[n] + (-1)^n x[n]] = \begin{cases} x[n] & \text{even } n \\ 0 & \text{odd } n \end{cases} \tag{5.259}$$

- Taking derivative of the right side of

$$\mathcal{Z}[a^m u[m]] = \frac{1}{1 - az^{-1}}, \quad |z| > a \tag{5.260}$$

we get

$$\frac{d}{dz}[\frac{1}{1-az^{-1}}] = \frac{-az^{-2}}{(1-az^{-1})^2} \tag{5.261}$$

Due to the property of differentiation in z-domain, we have

$$\mathcal{Z}[ma^m u[m]] = \frac{az^{-1}}{1-az^{-1}}, \quad |z| > a \tag{5.262}$$

Note that for a different ROC $|z| < a$, we have

$$\mathcal{Z}[-ma^m u[-m-1]] = \frac{az^{-1}}{1-az^{-1}}, \quad |z| < a \tag{5.263}$$

5.2.4 Z-Transform of Typical Signals

- $\delta[m]$, $\delta[m-n]$

$$\mathcal{Z}[\delta[m]] = \sum_{m=-\infty}^{\infty} \delta[m] z^{-m} = 1, \quad \text{for all } z \tag{5.264}$$

Due to the time shifting property, we also have

$$\mathcal{Z}[\delta[m-n]] = z^{-n}, \quad \text{for all } z \tag{5.265}$$

- $u[m]$, $a^m u[m]$, $ma^m u[m]$

$$\mathcal{Z}[u[m]] = \sum_{m=0}^{\infty} z^{-m} = \frac{1}{1-z^{-1}}, \quad |z| > 1 \tag{5.266}$$

Due to the scaling in z-domain property, we have

$$\mathcal{Z}[a^m u[m]] = \frac{1}{1-(z/a)^{-1}} = \frac{1}{1-az^{-1}}, \quad |z| > a \tag{5.267}$$

Applying the property of differentiation in z-Domain to the above, we have

$$\mathcal{Z}[ma^m u[m]] = -z\frac{d}{dz}[\frac{1}{1-az^{-1}}] = -z\frac{-az^{-2}}{(1-az^{-1})^2} = \frac{az^{-1}}{(1-az^{-1})^2}, \quad |z| > a \tag{5.268}$$

- $e^{\pm jm\omega_0}u[m]$, $cos[m\omega_0]u[m]$, $sin[m\omega_0]u[m]$
  Applying the scaling in z-domain property to $\mathcal{Z}[u[m]] = 1/(1-z^{-1})$, we have

$$\mathcal{Z}[e^{jm\omega_0}u[m]] = \frac{1}{1-(e^{j\omega_0}z)^{-1}} = \frac{1}{1-e^{-j\omega_0}z^{-1}}, \quad |z| > 1 \tag{5.269}$$

and similarly, we have

$$\mathcal{Z}[e^{-jm\omega_0}u[m]] = \frac{1}{1-e^{j\omega_0}z^{-1}}, \quad |z| > 1 \tag{5.270}$$

Moreover, we have

$$\begin{aligned}\mathcal{Z}[cos(m\omega_0)u[m]] &= \mathcal{Z}[\frac{e^{jm\omega_0}+e^{-jm\omega_0}}{2}u[m]] = \frac{1}{2}[\frac{1}{1-e^{j\omega_0}z^{-1}}+\frac{1}{1-e^{-j\omega_0}z^{-1}}] \\ &= \frac{2-(e^{j\omega_0}+e^{-j\omega_0})z^{-1}}{2[1-(e^{j\omega_0}+e^{-j\omega_0})z^{-1}+z^{-2}]} \\ &= \frac{1-cos\omega_0 z^{-1}}{1-2cos\omega_0 z^{-1}+z^{-2}} \quad |z|>1 \end{aligned} \tag{5.271}$$

Similarly we have

$$\mathcal{Z}[sin(m\omega_0)u[m]] = \frac{\sin\omega_0 z^{-1}}{1-2\cos\omega_0 z^{-1}+z^{-2}}, \quad |z|>1 \tag{5.272}$$

- $r^m cos[m\omega_0]u[m]$, $r^m sin[m\omega_0]u[m]$
  Applying the z-domain scaling property to the above, we have

$$\mathcal{Z}[r^m cos(m\omega_0)u[m]] = \frac{1-r\ cos\omega_0 z^{-1}}{1-2r\ cos\omega_0 z^{-1}+r^2 z^{-2}}, \quad |z|>r \tag{5.273}$$

and

$$\mathcal{Z}[r^m \sin(m\omega_0)u[m]] = \frac{r\ sin\omega_0 z^{-1}}{1-2r\ cos\omega_0 z^{-1}+r^2 z^{-2}}, \quad |z|>r \tag{5.274}$$

5.2.5 Analysis of LTI Systems by Z-Transform

The Z-transform is a convenient tool for the analysis and design of discrete LTI systems whose output $y[m]$ is the convolution of the input $x[m]$ and its impulse response function $h[m]$:

$$y[m] = \mathcal{O}[x[m]] = h[m] * x[m] = \sum_{k=-\infty}^{\infty} h[k]x[m-k] \tag{5.275}$$

In particular, if the input is an impulse $x[m] = \delta[m]$, then the out is the impulse response function:

$$y[m] = \mathcal{O}[\delta[m]] = h[m] * \delta[m] = \sum_{n=-\infty}^{\infty} h[n]\delta[m-n] = h[m] \tag{5.276}$$

If the input is a complex exponential $x[m] = e^{sm} = z^m$ where $z = e^s = e^{\sigma+j\omega}$, then the output is:

$$y[m] = \mathcal{O}[z^m] = \sum_{k=-\infty}^{\infty} h[k]z^{m-k} = z^m \sum_{k=-\infty}^{\infty} h[k]z^{-k} = H(z)z^m \tag{5.277}$$

where $H(z)$ is the *transfer function* of the system, first defined in Eq.1.88 in Chapter 1, which is actually the Z-transform of the impulse response $h[m]$ of the

system:

$$H(z) = \sum_{k=-\infty}^{\infty} h[k]z^{-k} \tag{5.278}$$

Note that Eq.5.277 is the eigenequation of *any* discrete LTI system, where the transfer function $H(z)$ is the eigenvalue, and the complex exponential input $x[m] = e^{sm} = z^m$ 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 $h[m]$ of the system:

$$H(z)\big|_{s=j\omega} = H(e^{j\omega}) = \sum_{m=-\infty}^{\infty} h[m]e^{-j\omega m} = \mathcal{F}[h[m]] \tag{5.279}$$

This is the frequency response function of the discrete LTI system first defined in Eq.4.81 of Chapter 3.

Moreover, due to its convolution property of the Z-transform, the convolution in Eq.5.275 can be converted to a multiplication in z-domain:

$$y[m] = h[m] * x[m] \xrightarrow{\mathcal{Z}} Y(z) = H(z)X(z) \tag{5.280}$$

Based on this relationship the transfer function $H(z)$ can also be found in z-domain as the ratio of the output $Y(z)$ and input $X(z)$:

$$H(z) = \frac{Y(z)}{X(z)} \tag{5.281}$$

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 $|x[m]| < B$ its response $y[m]$ is also bounded for all m, and its impulse response function $h[m]$ needs to be absolutely summable (Eq.1.96):

  $$\sum_{m=-\infty}^{\infty} \big|h[m]\big| < \infty \tag{5.282}$$

  i.e., the frequency response function $\mathcal{F}[h[m]] = H(e^{j\omega}) = H(z)\big|_{z=e^{j\omega}}$ 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**
  As discussed in Chapter 1, a discrete LTI system is causal if its impulse response $h[m]$ is a consequence of the impulse input $\delta[m]$, i.e., $h[m]$ comes after $\delta[m]$:

  $$h[m] = h[m]u[m] = \begin{cases} h[m] & m \geq 0 \\ 0 & m < 0 \end{cases} \tag{5.283}$$

and its output is (Eq.1.97):

$$y[m] = \sum_{n=-\infty}^{\infty} h[n]x[m-n] = \sum_{m=0}^{\infty} h[n]x[m-n] \tag{5.284}$$

We see that 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 (the magnitudes of all poles are smaller than 1).

Many LTI system can be described by a linear constant-coefficient difference equation (LCCDE) in time domain

$$\sum_{k=0}^{N} a_k y[m-k] = \sum_{k=0}^{M} b_k x[m-k] \tag{5.285}$$

Taking Z-transform of this equation, we get an algebraic equation in the z domain:

$$Y(z)[\sum_{k=0}^{N} a_k z^{-k}] = X(s)[\sum_{k=0}^{M} b_k z^{-k}] \tag{5.286}$$

The transfer function of such a system is rational:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{\sum_{k=0}^{N} a_k z^{-k}} = \frac{N(z)}{D(z)} = \frac{b_M}{a_N} \frac{\prod_{k=1}^{M}(z - z_{0_k})}{\prod_{k=1}^{N}(z - z_{0_k})} \tag{5.287}$$

where z_k, $(k = 1, 2, \cdots, M)$ are the roots of the numerator polynomial $N(z)$, and p_k, $(k = 1, 2, \cdots, N)$ are the roots of the denominator polynomial $D(z)$, they are also respectively the zeros and poles of $H(z)$.

Note that just as the LCCDE alone does not completely specify the relationship between $x[m]$ and $y[m]$ (additional information such as the initial conditions is needed), the transfer function $H(z)$ does not completely specify the system. For example, the same $H(z)$ with different ROCs will represent different systems (e.g., causal or anti-causal).

Example 5.15: The input and output of an LTI system are related by

$$y[m] - \frac{1}{2}y[m-1] = x[m] + \frac{1}{3}x[m-1] \tag{5.288}$$

Note that without further information such as the initial condition, this equation does not uniquely specify $y[m]$ when $x[m]$ is given. Taking Z-transform of this equation and using the time shifting property, we get

$$Y(z) - \frac{1}{2}z^{-1}Y(z) = X(z) + \frac{1}{3}z^{-1}X(z) \tag{5.289}$$

and the transfer function can be obtained

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 + \frac{1}{3}z^{-1}}{1 - \frac{1}{2}z^{-1}} = \frac{1}{1 - \frac{1}{2}z^{-1}}(1 + \frac{1}{3}z^{-1}) \tag{5.290}$$

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:

$$h[m] = (\frac{1}{2})^m u[m] + \frac{1}{3}(\frac{1}{2})^{m-1} u[m-1] \tag{5.291}$$

- 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:

$$h[m] = -(\frac{1}{2})^m u[-m-1] - \frac{1}{3}(\frac{1}{2})^{m-1} u[-m] \tag{5.292}$$

5.2.6 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 problem can be resolved by the *unilateral* Z-transform defined below:

$$\mathcal{U}Z[x[n]] = X(z) = \sum_{m=-\infty}^{\infty} x[m]u[m]z^{-m} = \sum_{m=0}^{\infty} x[m]z^{-m} \tag{5.293}$$

When the unilateral Z-transform is applied to a signal $x[m]$, it is always assumed that the signal starts at time $m = 0$, i.e., $x[m] = 0$ for $m < 0$; when it is applied to the impulse response function of a LTI system to find the transfer function $H(z) = \mathcal{U}Z[h[m]]$, it is always assumed that the system is causal, i.e., $h[m] = 0$ for $m < 0$. In both cases, the ROC is always the exterior of a circle.

By definition, the unilateral Z-transform of any signal $x[m] = x[m]u[m]$ is identical to its bilateral Z-transform. However, when $x[m] \neq x[m]u[m]$, 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**

$$\begin{aligned} \mathcal{U}Z[x[m+1]] &= \sum_{m=0}^{\infty} x[m+1]z^{-m} = z\sum_{n=1}^{\infty} x[n]z^{-n} \\ &= z[\sum_{n=0}^{\infty} x[n]z^{-n} - x[0]] = zX(z) - zx[0] \end{aligned} \tag{5.294}$$

  where $n = m + 1$.

- **Time delay**

$$\mathcal{U}Z[x[m-1]] = \sum_{m=0}^{\infty} x[m-1]z^{-m} = z^{-1}\sum_{n=-1}^{\infty} x[n]z^{-n}$$

$$= z^{-1}[\sum_{n=0}^{\infty} x[n]z^{-n} + zx[-1]] = z^{-1}X(z) + x[-1] \tag{5.295}$$

where $n = m-1$. Similarly, we have

$$\mathcal{U}Z[x[m-2]] = \sum_{m=0}^{\infty} x[m-2]z^{-m} = z^{-2}\sum_{n=-2}^{\infty} x[n]z^{-n}$$

$$= z^{-2}[\sum_{n=0}^{\infty} x[n]z^{-n} + zx[-1] + z^2x[-2]] = z^{-2}X(z) + x[-1]z^{-1} + x[-2] \tag{5.296}$$

where $n = m-2$. In general, we have

$$\mathcal{U}Z[x[m-m_0]] = z^{-m_0}X(z) + \sum_{k=0}^{m_0-1} z^{-k}x[k-m_0] \tag{5.297}$$

- **Initial value theorem**
  If $x[m] = x[m]u[mn]$, i.e., $x[m] = 0$ for $m < 0$, then

$$x[0] = \lim_{z\to\infty} X(z) \tag{5.298}$$

  Proof:

$$\lim_{z\to\infty} X(z) = \lim_{z\to\infty}[\sum_{m=0}^{\infty} x[m]z^{-m}] = x[0] \tag{5.299}$$

  All terms with $n > 0$ become zero as $z^{-m} = 1/z^m \to 0$ as $z \to \infty$, except the first one which is always $x[0]$.
- **Final value theorem**
  If $x[m] = x[m]u[m]$, i.e., $x[m] = 0$ for $m < 0$, then

$$\lim_{m\to\infty} x[m] = \lim_{z\to 1}(1-z^{-1})X(z) \tag{5.300}$$

  Proof:

$$\mathcal{Z}[x[m] - x[m-1]] = X(z) - X(z)z^{-1} = \sum_{m=0}^{\infty}[x[m] - x[m-1]]z^{-m} \tag{5.301}$$

  i.e.

$$(1-z^{-1})X(z) = \lim_{N\to\infty}\sum_{m=0}^{N}[x[m] - x[m-1]]z^{-m} \tag{5.302}$$

Letting $z \to 1$ in the above, we get

$$\lim_{z\to 1}(1-z^{-1})X(z) = \lim_{N\to\infty}\sum_{m=0}^{N}[x[m]-x[m-1]]$$
$$= \lim_{N\to\infty}\{\sum_{m=0}^{N-1}[x[m]-x[m]]+x[N]-x[-1]\} = \lim_{N\to\infty}x[N]$$

as $x[-1]=0$.

Due to these properties, the unilateral Z-transform is a powerful tool for solving LCCDEs with non-zero initial conditions.

Example 5.16: A system is described by this LCCDE

$$y[m]+3y[m-1]=x[m]=\alpha u[m] \tag{5.303}$$

Taking unilateral Z-transform of the DE, we get

$$Y(z)+3Y(z)z^{-1}+3y[-1]=X(z)=\frac{\alpha}{1-z^{-1}} \tag{5.304}$$

- **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

$$Y(z)=H(z)X(z)=\frac{1}{1+3z^{-1}}\ \frac{\alpha}{1-z^{-1}}=\frac{3\alpha/4}{1+3z^{-1}}+\frac{\alpha/4}{1-z^{-1}} \tag{5.305}$$

  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):

$$y_p[m]=\alpha[\frac{1}{4}+\frac{3}{4}(-3)^m]u[m] \tag{5.306}$$

- **The homogeneous (zero-input) solution**
  When the initial condition is nonzero

$$y[-1]=\beta \tag{5.307}$$

  but the input is zero $x[m]=0$, the Z-transform of the difference equation becomes

$$Y(z)+3Y(z)z^{-1}+3\beta=0 \tag{5.308}$$

  Solving this for $Y(z)$ we get

$$Y(z)=\frac{-3\beta}{1+3z^{-1}} \tag{5.309}$$

In time domain, this is the homogeneous (or zero-input) solution (caused by the initial condition with zero input):

$$y_h[m] = -3\beta(-3)^m u[m] \tag{5.310}$$

When neither $y[-1]$ nor $x[m]$ is zero, we have

$$Y(z) + 3Y(z)z^{-1} + 3\beta = X(z) = \frac{\alpha}{1 - z^{-1}} \tag{5.311}$$

Solving this algebraic equation in z-domain for $Y(z)$ we get

$$Y(z) = \frac{\alpha}{(1 + 3z^{-1})(1 - z^{-1})} - \frac{3\beta}{1 + 3z^{-1}} \tag{5.312}$$

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

$$Y(z) = \frac{1}{1 + 3z^{-1}}(\frac{3}{4}\alpha - 3\beta) + \frac{\alpha}{4}\frac{1}{1 - z^{-1}} \tag{5.313}$$

and in time domain, we have the general solution

$$y_g[m] = [(\frac{3}{4}\alpha - 3\beta)(-3)^m + \frac{\alpha}{4}]u[m] = y_h[m] + y_p[m] \tag{5.314}$$

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.

Example 5.17: The input to an LTI is

$$x(t) = e^{-3t}u(t) \tag{5.315}$$

and the output is

$$y(t) = h(t) * x(t) = (e^{-t} - e^{-2t})u(t) \tag{5.316}$$

We want to identify the system by finding $h(t)$ and $H(s)$. In s-domain, input and output signals are

$$X(s) = \frac{1}{s + 3} \quad Re[s] > -3 \tag{5.317}$$

and

$$Y(s) = H(s)X(s) = \frac{1}{s + 1} - \frac{1}{s + 2} = \frac{1}{(s + 1)(s + 2)} \quad Re[s] > -1 \tag{5.318}$$

The transfer function can therefore be obtained

$$H(s) = \frac{Y(s)}{X(s)} = \frac{s + 3}{(s + 1)(s + 2)} = \frac{s + 3}{s^2 + 3s + 2} \tag{5.319}$$

This system $H(s)$ has two poles $p_1 = -1$ and $p_2 = -2$ and therefore three possible ROCs: $Re[s] < -2$, $-2 < Re[s] < -1$ and $Re[s] > -1$ 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 $Re[s] > -1$, 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:

$$\frac{d^2}{dt^2}y(t) + 3\frac{d}{dt}y(t) + 2y(t) = \frac{d}{dt}x(t) + 3x(t) \tag{5.320}$$

Example 5.18: Find the inverse of the given Z-transform

$$X(z) = 4z^2 + 2 + 3z^{-1} \tag{5.321}$$

Comparing this with the definition of Z-transform:

$$X(s) = \sum_{m=-\infty}^{\infty} x[m]z^{-m} = \cdots x[-2]z^2 + x[-1]z^1 + x[0] + x[1]z^{-1} + x[2]z^{-2} + \cdots \tag{5.322}$$

we get

$$x[m] = 4\delta[m+2] + 2\delta[m] + 3\delta[m-1] \tag{5.323}$$

In general, we can use the time shifting property

$$\mathcal{Z}[\delta[m + m_0]] = z^{m_0} \tag{5.324}$$

to inverse transform the $X(z)$ given above to $x[m]$ directly.

///—-

The Z-transform is also widely used to characterize discrete, linear, and time-invariant (LTI) systems. In fact, the transfer function $H(z)$ of a discrete LIT system defined in Eq.1.88 in Chapter 1 is just the Z-transform of the impulse response function $h[m]$ of the system:

$$H(z) = \sum_{m=-\infty}^{\infty} h[m]z^{-m} = \mathcal{Z}[h[m]] \tag{5.325}$$

In particular, if we let $Re[s] = \sigma = 0$, i.e., $z = e^{j\omega}$, the above becomes the discrete-time Fourier transform of $h[m]$:

$$H(z)\big|_{z=e^{j\omega}} = H(e^{j\omega}) = \sum_{m=-\infty}^{\infty} h[m]e^{-j\omega m} = \mathcal{F}[h[m]] \tag{5.326}$$

This is the frequency response function of the LTI system defined in Eq.4.81 of Chapter 4.

Example 5.19:

$$x(t) = e^{-a(t+1)}u(t+1) \tag{5.327}$$

This signal is right-sided starting at $t = -1$ (i.e., $x(t) \neq x(t)u(t)$). By definition, the bilateral Laplace transform of $x(t)$ is

$$\begin{aligned}\mathcal{L}[x(t)] &= \int_{-1}^{\infty} e^{-a(t+1)}e^{-st}dt = e^{-a}\int_{-1}^{\infty} e^{-(a+s)t}dt \\ &= \frac{e^{-a}}{-(a+s)}e^{-(a+s)t}\Big|_{-1}^{\infty} = \frac{e^{s}}{a+s}, \qquad Re[s] > -a\end{aligned}$$

The unilateral Laplace transform of this signal is

$$\begin{aligned}\mathcal{U}L[x(t)] &= \int_{0}^{\infty} e^{-a(t+1)}e^{-st}dt = e^{-a}\int_{0}^{\infty} e^{-(a+s)t}dt \\ &= \frac{e^{-a}}{-(a+s)}e^{-(a+s)t}\Big|_{0}^{\infty} = \frac{e^{-a}}{a+s}, \qquad Re[s] > -a\end{aligned}$$

Example 5.20:

$$x[m] = a^{-(m+1)}u[m+1] \tag{5.328}$$

This signal is right-sided starting at $m = -1$ (i.e., $x[m] \neq x[m]u[m]$). By definition, the bilateral Z-transform of $x[m]$ is

$$\mathcal{Z}[x[m]] = \sum_{m=-1}^{\infty} a^{-(m+1)}z^{-m} = z + a^{-1}\sum_{m=0}^{\infty} a^{-m}z^{-m} = z + \frac{a^{-1}}{1-(az)^{-1}} = \frac{z}{1-(az)^{-1}} \tag{5.329}$$

It was assumed that $|z| > a$. The unilateral Z-transform of this signal is

$$\mathcal{U}Z[x[m]] = \sum_{m=0}^{\infty} a^{-(m+1)}z^{-m} = a^{-1}\sum_{m=0}^{\infty}(az)^{-m} = \frac{a^{-1}}{1-(az)^{-1}} \tag{5.330}$$

If we assume zero initial condition $y[-1] = 0$,

6 Fourier Related Orthogonal Transforms

6.1 The Hartley Transform

6.1.1 Continuous Hartley Transform

Similar to the cosine transform considered previously, the Hartley transform is also a real integral transform that is closely related to the Fourier transform. Specifically, the kernel function of the Hartley transform is:

$$\begin{aligned}\phi_f(t) &= cas(2\pi ft) = \cos(2\pi ft) + \sin(2\pi ft) \\ &= \sqrt{2}\ \sin(2\pi ft + \frac{\pi}{4}) = \sqrt{2}\ \cos(2\pi ft - \frac{\pi}{4}), \quad (-\infty < t, f < \infty)\end{aligned} \tag{6.1}$$

This kernel function $\phi_f(t) = cas(2\pi ft)$ is the cosine-and-sine function defined as:

$$\phi_f(t) = cas(2\pi ft) = \cos(2\pi ft) + \sin(2\pi ft) = \phi_t(f) \tag{6.2}$$

which is symmetric with respect to f and t. We can show that this is a set of uncountable orthonormal functions satisfying:

$$< \phi_f(t), \phi_{f'}(t) >= \delta(f - f'), \quad \text{and} \quad < \phi_t(f), \phi_{t'}(f) >= \delta(t - t') \tag{6.3}$$

Proof:

$$\begin{aligned}&< \phi_f(t), \phi_{f'}(t) >= \int_{-\infty}^{\infty} \phi_f(t)\phi_{f'}(t)\ dt \\ &= \int_{-\infty}^{\infty} [\cos(2\pi ft) + \sin(2\pi ft)]\,[\cos(2\pi f't) + \sin(2\pi f't)]\ dt \\ &= \int_{-\infty}^{\infty} [\cos(2\pi ft)\cos(2\pi f't) + \sin(2\pi ft)\sin(2\pi f't)]\ dt \\ &+ \int_{-\infty}^{\infty} [\cos(2\pi ft)\sin(2\pi f't) + \sin(2\pi ft)\cos(2\pi f't)]\ dt \\ &= \int_{-\infty}^{\infty} \cos(2\pi(f - f')t)\ dt + \int_{-\infty}^{\infty} \sin(2\pi(f + f')t)\ dt\end{aligned} \tag{6.4}$$

Here the first term is a Dirac delta $\delta(f - f')$ according to Eq.1.30, while the second integral can be dropped as its integrand $\sin(2\pi(f + f')t)$ is odd with respect to t, so that the integral over all time is zero. Now the inner product

above becomes

$$< \phi_f(t), \phi_{f'}(t) >= \delta(f - f') \tag{6.5}$$

As $\phi_f(t) = \phi_t(f)$ is symmetric with respect to t and f, we also have:

$$< \phi_t(f), \phi_{t'}(f) >= \delta(t - t') \tag{6.6}$$

The Hartley transform can now be similarly defined as the Fourier transform. Based on the kernel function $\phi_f(t) = e^{j2\pi ft} = \cos(2\pi ft) + j\sin(2\pi ft)$, the Fourier transform is defined as:

$$\begin{aligned} X_F(f) = \mathcal{F}[x(t)] =< x(t), \phi_f(t) >= \int_{-\infty}^{\infty} x(t)\overline{\phi}_f(t)\, dt \\ = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}\, dt = \int_{-\infty}^{\infty} x(t)[\cos(2\pi ft) - j\sin(2\pi ft)]\, dt \end{aligned}$$

Based on a different kernel function $\phi_f(t) = \cos(2\pi ft) + \sin(2\pi ft)$, the Hartley transform is defined as:

$$\begin{aligned} X_H(f) = \mathcal{H}[x(t)] =< x(t), \phi_f(t) >= \int_{-\infty}^{\infty} x(t)\phi_f(t)\, dt \\ = \int_{-\infty}^{\infty} x(t)[\cos(2\pi ft) + \sin(2\pi ft)]\, dt \end{aligned} \tag{6.7}$$

Here the transform $X_H(f)$ is a function of frequency f and is therefore 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:

$$\begin{aligned} < X_H(f), \phi_{t'}(f) > = \int_{-\infty}^{\infty} X_H(f)\phi_(t')(f)\, df = \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} x(t)\phi_f(t)\, dt\right] \phi_(t')(f)\, df \\ = \int_{-\infty}^{\infty} x(t) \left[\int_{-\infty}^{\infty} \phi_f(t)\phi_(t')(f)\, df\right] dt \\ = \int_{-\infty}^{\infty} x(t)\delta(t - t')\, dt = x(t') \end{aligned} \tag{6.8}$$

We can write both the forward and inverse Hartley transform as the following pair of equations:

$$\begin{aligned} X_H(f) = \mathcal{H}[x(t)] = < x(t), \phi_f(t) >= \int_{-\infty}^{\infty} x(t)\, [\cos(2\pi ft) + \sin(2\pi ft)]\, dt \\ x(t) = \mathcal{H}^{-1}[X_H(f)] = < X_H(f), \phi_t(f) >= \int_{-\infty}^{\infty} X_H(f)\, [\cos(2\pi ft) + \sin(2\pi ft)]\, df \end{aligned} \tag{6.9}$$

As $\phi_f(t) = \phi_t(t)$ is symmetric, the inverse transform $\mathcal{H}$ is identical to the forward transform $\mathcal{H}^{-1}$:

$$x(t) = \mathcal{H}^{-1}[X_H(f)] = \mathcal{H}[X_H(f)] = \mathcal{H}[\, \mathcal{H}[x(t)]\,] \tag{6.10}$$

6.1.2 Properties of the Hartley Transform

- **Relation to Fourier transform:**
  Here we assume the signal $x(t) = \overline{x}(t)$ is real. First, if a signal $x(t) = x(-t)$ is even, its Fourier spectrum is real and therefore its the same as its Hartley spectrum; on the other hand, if a signal $x(t) = -x(-t)$ is odd, its Fourier spectrum is imaginary and its Hartley spectrum is the negative version of its Fourier spectrum. Second, we consider the general case where the signal is neither even nor odd. Then its Hartley spectrum is:

$$\begin{aligned} X_H(f) = \mathcal{H}[x(t)] &= \int_{-\infty}^{\infty} x(t)[\cos(2\pi ft) + \sin(2\pi ft)]\, dt \\ &= \int_{-\infty}^{\infty} x(t)\cos(2\pi ft)\, dt + \int_{-\infty}^{\infty} x(t)\sin(2\pi ft)\, dt \\ &= X_e(f)] + X_o(f) \end{aligned} \tag{6.11}$$

  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:

$$\begin{aligned} X_F(f) = \mathcal{F}[x(t)] &= \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}\, dt = \int_{-\infty}^{\infty} x(t)[\cos(2\pi ft) - j\sin(2\pi ft)]\, dt \\ &= \int_{-\infty}^{\infty} x(t)\cos(2\pi ft)\, dt - j\int_{-\infty}^{\infty} x(t)\sin(2\pi ft)\, dt \\ &= X_e(f) - j\, X_o(f) \end{aligned} \tag{6.12}$$

  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 $X_F(f)$:

$$X_e(f) = Re[X_F(f)], \qquad X_o(f) = -Im[X_F(f)]$$

  Now the Hartley spectrum can be obtained as a linear combination of the real and imaginary parts of the Fourier spectrum:

$$X_H(f) = X_e(f) + X_o(f) = Re[X_F(f)] - Im[X_F(f)] \tag{6.13}$$

- **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:

$$\begin{aligned} & Z_H(f) = \mathcal{H}[x(t) * y(t)] \\ = & \frac{1}{2}\left[X_H(f)Y_H(f) - X_H(-f)Y_H(-f) + X_H(f)Y_H(-f) + X_H(-f)Y_H(f)\right] \end{aligned} \tag{6.14}$$

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.3.113), the Fourier spectrum $Z_F(f) = \mathcal{F}[z(t)]$ is the product of the spectra $X_F(f) = \mathcal{F}[x(t)]$ and $Y_F(f) = \mathcal{F}[y(t)]$:

$$\begin{aligned} Z_F(t) &= X_F(f)\, Y_F(f) = [X_e(f) - j\, X_o(f)]\, [Y_e(f) - j\, Y_o(f)] \\ &= [X_e(f)Y_e(f) - X_o(f)Y_o(f)] - j\, [X_o(f)Y_e(f) + X_e(f)Y_o(f)] \\ &= Z_e(f) - j\, Z_o(f) \end{aligned} \tag{6.15}$$

where $X_e(f)$, $X_e(f)$ and $Y_e(f)$, $Y_e(f)$ are the even and odd components of $X_H(f)$ and $Y_H(f)$, respectively:

$$\begin{aligned} X_e(f) &= \frac{1}{2}[X_H(f) + X_H(-f)], \qquad X_o(f) = \frac{1}{2}[X_H(f) - X_H(-f)] \\ Y_e(f) &= \frac{1}{2}[Y_H(f) + Y_H(-f)], \qquad Y_o(f) = \frac{1}{2}[Y_H(f) - Y_H(-f)] \end{aligned}$$

and $Z_e(f)$ and $Z_o(f)$ are the even and odd components of $Z_H(f)$ (note that the product of two even or odd functions is even, and the product of an even function and an odd function is odd):

$$\begin{aligned} Z_e(f) &= X_e(f)Y_e(f) - X_o(f)Y_o(f) = \frac{1}{2}[X_H(f)Y_H(-f) + X_H(-f)Y_H(f)] \\ Z_o(f) &= X_e(f)Y_o(f) + X_o(f)Y_e(f) = \frac{1}{2}[X_H(f)Y_H(f) - X_H(-f)Y_H(-f)] \end{aligned} \tag{6.16}$$

Substituting these into $Z_H(f) = Z_e(f) + Z_o(f)$, we get Eq.6.14.

Also, based on Eq.3.114, we can similarly prove the Hartley spectrum of the product of two functions $z(t) = x(t)\, y(t)$ is:

$$\begin{aligned} & Z_H(t) = \mathcal{H}[x(t)\, y(t)] \\ = & \frac{1}{2}\left[X_H(f) * Y_H(f) - X_H(-f) * Y_H(-f) + X_H(f) * Y_H(-f) + X_H(-f) * Y_H(f)\right] \end{aligned} \tag{6.17}$$

- **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:

$$\begin{aligned} & Z_H(f) = \mathcal{H}[x(t) \star y(t)] \\ = & \frac{1}{2}\left[X_H(f)Y_H(f) + X_H(-f)Y_H(-f) + X_H(f)Y_H(-f) - X_H(-f)Y_H(f)\right] \end{aligned} \tag{6.18}$$

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.6.18 becomes:

$$\mathcal{H}[x(t) \star x(t)] = \frac{1}{2}[X_H^2(f) + X_H^2(-f)] \tag{6.19}$$

Proof:
According to the correlation property of the Fourier transform (Eq.3.108), the Fourier spectrum $Z_F(f) = \mathcal{F}[z(t)]$ is the product of the spectra $X_F(f) = \mathcal{F}[x(t)]$ and $Y_F(f) = \mathcal{F}[y(t)]$:

$$\begin{aligned} Z_F(t) &= X_F(f)\,\overline{Y}_F(f) = [X_e(f) - j\,X_o(f)]\,[Y_e(f) + j\,Y_o(f)] \\ &= [X_e(f)Y_e(f) + X_o(f)Y_o(f)] - j\,[X_o(f)Y_e(f) - X_e(f)Y_o(f)] \\ &= Z_e(f) - j\,Z_o(f) \end{aligned} \tag{6.20}$$

where $X_e(f)$, $X_e(f)$ and $Y_e(f)$, $Y_e(f)$ are the even and odd components of $X_H(f)$ and $Y_H(f)$, respectively:

$$X_e(f) = \frac{1}{2}[X_H(f) + X_H(-f)], \qquad X_o(f) = \frac{1}{2}[X_H(f) - X_H(-f)]$$
$$Y_e(f) = \frac{1}{2}[Y_H(f) + Y_H(-f)], \qquad Y_o(f) = \frac{1}{2}[Y_H(f) - Y_H(-f)]$$

and $Z_e(f)$ and $Z_o(f)$ are the even and odd components of $Z_H(f)$:

$$Z_e(f) = X_e(f)Y_e(f) + X_o(f)Y_o(f) = \frac{1}{2}[X_H(f)Y_H(f) + X_H(-f)Y_H(-f)]$$
$$Z_o(f) = X_o(f)Y_e(f) - X_e(f)Y_o(f) = \frac{1}{2}[X_H(f)Y_H(-f) - X_H(-f)Y_H(f)]$$

Substituting these into $Z_H(f) = Z_e(f) + Z_o(f)$, we get Eq.6.18.

6.1.3 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. We consider a few examples:

Example 6.1:

- $$x(t) = cos(2\pi f_0 t + \theta) = \frac{1}{2}[e^{j2\pi f_0 t}e^{j\theta} + e^{-j2\pi f_0 t}e^{-j\theta}]$$

  and its Fourier transform is

  $$\begin{aligned} X_F(f) &= \frac{1}{2}[\delta(f-f_0)e^{j\theta} + \delta(f+f_0)e^{-j\theta}] \\ &= \frac{1}{2}[\delta(f-f_0)(\cos\theta + j\sin\theta) + \delta(f+f_0)(\cos\theta - j\sin\theta)] \\ &= \frac{1}{2}[\delta(f-f_0)\cos\theta + \delta(f+f_0)\cos\theta] + \frac{j}{2}[\delta(f-f_0)\sin\theta - \delta(f+f_0)\sin\theta] \end{aligned}$$

  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 $\theta = 0$, we have $x(t) = \cos(2\pi f_0 t)$, and its Hartley spectrum becomes:

  $$X_H(f) = \mathcal{H}[\cos(2\pi f_0 t)] = \frac{1}{2}[\delta(f-f_0) + \delta(f+f_0)]$$

  which is the same as the Fourier spectrum $X_F(f)$. 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:

  $$X_H(f) = \mathcal{H}[\sin(2\pi f_0 t)] = \frac{1}{2}[\delta(f-f_0) - \delta(f+f_0)]$$

  which is the negative version of imaginary part of the Fourier spectrum

  $$X_F(f) = \frac{1}{2j}[\delta(f-f_0) - \delta(f+f_0)] = \frac{j}{2}[-\delta(f-f_0) + \delta(f+f_0)]$$

  For a specific example, consider a signal:

  $$x(t) = 1 + 3\cos(2\pi 32t) + 2\sin(2\pi 128t) + 2\cos(2\pi 256t + pi/3)$$

  In Fig.6.1 this signal is plotted (top) together with both of its Hartley (middle) and Fourier (bottom) spectra. Also, in the top panel, the reconstruction of the signal (dashed line) from its spectrum is plotted. We see that the reconstruction is perfect as its plot is right on top of that of the original signal. We also see that the DC and cosine component (without phase shift) appear the same in the two spectra. The sine component appears in the two spectra as the negative version of each other. When there is a phase shift of $\pi/3$, the Hartley spectrum is the difference between the real and imaginary parts of the Fourier spectrum.

- $$x(t) = e^{-at}u(t)$$

  This exponential decay function together with its Hartley and Fourier spectra are shown respectively in top, middle and bottom panels of Fig.6.2.

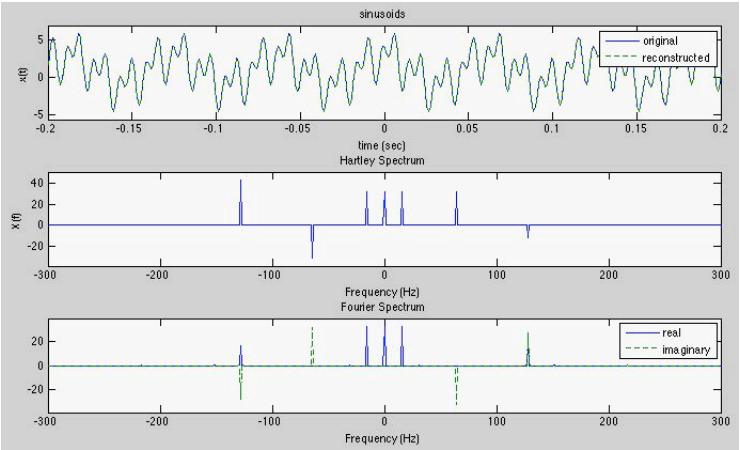

Figure 6.1 The Hartley and Fourier spectra of sinusoidal components of a signal

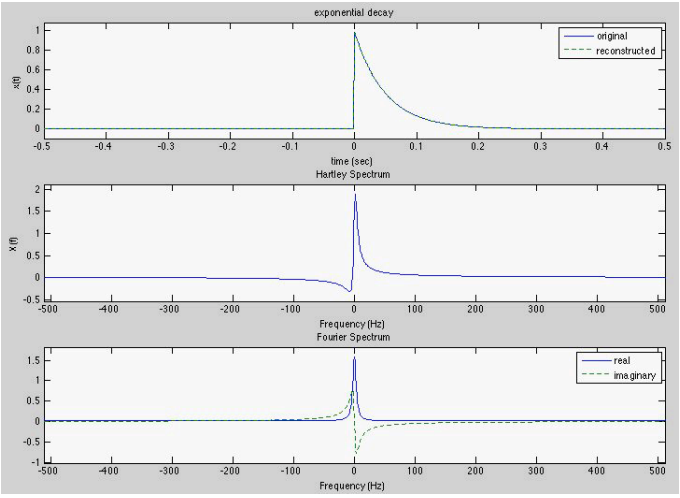

Figure 6.2 The Hartley and Fourier spectra of exponential decay

6.1.4 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 $M = T/t_0$ samples that form an M-D vector $\boldsymbol{x} = [x[0], \cdots, x[M-1]]^T$. Correspondingly the Hartley

transform also becomes discrete based on a discrete kernel:

$$\begin{aligned}\phi_k[m] &= \frac{1}{\sqrt{M}} cas(2\pi\left(\frac{mk}{M}\right)) \\ &= \frac{1}{\sqrt{M}}\left[\cos(2\pi\left(\frac{mk}{M}\right)) + \sin(2\pi\left(\frac{mk}{M}\right))\right] \end{aligned} \tag{6.21}$$

which form a set of basis vectors $\boldsymbol{\phi}_k = [cas(2\pi\, 0k/M), \cdots, cas(2\pi\,(M-1)k/M)]^T$ $(k = 0, \cdots, M-1)$ that span an M-D vector space. Following the proof of Eq.1.29, we can show that these vectors are orthogonal:

$$\begin{aligned} < \boldsymbol{\phi}_k, \boldsymbol{\phi}_{k'} > &= \frac{1}{M}\sum_{m=0}^{M-1} cas(2\pi\, mk/M) cas(2\pi\, mk'/M) \\ &= \frac{1}{M}\left[\sum_{m=0}^{M-1}\cos(2\pi\, m(k-k')/M) + \sum_{m=0}^{M-1}\sin(2\pi\, m(k-k')/M)\right] \\ &= \frac{1}{M}\sum_{m=0}^{M-1}\cos(2\pi\, m(k-k')/M) = \delta[k-k'] \end{aligned} \tag{6.22}$$

The discrete Hartley transform of a signal vector $\boldsymbol{x}$ is then defined as:

$$\begin{aligned} X_H[k] = \mathcal{H}[x[m]] &= \sum_{m=0}^{M-1} x[m] cas\left(2\pi\frac{mk}{M}\right) \\ &= \frac{1}{\sqrt{M}}\sum_{m=0}^{M-1}\left[\cos\left(2\pi\frac{mk}{M}\right) + \sin\left(2\pi\frac{mk}{M}\right)\right] \end{aligned} \tag{6.23}$$

Here $X_H[k]$ $(k = 0, \cdots, M-1)$ are M frequency components of the signal, similar to the case of the discrete Fourier transform. Due to the orthogonality of $\boldsymbol{\phi}_k$ and following the same method used to derive Eq.6.8, we get the inverse transform by which the signal can be reconstructed:

$$x[m] = \mathcal{H}^{-1}[X_H[k]] = \frac{1}{\sqrt{M}}\sum_{k=0}^{M-1} X[k] cas\left(2\pi\frac{mk}{M}\right) \tag{6.24}$$

Same as in the continuous case in Eq.6.13, the discrete Hartley transform is closely related to the discrete Fourier transform:

$$\begin{aligned} X_F[k] = \mathcal{F}[x[m]] &= \frac{1}{\sqrt{M}}\sum_{m=0}^{M-1} x[m] e^{-j2\pi mk/M} = \frac{1}{\sqrt{M}}\sum_{m=0}^{M-1}\left[\cos\left(2\pi\frac{mk}{M}\right) - j\,\sin\left(2\pi\frac{mk}{M}\right)\right] \\ &= X_e[k] - j\, X_o[k], \qquad (k = 0, \cdots, M-1) \end{aligned}$$

where

$$\begin{aligned} X_e[k] = Re[X_F[k]] &= \frac{1}{\sqrt{M}}\sum_{m=0}^{M-1} x[m]\cos\left(2\pi\frac{mk}{M}\right) \\ X_o[k] = -Im[X_F[k]] &= \frac{1}{\sqrt{M}}\sum_{m=0}^{M-1} x[m]\sin\left(2\pi\frac{mk}{M}\right) \end{aligned}$$

and the discrete Hartley spectrum can also be obtained from the discrete Fourier transform:

$$X_H[k] = \mathcal{H}[x[m]] = X_e[k] + X_o[k] = Re[X_F[k] - Im[X_F[k]]] \tag{6.25}$$

Example 6.2: As considered before, the DFT of a 8-D signal vector $\boldsymbol{x} = [0, 0, 2, 3, 4, 0, 0, 0]^T$ is (Eq.4.147) $\boldsymbol{X} = \boldsymbol{X}_r + j\boldsymbol{X}_j$ where:

$$\begin{aligned} \boldsymbol{X}_r &= \big[\,3.18, -2.16, 0.71, -0.66, 1.06, -0.66, 0.71, -2.16\,\big]^T \\ \boldsymbol{X}_j &= \big[\,0.0, -1.46, 1.06, -0.04, 0.0, 0.04, -1.06, 1.46\,\big]^T \end{aligned} \tag{6.26}$$

The discrete Hartley transform of this signal vector is:

$$\boldsymbol{X}_H = \boldsymbol{X}_r - \boldsymbol{X}_j = [3.18, -0.71, -0.35, -0.62, 1.06, -0.71, 1.77, -3.62]^T$$

The inverse Hartley transform will convert this spectrum $\boldsymbol{X}_H$ back to the original signal.

6.1.5 2-D Hartley Transform

Similar to the 2-D Fourier transform, the 2-D Hartley transform of a signal array $x[m, n]$ $(0 \le m \le M-1,\ 0 \le n \le N-1)$ can be defined as:

$$X[k, l] = \mathcal{H}[x[m, n]] = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m, n]\phi_{k,l}[m, n] \tag{6.27}$$

where $\phi_{k,l}[m, n]$ is a discrete 2-D kernel function. There exist two different definitions for the 2-D Hartley transform depending on which of the following 2-D kernel function is assumed:

$$\phi'_{k,l}[m, n] = cas(2\pi(mk/M + nl/N))/\sqrt{MN} \tag{6.28}$$

$$\phi''_{k,l}[m, n] = cas(2\pi mk/M)\ cas(2\pi nl/N)/\sqrt{MN} \tag{6.29}$$

Note that the second kernel is separable, i.e., it is a product of two 1-D kernels one for each of the two dimensions, while the first one is inseparable. As shown below, these two different kernel functions are very similar to each other:

$$\begin{aligned} & cas(2\pi\, mk/M)\ cas(2\pi\, nl/N) \\ &= [\cos(2\pi\, mk/M) + \sin(2\pi\, mk/M)]\,[\cos(2\pi\, nl/N) + \sin(2\pi\, nl/N)] \\ &= [\cos(2\pi\, mk/M)\cos(2\pi\, nl/N) + \sin(2\pi\, mk/M)\sin(2\pi\, nl/N)] \\ &\quad [\sin(2\pi\, mk/M)\cos(2\pi\, nl/N) + \cos(2\pi\, mk/M)\sin(2\pi\, nl/N)] \\ &= \cos(2\pi(mk/M - nl/N)) + \sin(2\pi(mk/M + nl/N)) \\ &\neq \cos(2\pi(mk/M + nl/N)) + \sin(2\pi(mk/M + nl/N)) \\ &= cas(2\pi(mk/M + nl/N)) \end{aligned} \tag{6.30}$$

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:

$$\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\phi_{k,l}[m,n]\phi_{k',l'}[m,n] = \delta[k-k', l-l'] \tag{6.31}$$

therefore either of which can be used for the 2-D Hartley transform.

- First we consider $\phi'_{k,l}[m,n] = cas(2\pi(mk/M + nl/N))$. The forward transform is:

$$\begin{aligned} X'_H[k,l] &= \frac{1}{\sqrt{MN}}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} x[m,n]cas(2\pi(mk/M+nl/N)) \\ &= \frac{1}{\sqrt{MN}}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} x[m,n][\cos(2\pi(mk/M+nl/N)) + \sin(2\pi(mk/M+nl/N))] \end{aligned} \tag{6.32}$$

  This Hartley transform can be compared with the 2-D Fourier transform:

$$\begin{aligned} X_F[k,l] &= \frac{1}{\sqrt{MN}}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} x[m,n]e^{-2\pi(mk/M+nl/N)} \\ &= \frac{1}{\sqrt{MN}}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} x[m,n][\cos(2\pi(mk/M+nl/N)) - j\sin(2\pi(mk/M+nl/N))] \\ &= X_e[k,l] - jX_o[k,l] = Re[X_e[k,l]] + jIm[X_o[k,l]] \end{aligned} \tag{6.33}$$

  where

$$X_e[k,l] = Re[X_F[k,l]] = \frac{1}{\sqrt{MN}}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} x[m,n]\cos(2\pi(mk/M+nl/N))$$

$$X_o[k,l] = -Im[X_F[k,l]] = \frac{1}{\sqrt{MN}}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} x[m,n]\sin(2\pi(mk/M+nl/N))$$

  are respectively the 2-D even and odd components of $X_F[k,l]$. We can see the same relationship between the Hartley and Fourier transforms as in 1-D case:

$$X'_H[k,l] = X_e[k,l] + X_o[k,l] = Re[X_F[k,l]] - Im[X_F[k,l]] \tag{6.34}$$

  Extending the orthogonality in Eq.6.22 from 1-D to 2-D, we get:

$$\frac{1}{MN}\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} cas(2\pi(mk/M+nl/N))cas(2\pi(mk'/M+nl'/N)) = \delta[k-k', l-l'] \tag{6.35}$$

Based on this orthogonality and following the same method used to derive Eq.6.24, we get the inverse transform by which the signal can be reconstructed:

$$x[m,n] = \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} X'_H[k,l] cas(2\pi(mk/M + nl/N)) \tag{6.36}$$

- Next we consider $\phi''_{k,l}[m,n] = cas(2\pi\, mk/M)cas(2\pi\, nl/N)$, which is is separable, same as the 2-D Fourier kernel $e^{j2\pi(mk/M+nl/N)} = e^{j2\pi(mk/M)} e^{j2\pi(nl/N)}$, therefore the 2-D transform can also be carried out in two stages of 1-D transforms for each of the two dimensions:

$$\begin{aligned} X''_H[k,l] &= \frac{1}{\sqrt{MN}} \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} x[m,n] cas(2\pi\, mk/M) cas(2\pi\, nl/N) \\ &= \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} \left[\frac{1}{\sqrt{M}} \sum_{m=0}^{M-1} x[m,n] cas(2\pi\, mk/M) \right] cas(2\pi\, nl/N) \end{aligned} \tag{6.37}$$

According to Eq.6.30, this transform can be further written as:

$$\begin{aligned} X''_H[k,l] &= \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m,n][\cos(2\pi(mk/M - nl/N)) + \sin(2\pi(mk/M + nl/N))] \\ &= X_e[k,-l] + X_o[k,l] = Re[X_F[k,-l]] - Im[X_F[k,l]] \end{aligned} \tag{6.38}$$

Similarly the inverse transform can also be carried out in two stages

$$\begin{aligned} x[m,n] &= \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} X''_H[k,l] cas(2\pi\, mk/M) cas(2\pi\, nl/N) \\ &= \frac{1}{\sqrt{N}} \sum_{m=0}^{M-1} \left[\frac{1}{\sqrt{M}} \sum_{n=0}^{N-1} X''_H[k,l] cas(2\pi\, mk/M) \right] cas(2\pi\, nl/N) \end{aligned} \tag{6.39}$$

Note again that the inverse transform in either Eq.6.36 or Eq.6.39 is identical to the forward transform. Also, to better compare the two versions of the 2-D Hartley transform, we put Eqs.6.34 and 6.38 side by side:

$$\begin{aligned} X'_H[k,l] &= X_e[k,l] + X_o[k,l] = Re[X_F[k,l]] - Im[X_F[k,l]] \\ X''_H[k,l] &= X_e[k,-l] + X_o[k,l] = Re[X_F[k,-l]] - Im[X_F[k,l]] \end{aligned}$$

We see 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 $X_e[k,-l] = X_e[k,N-l]$.

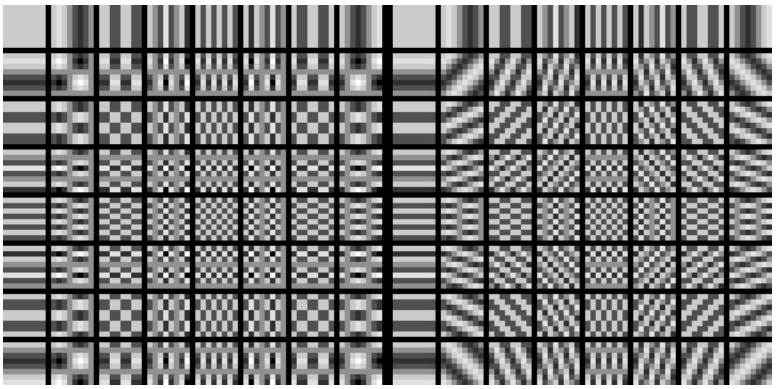

Figure 6.3 The 8 by 8 basis functions for the 2-D Hartley transform
The left half of the image shows the basis functions based on the separable kernel $\phi''_{k,l}[m,n]$, and the right half based on the inseparable kernel $\phi'_{k,l}[m,n]$. The DC component is at the top-left corner, and the highest frequency component in both horizontal and vertical directions is at the middle, same as the 2-D Fourier basis.

Example 6.3: Given a 2-D signal array:

$$\boldsymbol{x} = \begin{bmatrix} 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 70.0 & 80.0 & 90.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 90.0 & 100.0 & 110.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 110.0 & 120.0 & 130.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 130.0 & 140.0 & 150.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \end{bmatrix}$$

The Hartley spectrum corresponding to inseparable kernel $\phi'_{k,l}[m,n]$ is:

$$\boldsymbol{X}' = \begin{bmatrix} 165.0 & -10.0 & -45.0 & -32.2 & 55.0 & -10.0 & 65.0 & -187.8 \\ 27.4 & -100.5 & 54.8 & 6.3 & 9.1 & -15.2 & -47.7 & 65.8 \\ 0.0 & 17.1 & -10.0 & -2.9 & 0.0 & 2.9 & 10.0 & -17.1 \\ -26.4 & -15.2 & 17.7 & 7.1 & -8.8 & -0.5 & -20.6 & 46.6 \\ 15.0 & 0.0 & -5.0 & -2.9 & 5.0 & 0.0 & 5.0 & -17.1 \\ -57.4 & 20.2 & 5.3 & 9.2 & -19.1 & 5.5 & -12.3 & 48.7 \\ 30.0 & -17.1 & 0.0 & -2.9 & 10.0 & -2.9 & 0.0 & -17.1 \\ -153.6 & 105.5 & -17.7 & 18.4 & -51.2 & 20.2 & 0.6 & 77.9 \end{bmatrix}$$

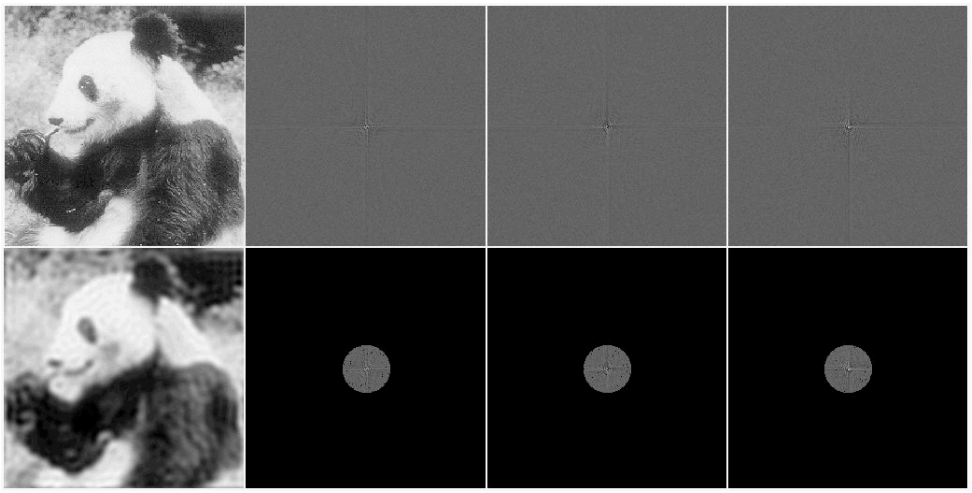

Figure 6.4 The Hartley and Fourier filtering of an image
The image and its Fourier and Hartley spectra before and after a low-pass filtering are shown in the top and bottom rows, respectively. The second and third panel of each row are the real and imaginary parts of the Fourier spectrum, while the forth panel is for the Hartley spectrum.

The Hartley spectrum corresponding to separable kernel $\phi''_{k,l}[m,n]$ is:

$$\boldsymbol{X}'' = \begin{bmatrix} 165.0 & -10.0 & -45.0 & -32.2 & 55.0 & -10.0 & 65.0 & -187.8 \\ 27.4 & -3.5 & -5.6 & -5.4 & 9.1 & -3.5 & 12.7 & -31.2 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ -26.4 & 1.5 & 7.3 & 5.2 & -8.8 & 1.5 & -10.3 & 30.0 \\ 15.0 & 0.0 & -5.0 & -2.9 & 5.0 & 0.0 & 5.0 & -17.1 \\ -57.4 & 3.5 & 15.6 & 11.2 & -19.1 & 3.5 & -22.7 & 65.4 \\ 30.0 & 0.0 & -10.0 & 5.9 & 10.0 & 0.0 & 10.0 & -34.1 \\ -153.6 & 8.5 & 42.7 & 30.0 & -51.2 & 8.5 & -59.8 & 174.9 \end{bmatrix}$$

In either case, the signal is perfectly reconstructed by the inverse transform (identical to the forward transform) corresponding to each of the two kernels.

Example 6.4: An image and both of its Fourier and Hartley spectra are shown in the top row of Fig.6.4. 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.

6.2 The Discrete Cosine Transform

In general, the discrete Fourier transform (DFT) converts a complex signal into its complex spectrum. If the signal is real, as in most of the applications, the imaginary part of the signal is all zero, and its spectrum is symmetric (real part is even and imaginary part odd), i.e., half of the data is redundant in frequency domain as well as in time domains. Consequently, half of the computational time and storage space in the transform is unnecessary.

Such redundancy can be avoided in the discrete cosine transform (DCT) which transforms a real signal into its real spectrum. Also, as the DCT can be derived from the DFT, the fast algorithm FFT can still be used for the DCT computation.

6.2.1 Fourier Cosine Transform

Let us first review the Fourier transform of a real signal $x(t)$:

$$\begin{aligned} X(f) &= \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt = \int_{-\infty}^{\infty} x(t)[\cos(2\pi ft) - j\sin(2\pi ft)]dt \\ &= X_r(f) - jX_j(f) \end{aligned} \tag{6.40}$$

where the real part of the spectrum $X_r(f)$ is even and the imaginary part $X_j(f)$ is odd:

$$\begin{aligned} X_r(f) &= \int_{-\infty}^{\infty} x(t)\cos(2\pi ft)dt = X_r(-f) \\ X_j(f) &= \int_{-\infty}^{\infty} x(t)\sin(2\pi ft)dt = -X_j(-f) \end{aligned} \tag{6.41}$$

Moreover, if we can further assume the signal is even, i.e., $x(t) = x(-t)$, then $X_j(f) = 0$ as the integral of the odd integrand $x(t)\sin(2\pi ft)$ is zero. Now the Fourier transform becomes a real cosine transform:

$$X(f) = \int_{-\infty}^{\infty} x(t)\cos(2\pi ft)dt = 2\int_{0}^{\infty} x(t)\cos(2\pi ft)dt = X(-f) \tag{6.42}$$

The second equal sign is due to the fact that the integrand $x(t)\cos(2\pi ft)$ is even with respect to t. This spectrum $X(f)$ is real and even with respect to f. The inverse transform becomes:

$$\begin{aligned} x(t) &= \int_{-\infty}^{\infty} X(f)e^{j2\pi ft}df = \int_{-\infty}^{\infty} X(f)\cos(2\pi ft)df + j\int_{-\infty}^{\infty} X(f)\sin(2\pi ft)df \\ &= 2\int_{0}^{\infty} X(f)\cos(2\pi ft)df \end{aligned} \tag{6.43}$$

The last equal sign is due to the fact that the integrands $X(f)\cos(2\pi ft)$ and $X(f)\sin(2\pi ft)$ are even and odd respectively. We see that now both the forward and inverse cosine transforms involve only real operations. As a real transform,

the cosine transform is computationally more advantageous compared to the Fourier transform.

Of course this cosine transform is valid only if the signal of interest is even. However, if the signal is not even but it is known to be zero before a certain moment, i.e., $x(t) = x(t)u(t)$, we can construct an even signal:

$$x'_e(t) = \begin{cases} x(t) & t \geq 0 \\ x(-t) & t \leq 0 \end{cases}$$

so that the cosine transform can still be used.

Obviously if we can assume or construct an odd signal $x(t) = -x(-t)$, we can also derive in a similar manner the sine transform.

The consideration above for the continuous signals can be extended to discrete signals of a finite duration. The corresponding cosine transform is called the discrete cosine transform (DCT). However, different from the continuous case, here we have more than one way to construct an even signal based on a set of finite data samples $x[0], \cdots, x[N-1]$. For example, by assuming $x[-m] = x[m]$, we can obtain a sequence of $2N-1$ samples that is even with respect to the point $m = 0$. Alternatively, we could also let $x[-m] = x[m-1]$, i.e., $x[-1] = x[0]$, $x[-2] = x[1]$, and $x[-N] = x[N-1]$ to get a sequence of $2N$ samples that is even with respect to the point $m = -1/2$. 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 second approach to construct a sequence of $2N$ points and assume it is periodic beyond its two ends. Then the DCT can be derived by applying the DFT to this even sequence of 2N points.

6.2.2 From Discrete Fourier Transform to Discrete Cosine Transform

We now derive the DCT of an N-point discrete signal, based on the 2N-point DFT, as described above. First, given an N-point real signal sequence $x[0], \cdots, x[N-1]$, we construct a new sequence of $2N$ points:

$$x'[m] = \begin{cases} x[m] & (0 \leq m \leq N-1) \\ x[-m-1] & (-N \leq m \leq -1) \end{cases} \tag{6.44}$$

This 2N-point sequence $x'[m]$ is assumed to repeat itself outside the range $-N \leq n \leq N-1$, i.e., it is periodic with period $2N$:

$$x'[m] = x'[-m-1] = x'[2N-m-1] \tag{6.45}$$

In the following we simply denote this constructed sequence by $x[m]$. Note that this signal $x'[m]$ is even with respect to the point $m = -1/2$. If we shift it to the right by $1/2$, or, equivalently, if we define a new index $m' = m + 1/2$, i.e., $m = m' - 1/2$, then the function $x[m] = x[m' - 1/2]$ is even with respect to $m' = 0$.

Fig. 6.5 shows a discrete signal of $N = 4$ data points $x[0], \cdots, x[3]$. A new signal is then constructed by including $N = 4$ additional points $x[-1] = x[0], \cdots, x[-4] = x[3]$. This signal of $2N = 8$ points is even with respect to

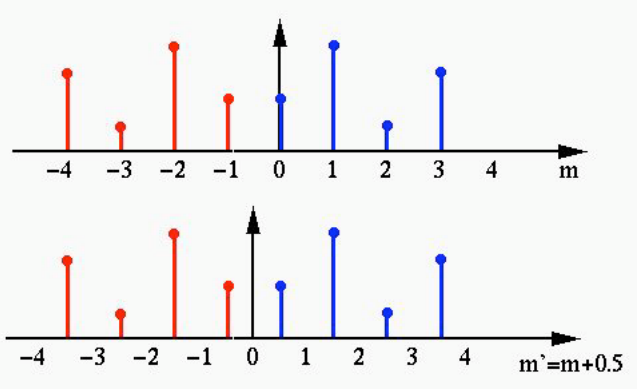

Figure 6.5 Formulation of DCT

$m = 1/2$. After define $m' = m + 1/2$, these points $x[m' - 1/2]$ (from $-N + 1/2 = -3.5$ to $N - 1/2 = 3.5$) are even with respect to $m' = 0$.

Now we can apply a 2N-point DFT to this constructed even signal of 2N points and get:

$$\begin{aligned} X[n] &= \frac{1}{\sqrt{2N}} \sum_{m'=-N+1/2}^{N-1/2} x[m' - \frac{1}{2}] e^{-j2\pi m' n/2N} \\ &= \frac{1}{\sqrt{2N}} \sum_{m'=-N+1/2}^{N-1/2} x[m' - \frac{1}{2}] \cos\left(\frac{2\pi m' n}{2N}\right) \\ &\quad - \frac{j}{\sqrt{2N}} \sum_{m'=-N+1/2}^{N-1/2} x[m' - \frac{1}{2}] \sin\left(\frac{2\pi m' n}{2N}\right) \qquad (n = 0, \cdots, 2N - 1) \end{aligned} \tag{6.46}$$

Note that as $\cos(2\pi m' n/2N)$ and $\sin(2\pi m' n/2N)$ are even and odd, respectively, and $x[m' - 1/2]$ is even, all with respect to $m' = 0$, the terms in the first summation are even while those in the second summation are odd. Consequently, the first summation of $2N$ terms is equal to twice the sum of the first N terms, and the second summation is simply zero, and we have

$$X[n] = \sqrt{\frac{2}{N}} \sum_{m'=1/2}^{N-1/2} x[m' - \frac{1}{2}] \cos\left(\frac{2\pi m' n}{2N}\right) \qquad (n = 0, \cdots, 2N - 1) \tag{6.47}$$

Note that $X[n] = X[-n]$ is real, even (as $\cos(-2\pi m' n/2N) = \cos(2\pi m' n/2N)$), and periodic with period $2N$. Specifically, we have $X[N + n] = X[N + n - 2N] = X[-N + n] = X[N - n]$, indicating a point $X[N + n]$ in the second half of the $2N$ coefficients is equal to its corresponding point $X[N - n]$ in the first half, for all $n = 0, 1, \cdots, N - 1$. In other words, the range for the index n in the equation above can be from 0 to $N - 1$, as the second half is redundant and can therefore be dropped. Finally, replacing m' by $m + 1/2$, we get the discrete

cosine transform (DCT):

$$X[n] = \sqrt{\frac{2}{N}} \sum_{m=0}^{N-1} x[m] \cos\left(\frac{(2m+1)n\pi}{2N}\right), \qquad (n = 0, \cdots, N-1) \tag{6.48}$$

We compare the DCT discussed above with the DFT considered in previous chapters to realize the following advantages:

- The DCT is a real transform with better computational efficiency than the complex DFT as no complex operations are needed.
- The DCT does not introduce discontinuity while truncating and imposing periodicity on the time signal. To perform the DFT of a physical signal, it needs to be truncated to have a finite duration $0 \leq t \leq T$, and assumed periodic with period T beyond this range. In this process, discontinuity is inevitably introduced in time domain and some corresponding artifacts, most likely some high frequency components, are introduced in frequency domain. But as even symmetry is assumed in the case of DCT while truncating the time signal, no discontinuity and related artifacts are introduced in DCT.
- The DCT coefficient $X[n]$ given in Eq.6.48 corresponds to a sinusoid $cos(2\pi f_n(m+1/2))$ of frequency $f_n = n/2N$, which is half of the frequency $f_n = n/N$ represented by the nth DFT coefficient:

  $$X[n] = \sum_{m=0}^{N-1} x[m] exp(j2\pi mn/N) = \sum_{m=0}^{N-1} x[m][\cos(2\pi f_n m) + j\sin(2\pi f_n m)]$$

  Both the DCT and DFT spectra contain N coefficients. However, half of the N DFT coefficients represents the negative frequencies of the complex exponentials, while each of the N DCT coefficients represents a different frequency of a sinusoid. Both DCT and DFT spectra represent the same frequency range but the resolution of the DCT spectrum is twice that of the DFT.

6.2.3 Discrete Cosine Transform in Matrix Form

An N by N DCT matrix $\boldsymbol{C}$ can be constructed by defining its element of the mth row and nth column as:

$$c[m,n] = cos(\frac{(2m+1)n\pi}{2N}) = cos(\frac{(m+1/2)n\pi}{N}), \qquad (m,n = 0,1,\cdots,N-1) \tag{6.49}$$

This matrix can also be expressed in terms of its N column vectors

$$\boldsymbol{C} = \begin{bmatrix} c[0,0] & \cdots & c[0,N-1] \\ \vdots & \ddots & \vdots \\ c[0,N-1] & \cdots & c[N-1,N-1] \end{bmatrix} = [\boldsymbol{c}_0 \cdots \boldsymbol{c}_{N-1}] \tag{6.50}$$

where the nth column $\boldsymbol{c}_n$ of this matrix is:

$$\boldsymbol{c}_n = \left[\cos\left(\frac{n\pi}{2N}\right), \cos\left(\frac{3n\pi}{2N}\right), \cos\left(\frac{5n\pi}{2N}\right), \cdots, \cos\left(\frac{(2N-1)n\pi}{2N}\right)\right]^T \tag{6.51}$$

We can show that all column vectors of $\boldsymbol{C}$ are orthogonal:

$$< \boldsymbol{c}_k, \boldsymbol{c}_n >= 0, \quad \text{for any } n \neq k \tag{6.52}$$

We first show the following is true:

$$\sum_{m=0}^{N-1} \cos(\frac{(2m+1)k\pi}{2N}) = \begin{cases} N & k = 0 \\ 0 & k \neq 0 \end{cases} \tag{6.53}$$

Obviously when $k = 0$, all N cosine functions are zero and the summation is indeed N. We only need to show the summation is zero when $k \neq 0$. Consider:

$$\begin{aligned} &\sum_{m=0}^{N-1} \cos(\frac{(2m+1)k\pi}{2N}) = \frac{1}{2}\sum_{m=0}^{N-1}[e^{j(2m+1)k\pi/2N} + e^{-j(2m+1)k\pi/2N}] \\ &= \frac{1}{2}[e^{jk\pi/2N}\sum_{m=0}^{N-1}(e^{jk\pi/N})^m + e^{-jk\pi/2N}\sum_{m=0}^{N-1}(e^{-jk\pi/N})^m] \\ &= \frac{1}{2}[e^{jk\pi/2N}\frac{1-e^{jk\pi}}{1-e^{jk\pi/N}} + e^{-jk\pi/2N}\frac{1-e^{-jk\pi}}{1-e^{-jk\pi/N}}] \end{aligned} \tag{6.54}$$

Here we have used the identity $\sum_{m=0}^{N-1} x^m = (1-x^N)/(1-x)$. When k is even, $e^{jk\pi} = e^{-jk\pi} = 1$ and the numerators of both fractions are zero. When k is odd, $e^{jk\pi} = e^{-jk\pi} = -1$, and the above becomes:

$$\begin{aligned} &\frac{1}{2}[\frac{2}{e^{-jk\pi/2N}(1-e^{jk\pi/N})} + \frac{2}{e^{jk\pi/2N}(1-e^{-jk\pi/N})}] \\ &= \frac{1}{e^{-jk\pi/2N} - e^{jk\pi/2N}} + \frac{1}{e^{jk\pi/2N} - e^{-jk\pi/2N}} = 0 \end{aligned} \tag{6.55}$$

Now we can consider the inner product of the kth and nth columns of $\boldsymbol{C}$:

$$\begin{aligned} &< \boldsymbol{c}_k, \boldsymbol{c}_n >= \sum_{m=0}^{N-1} c[m,k]c[m,n] = \sum_{m=0}^{N-1} \cos(\frac{(2m+1)k\pi}{2N})\cos(\frac{(2m+1)n\pi}{2N}) \\ &= \frac{1}{2}\sum_{m=0}^{N-1} \cos(\frac{(2m+1)(k+n)\pi}{2N}) + \frac{1}{2}\sum_{m=0}^{N-1} cos(\frac{(2m+1)(k-n)\pi}{2N}) \end{aligned} \tag{6.56}$$

Here we have used the trigonometric identity $\cos\alpha\cos\beta = [\cos(\alpha+\beta) + \cos(\alpha-\beta)]/2$. When $k \neq n$, both terms are zero according to Eq. 6.53, i.e., the inner product is zero, indicating the column vectors of $\boldsymbol{C}$ are orthogonal. When $n = k$,

the inner product becomes the nth column's norm squared:

$$< \boldsymbol{c}_n, \boldsymbol{c}_n >= ||\boldsymbol{c}_m||^2 = \frac{1}{2}\sum_{m=0}^{N-1}\cos(\frac{(2m+1)2n\pi}{2N}) + \frac{1}{2}\sum_{m=0}^{N-1}\cos(\frac{(2m+1)0\pi}{2N})$$

$$= \begin{cases} N & \text{if } n=0 \\ N/2 & \text{otherwise} \end{cases} \tag{6.57}$$

This is because the first term is either $N/2$ zero if $n = 0$ or zero if $n \neq 0$, while the second term is always $N/2$. In order to make all columns of $\boldsymbol{C}$ normalized, we define a coefficient $a[n]$:

$$a[n] = \begin{cases} \sqrt{1/N} & \text{if } n=0 \\ \sqrt{2/N} & \text{otherwise} \end{cases} \tag{6.58}$$

and multiply $\boldsymbol{C}$ by $a[n]$, so that the columns of the modified version of the DCT matrix, still denoted by $\boldsymbol{C}$, are orthonormal:

$$< \boldsymbol{c}_k, \boldsymbol{c}_n >= \delta[k-n] = \begin{cases} 1 & k=n \\ 0 & k \neq n \end{cases} \tag{6.59}$$

and they can therefore used as a basis of the N-D space $\mathbb{R}^N$, and the modified DCT matrix $\boldsymbol{C}$ becomes orthogonal:

$$\boldsymbol{C}^T = \boldsymbol{C}^{-1}, \quad \text{i.e.} \quad \boldsymbol{C}^T\boldsymbol{C} = \boldsymbol{I} \tag{6.60}$$

Expressing both the discrete signal $x[m]$ $(m = 0, 1, \cdots, N-1)$ and its DCT transform coefficients $X[n]$ $(n = 0, 1, \cdots, N-1)$ as vectors:

$$\boldsymbol{x} = [x[0], \cdots, x[N-1]]^T, \qquad \boldsymbol{X} = [X[0], \cdots, X[N-1]]^T \tag{6.61}$$

we can represent the forward DCT as a matrix multiplication:

$$\boldsymbol{X} = \boldsymbol{C}^T\boldsymbol{x} = \begin{bmatrix} \boldsymbol{c}_0^T \\ \vdots \\ \boldsymbol{c}_{N-1}^T \end{bmatrix} \boldsymbol{x} \tag{6.62}$$

The nth component $X[n]$ of $\boldsymbol{X}$ is actually the projection of the signal vector $\boldsymbol{x}$ onto the nth basis vector $\boldsymbol{c}_n$:

$$X[n] = < \boldsymbol{c}_n, \boldsymbol{x} >= \boldsymbol{c}_n^T\boldsymbol{x} = \sum_{m=0}^{N-1} c[n,m]x[m]$$

$$= a[n]\sum_{m=0}^{N-1} x[m]\, cos(\frac{(2m+1)n\pi}{2N}), \quad (n = 0, \cdots, N-1) \tag{6.63}$$

which is the same as Eq. 6.48 derived previously as $a[n] = \sqrt{2/N}$ for all $n = 1, \cdots, N-1$, except when $n = 0$ we have a different scaling constant $a[0] =$

$1/\sqrt{N}$:

$$X[0] = \sum_{m=0}^{N-1} c[0,m]x[m] = \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} x[m] \tag{6.64}$$

representing the DC component of the signal.

The inverse DCT can be obtained by pre-multiplying $\boldsymbol{C}$ on both sides of Eq. 6.62 for the forward DCT. As $\boldsymbol{C}\boldsymbol{C}^T = \boldsymbol{C}\boldsymbol{C}^{-1} = \boldsymbol{I}$, we get

$$\boldsymbol{x} = \boldsymbol{C}\boldsymbol{X} = [\boldsymbol{c}_0, \cdots, \boldsymbol{c}_{N-1}] \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \sum_{n=0}^{N-1} X[n]\boldsymbol{c}_n \tag{6.65}$$

i.e., the signal vector $\boldsymbol{x}$ is expressed as a linear combination of the basis vectors $\boldsymbol{c}_n$, $(n = 0, 1, \cdots, N-1)$. The inverse DCT can also be expressed in component form for $x[m]$ $(m = 0, \cdots, N-1)$:

$$x[m] = \sum_{n=0}^{N-1} c[m,n]X[n] = \sum_{n=0}^{N-1} X[n]a[n]\cos(\frac{(2m+1)n\pi}{2N}) \tag{6.66}$$

Putting Eqs. 6.63 and 6.66 together, we have the DCT pairs:

$$\begin{aligned} X[n] &= a[n] \sum_{m=0}^{N-1} x[m]\ cos(\frac{(2m+1)n\pi}{2N}), \quad (n = 0, \cdots, N-1) \\ x[m] &= \sum_{n=0}^{N-1} X[n]a[n]\cos(\frac{(2m+1)n\pi}{2N}), \quad (m = 0, \cdots, N-1) \end{aligned} \tag{6.67}$$

In matrix form, the DCT pair can be written as:

$$\begin{cases} \boldsymbol{X} = \boldsymbol{C}^T\boldsymbol{x} & \text{(forward)} \\ \boldsymbol{x} = \boldsymbol{C}\boldsymbol{X} & \text{(inverse)} \end{cases} \tag{6.68}$$

As a specific example, consider the 2-point DCT matrix:

$$\boldsymbol{C}_{2\times 2}^T = \begin{bmatrix} \boldsymbol{c}_0^T \\ \boldsymbol{c}_2^T \end{bmatrix} = \begin{bmatrix} \sqrt{1/2} & \sqrt{1/2} \\ \cos(\pi/4) & \cos(3\pi/4) \end{bmatrix} = 0.707 \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{6.69}$$

This matrix is composed of two row vectors $\boldsymbol{c}_0^T = [1\ \ 1]/\sqrt{2}$ and $\boldsymbol{c}_1^T = [1\ \ -1]/\sqrt{2}$ and is identical to the 2-point DFT matrix $\boldsymbol{W}_{2\times 2}$ discussed before. The DCT of a 2-point signal $\boldsymbol{x} = [x[0], x[1]]^T$ is

$$\boldsymbol{X} = \begin{bmatrix} X[0] \\ X[1] \end{bmatrix} = 0.707 \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \end{bmatrix} = 0.707 \begin{bmatrix} x[0] + x[1] \\ x[1] - x[1] \end{bmatrix} \tag{6.70}$$

The first component $X[0]$ is proportional to the sum $x[0] + x[1]$ of the two signal 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.

When $N = 4$, the 4-point DCT matrix is:

$$\boldsymbol{C}_{4\times 4}^T = \begin{bmatrix} \boldsymbol{c}_0^T \\ \boldsymbol{c}_1^T \\ \boldsymbol{c}_2^T \\ \boldsymbol{c}_3^T \end{bmatrix} = \begin{bmatrix} 0.50 & 0.50 & 0.50 & 0.50 \\ 0.65 & 0.27 & -0.27 & -0.65 \\ 0.50 & -0.50 & -0.50 & 0.50 \\ 0.27 & -0.65 & 0.65 & -0.27 \end{bmatrix} \tag{6.71}$$

which is composed of four row vectors, corresponding to four sinusoids with progressively higher frequencies.

Example 6.5: Find the DCT of a real 8-point signal: $\boldsymbol{x} = [0, 0, 2, 3, 4, 0, 0, 0]^T$. First we find the 8-point matrix $\boldsymbol{C}^T$:

$$\boldsymbol{C}_{8\times 8}^T = \begin{bmatrix} \boldsymbol{c}_0^T \\ \vdots \\ \boldsymbol{c}_7^T \end{bmatrix} = \begin{bmatrix} 0.35 & 0.35 & 0.35 & 0.35 & 0.35 & 0.35 & 0.35 & 0.35 \\ 0.49 & 0.42 & 0.28 & 0.10 & -0.10 & -0.28 & -0.42 & -0.49 \\ 0.46 & 0.19 & -0.19 & -0.46 & -0.46 & -0.19 & 0.19 & 0.46 \\ 0.42 & -0.10 & -0.49 & -0.28 & 0.28 & 0.49 & 0.10 & -0.42 \\ 0.35 & -0.35 & -0.35 & 0.35 & 0.35 & -0.35 & -0.35 & 0.35 \\ 0.28 & -0.49 & 0.10 & 0.42 & -0.42 & -0.10 & 0.49 & -0.28 \\ 0.19 & -0.46 & 0.46 & -0.19 & -0.19 & 0.46 & -0.46 & 0.19 \\ 0.10 & -0.28 & 0.42 & -0.49 & 0.49 & -0.42 & 0.28 & -0.10 \end{bmatrix} \tag{6.72}$$

The $N = 8$ values of the nth row vector $\boldsymbol{c}_n^T$ $(n = 0, 1, \cdots, 7)$ can be considered as eight samples of the corresponding continuous cosine function $b_n(t) = a[n]\cos((2t+1)n\pi)/2N)$ shown in the left column in Fig.6.6, with progressively higher frequencies as the index n increases. The the DCT of $\boldsymbol{x}$ can be found by a matrix multiplication:

$$\boldsymbol{X} = \boldsymbol{C}^T\boldsymbol{x} = [3.18, 0.46, -3.62, -0.70, 1.77, -0.22, -0.42, 1.32]^T. \tag{6.73}$$

The interpretation of these DCT coefficients is very straight forward. $X[0]$ represents the DC component or the average of the signal, while the subsequent coefficients $X[n]$ represent the magnitudes of progressively high frequency components contained in the signal.

The inverse DCT represents the signal vector as a linear combination of the eight column vectors of $\boldsymbol{C} = [\boldsymbol{c}_0, \cdots, \boldsymbol{c}_7]$, which form a set of orthonormal basis vectors that span the 8-D vector space:

$$\boldsymbol{x} = \begin{bmatrix} x[0] \\ \vdots \\ x[7] \end{bmatrix} = \boldsymbol{C}\boldsymbol{X} = [\boldsymbol{c}_0, \cdots, \boldsymbol{c}_7] \begin{bmatrix} X[0] \\ \vdots \\ X[7] \end{bmatrix} = \sum_{n=0}^{7} X[n]\boldsymbol{c}_n \tag{6.74}$$

The reconstruction of the signal by the linear combination of the eight vectors is shown in Fig. 6.6

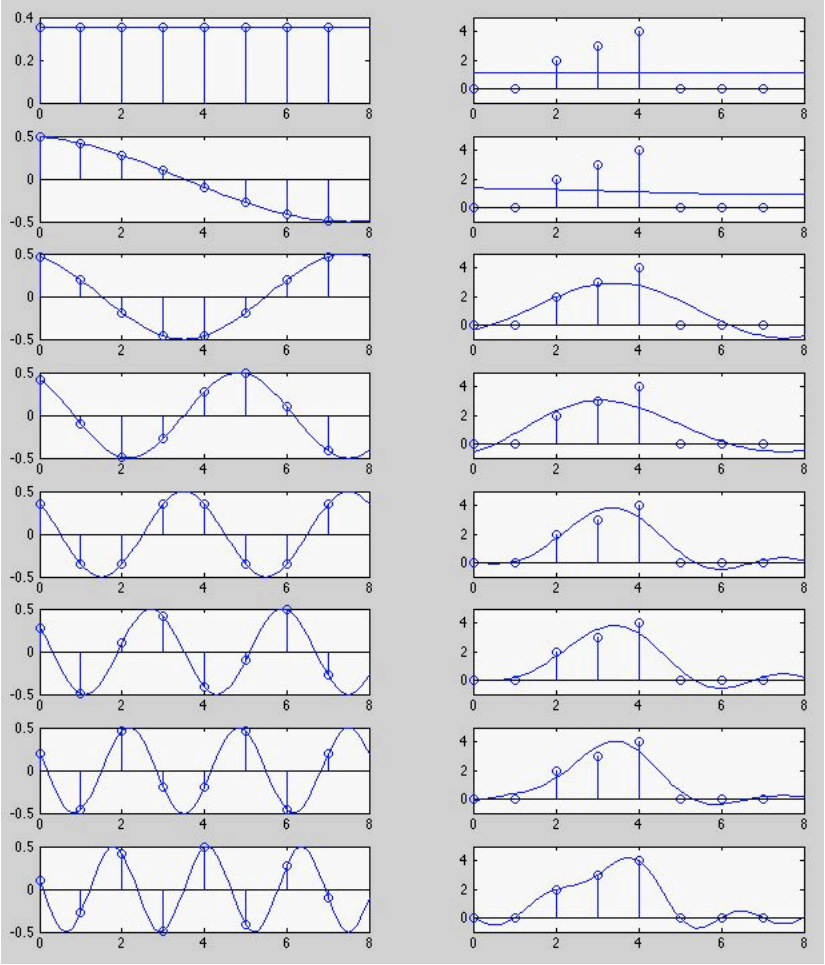

Figure 6.6 Basis functions of 8-point DCT and reconstruction of a signal
The left column shows the 8 continuous and discrete basis DCT functions while the right column shows how a discrete signal can be reconstructed by the inverse DCT as a linear combination of these basis functions weighted by WHT coefficients (Eq.6.73). From the top down, the plots on the right show the reconstructed signal as progressively more components of higher frequencies are included.

6.2.4 Fast DCT algorithm

If the DCT is implemented as a matrix multiplication, the computational complexity is $O(N^2)$ ($O(N)$ for each of the N coefficients $X[n]$. However, as the DCT is closely related to the DFT, it can also be implemented by the FFT algorithm with complexity $O(N\,log_2\,N)$.

We first define a new sequence $y[0], \cdots, y[N-1]$ based on the given signal $x[0], \cdots, x[N-1]$:

$$\begin{cases} y[m] = x[2m] \\ y[N-1-m] = x[2m+1] \end{cases} \qquad (m = 0, \cdots, N/2-1) \tag{6.75}$$

Note that the first half of $y[m]$ contains all the even components of $x[m]$, while the second half of $y[m]$ contains all the odd ones but in reverse order. The N-point DCT of the given signal $x[m]$ now becomes:

$$\begin{aligned} X[n] &= \sum_{m=0}^{N-1} x[m] cos(\frac{(2m+1)n\pi}{2N}) \\ &= \sum_{m=0}^{N/2-1} x[2m] cos(\frac{(4m+1)n\pi}{2N}) + \sum_{m=0}^{N/2-1} x[2m+1] cos(\frac{(4m+3)n\pi}{2N}) \\ &= \sum_{m=0}^{N/2-1} y[m] cos(\frac{(4m+1)n\pi}{2N}) + \sum_{m=0}^{N/2-1} y[N-1-m] cos(\frac{(4m+3)n\pi}{2N}) \end{aligned} \tag{6.76}$$

For simplicity, we temporarily dropped the scaling factor $a[n]$. Here the first summation is for all even terms and second all odd terms. We define $m' = N - 1 - m$ and rewrite the second summation as:

$$\sum_{m'=N/2}^{N-1} y[m'] cos(2n\pi - \frac{(4m'+1)n\pi}{2N}) = \sum_{m'=N/2}^{N-1} y[m'] cos(\frac{(4m'+1)n\pi}{2N}) \tag{6.77}$$

Now the two summations in the expression of $X[n]$ can be combined to become

$$X[n] = a[n] \sum_{m=0}^{N-1} y[m] cos(\frac{(4m+1)n\pi}{2N}) \tag{6.78}$$

We next consider the DFT of $y[m]$:

$$Y[n] = \sum_{m=0}^{N-1} y[m] e^{-j2\pi mn/N} \tag{6.79}$$

If we multiply both sides by $e^{-jn\pi/2N}$ and take the real part of the result, we get:

$$\begin{aligned} Re[e^{-jn\pi/2N} Y[n]] &= Re[\sum_{m=0}^{N-1} y[m] e^{-j2\pi mn/N} e^{-jn\pi/2N}] = Re[\sum_{m=0}^{N-1} y[m] e^{-j(4m+1)n\pi/2N}] \\ &= Re[\sum_{m=0}^{N-1} y[m] \, [\cos(\frac{(4m+1)n\pi}{2N}) - j \sin(\frac{(4m+1)n\pi}{2N})] \,] \\ &= \sum_{m=0}^{N-1} y[m] \cos(\frac{(4m+1)n\pi}{2N}) \end{aligned} \tag{6.80}$$

Note that the second term of the sine function is imaginary (as $y[m]$ is real) and has been dropped. As the right-hand side of this equation is identical to that of Eq. 6.78, we have

$$X[n] = Re[\, e^{-jn\pi/2N} Y[n]\,], \quad (n = 0, \cdots, N-1) \tag{6.81}$$

The DCT coefficient $X[n]$ for signal $x[m]$ can be obtained once the DFT coefficient $Y[n]$ for $y[m]$ is computed using the FFT algorithm with complexity $O(N\, log_2\, N)$.

In summary, the fast algorithm for forward DCT can be implemented in 3 steps:

- **Step 1:** Generate a sequence $y[m]$ from the given sequence $x[m]$:

$$\begin{cases} y[m] = x[2m] \\ y[N-1-m] = x[2m+1] \end{cases} \quad (m = 0, \cdots, N/2-1) \tag{6.82}$$

- **step 2:** Obtain DFT $Y[n]$ of $y[m]$ by FFT. As $y[m]$ is real, $Y[n]$ is symmetric and only half of the data points need be computed.

$$Y[n] = \mathcal{F}[y[m]], \quad (n = 0, \cdots, N-1) \tag{6.83}$$

- **step 3:** Obtain DCT $X[n]$ from $Y[n]$ $(n = 0, \cdots, N-1)$:

$$\begin{aligned} X[n] &= a[n]\, Re[\, e^{-jn\pi/2N} Y[n]\,] \\ &= a[n]\, [Y_r[n] \cos(n\pi/2N) + Y_i[n] \sin(n\pi/2N)] \end{aligned} \tag{6.84}$$

  where $Y_r[n]$ and $Y_i[n]$ are the real and imaginary part of $Y[n]$, respectively.

Note that the DCT scaling factor $a[n]$ is included in the third step, and no scaling factor (either $1/N$ or $1/\sqrt{N}$) is used during the DFT of $y[m]$.

Next we consider the inverse DCT. The most obvious way to do inverse DCT is to reverse the order and the mathematical operations of the three steps for the forward DCT:

- **step 1:** Obtain $Y[n]$ from $X[n]$ by solving the N equations in Eq. 6.84. There are N equations but $2N$ variables (both $Y_r[n]$ and $Y_i[n]$). However, note that as $y[m]$ is *real*, $Y_r[n]$ is even (N+1 independent variables) and $Y_i[n]$ is odd (N-1 independent variables with $Y_i[0] = Y_i[N/2] = 0$). So there are only N independent variables which can be obtained by solving the N equations.
- **step 2:** Obtain $y[m]$ from $Y[n]$ by inverse DFT also using FFT in $N\, log_2 N$ complexity.

$$y[m] = \mathcal{F}^{-1}[Y[n]] \tag{6.85}$$

- **step 3:** Obtain $x[m]$ from $y[m]$ by

$$\begin{cases} x[2m] = y[m] \\ x[2m+1] = y[N-1-m] \end{cases} \quad (i = 0, \cdots, N/2-1) \tag{6.86}$$

However, there is a more efficient way to do the inverse DCT without the need to solve an equation system. First consider the real part of the inverse DFT of a sequence $a[n]e^{jn\pi/2N}X[n]$ $(n = 0, \cdots, N-1)$:

$$Re\,[e^{j2\pi mn/N}\sum_{n=0}^{N-1} a[n]X[n]e^{jn\pi/2N} = Re\,[\sum_{n=0}^{N-1} a[n]X[n]e^{j(4m+1)n\pi/2N}]$$

$$= \sum_{n=0}^{N-1} a[n]X[n]cos(\frac{(4m+1)n\pi}{2N}) = x[2m], \qquad (m = 0, \cdots, N-1) \tag{6.87}$$

The first half of these m values are the $N/2$ even samples $x[2m]$, $(m = 0, \cdots, N/2-1)$. To obtain the odd samples, recall that $x[m] = x[2N-m-1]$ (Eq. 6.45), and the $N/2$ odd samples are actually the second half of the previous equation in reverse order:

$$x[2m+1] = x[2N-(2m+1)-1] = x[2(N-m-1)], \qquad (m = 0, \cdots, N/2-1) \tag{6.88}$$

In summary, we have these steps for the inverse DCT:

- **step 1:** Generate a sequence $Y[n]$ from the given DCT coefficients $X[n]$:

$$Y[n] = a[n]X[n]e^{jn\pi/2N}, \qquad (n = 0, \cdots, N-1) \tag{6.89}$$

- **step 2:** Obtain $y[m]$ from $Y[n]$ by inverse DFT by FFT. (Only the real part need be computed.)

$$y[m] = Re[\mathcal{F}^{-1}[Y[n]]] \tag{6.90}$$

- **Step 3:** Obtain $x[m]'s$ from $y[m]'s$ by

$$\begin{cases} x[2m] = y[m] \\ x[2m+1] = y[N-1-m] \end{cases} \qquad (i = 0, \cdots, N/2-1) \tag{6.91}$$

These three steps are mathematically equivalent to the steps of the first method. Also note that no scaling factor (either $1/N$ or $1/\sqrt{N}$) is used during the inverse DFT of $Y[n]$. 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 $O(N\,log_2 N)$.

The C code for for the fast DCT algorithm is given below. The DCT function takes a data vector $x[m]$ $(m = 0, \cdots, N-1)$ and converts it to the DCT coefficients $X[k]$. This is an in-place algorithm, i.e., the input data will be overwritten by the output. This function is also used used for the inverse DCT, in which case the input is the DCT coefficients while the output is the reconstructed signal vector in time domain. The function carries out the forward DCT when the argument inv=0, or inverse DCT when inv=1.

```
fdct(x,N,inverse)
     float *x;
     int N,inv;
```

```
{
  int m,n,N2=N/2;
  float a,u,v,w, *yr,*yi;
  w=3.14159265/2/N;
  a=sqrt(2.0/N);
  yr=(float *)malloc(N*sizeof(float)); // allocate memory for two
  yi=(float *)malloc(N*sizeof(float)); // temperary vector variables
  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<N2; m++) {
      yr[m]=x[2*m];
      yr[N-1-m]=x[2*m+1];
      yi[m]=yi[N2+m]=0;
    }
  }
  fft(yr,yi,N,inv);                    // call FFT function
  if (inv) {                           // for IDCT
    for (m=0; m<N2; 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);
}
```

6.2.5 DCT Filtering

As a real-valued transform, the computation of the DCT filtering is more straight forward compared to the DFT filtering. A simple example is illustrated in the example below.

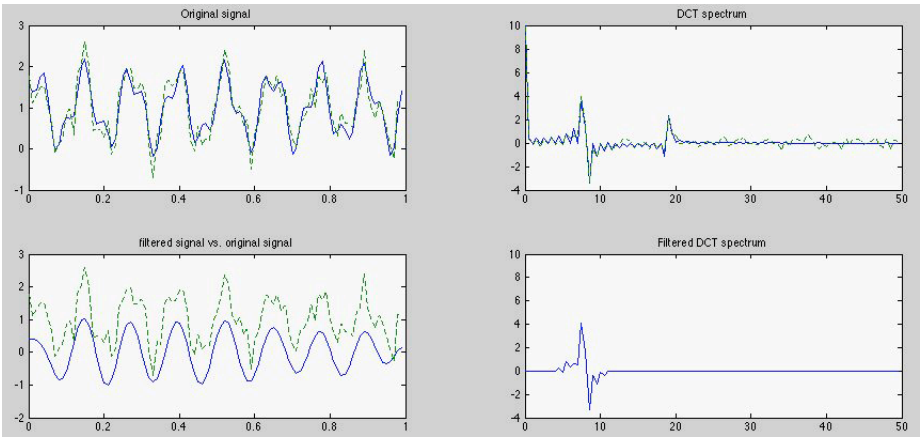

Figure 6.7 DCT Filtering

Example 6.6: The signal shown in the top-left panel of Fig.6.7 is a signal with three frequency components: the DC, we 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 6.7: Here we compare two different types of signals and their DCTs. Shown in Fig.6.8 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

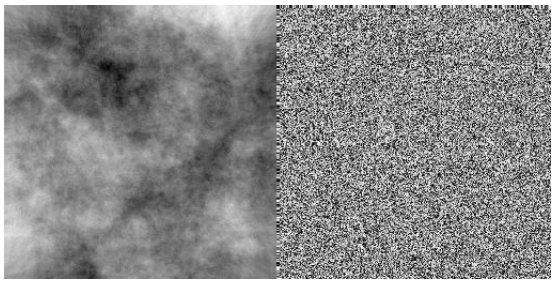

Figure 6.8 Two types of natural scenes: clouds and sand
From left to right and then top to bottom:

defined before in Eq.3.111:

$$r_x(t) = \int_{-\infty}^{\infty} x(\tau)x(\tau - t)d\tau = \int_{-\infty}^{\infty} |X(f)|^2 e^{j2\pi f\tau} df = \mathcal{F}^{-1}[S_x(f)]$$

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.6.9. The four panels on the left are for the clouds showing the signal (1st) and its DCT (3rd), together with their auto-correlation (2nd and 4th). Note that the original signal is highly correlated, and the closer two samples of the signal the more they are correlated. But after the DCT, the frequency components are not correlated at all. (Note that the auto-correlations look symmetric due to the periodicity assumed by the DCT.) In the same manner, the four panels on the right show the signal of the sand and its DCT together with their auto-correlations. In this case, the signal is hardly correlated, and the frequency components in its DCT spectrum are even less so.

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 DCT, or any other orthogonal transform for this matter, are much less correlated. This example illustrates that signal decorrelation is an important feature of all orthogonal transforms, by which the autocorrelation of a typical signal will be significantly reduced.

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).

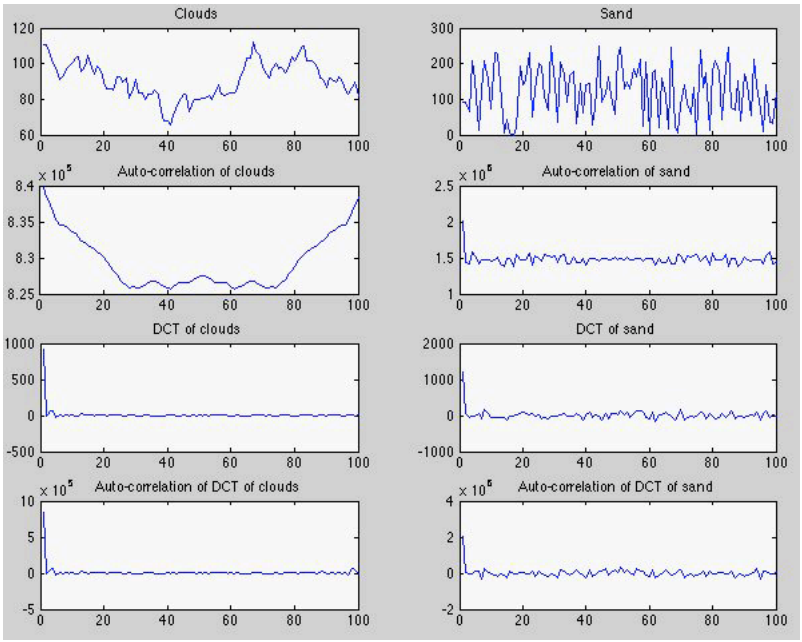

Figure 6.9 Decorrelation of cloud and sand signals

6.2.6 Two-Dimensional DCT and Filtering

The DCT of a 2-D signal $x[m,n]$ $(m = 0, \cdots, M-1,\ n = 0, \cdots, N-1)$ such as an image, and the inverse DCT are defined respectively as:

$$X[k,l] = a[k]a[l] \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} x[m,n]\ cos(\frac{(2m+1)k\pi}{2M})\ cos(\frac{(2n+1)l\pi}{2N}),$$

$$x[m,n] = \sum_{l=0}^{N-1} a[l] \sum_{k=0}^{M-1} a[k]X[k,l] \cos(\frac{(2m+1)k\pi}{2M}) \cos(\frac{(2m+1)l\pi}{2N}),$$

$$(m,k = 0, \cdots, M-1, n,l = 0, \cdots, N-1) \tag{6.92}$$

The inverse DCT (second equation) expresses the given signal as a linear combination of a set of M by N 2-D basis functions, a product of two sinusoidal functions in horizontal and vertical directions, respectively. Each of these basis function is weighted by the 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. These 2-D basis functions can be visualized as shown in Fig.6.10 for $M = N = 8$.

Similar to the 2-D DFT, the two summations in either the forward or inverse DCT in Eq. 6.92 can be carried separately in two separate steps, First, we can carry out N M-point 1-D DCTs for each of the N columns of the 2-D signal array (the inner summation with respect to m in Eq.6.92), and then carry out M N-point 1-D DCTs for each of the M rows of the resulting array after the first step

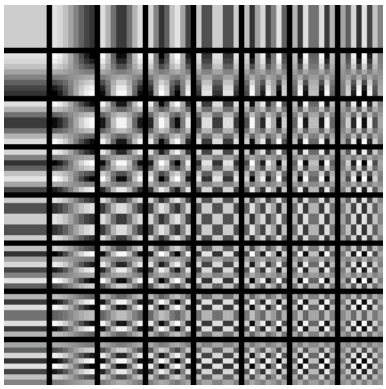

Figure 6.10 The basis functions for the 2-D DCT ($M = N = 8$)
The DC component is at the top-left corner, and the highest frequency component in both horizontal and vertical directions is at the lower-right corner.

(the outer summation with respect to n in Eq.6.92). 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

$$\begin{cases} \boldsymbol{X}_{M\times N} = \boldsymbol{C}_M^T \boldsymbol{x}_{M\times N} \boldsymbol{C}_N & \text{(forward)} \\ \boldsymbol{x}_{M\times N} = \boldsymbol{C}_M \boldsymbol{X}_{M\times N} \boldsymbol{C}_N^T & \text{(inverse)} \end{cases} \tag{6.93}$$

Here $\boldsymbol{C}_M = [\boldsymbol{c}_0, \cdots, \boldsymbol{c}_{M-1}]$ is an M by M matrix for the column transform and $\boldsymbol{C}_N = [\boldsymbol{c}_0, \cdots, \boldsymbol{c}_{N-1}]$ is an N by N matrix for the row transform. The DCT spectrum of a 2-D signal, e.g., an image, is a real matrix composed of the M by N coefficients $X[k, l]$ representing the magnitudes of the corresponding basis functions. Different from the Fourier transform, the phases of the basis functions are not of interest in the DCT.

The DCT matrix $\boldsymbol{C}$ can be expressed in terms of its column vectors and the inverse transform can be written as:

$$\begin{aligned} \boldsymbol{x} &= [\boldsymbol{c}_0, \cdots, \boldsymbol{c}_{M-1}] \begin{bmatrix} X[0,0] & \cdots & X[0,N-1] \\ \vdots & \ddots & \vdots \\ X[M-1,0] & \cdots & X[M-1,N-1] \end{bmatrix} \begin{bmatrix} \boldsymbol{c}_0^T \\ \vdots \\ \boldsymbol{c}_{N-1}^T \end{bmatrix} \\ &= \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} X[k,l] \boldsymbol{c}_k \boldsymbol{c}_l^T = \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} X[k,l] \boldsymbol{B}_{kl} \end{aligned} \tag{6.94}$$

Here we have defined $\boldsymbol{B}_{kl} = \boldsymbol{c}_k \boldsymbol{c}_l^T$, where $\boldsymbol{c}_k$ is the kth column vector of the M by M DCT matrix for the row transforms and $\boldsymbol{c}_l$ is the lth column vector of the N by N DCT matrix for the column transforms. We see that the 2-D signal $\boldsymbol{x}_{M\times N}$ is now expressed as a linear combination of a set of MN 2-D ($M \times N$) DCT basis functions $\boldsymbol{B}_{kl}$ ($k, l = 0. \cdots, N-1$), which can be obtained from the equation above for the inverse transform when all elements of $\boldsymbol{X}$ are zero except

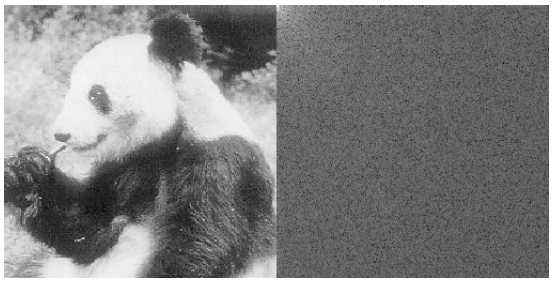

Figure 6.11 An image and its DCT spectrum

$X[k,l] = 1$. When $M = N = 8$, the $8 \times 8 = 64$ such 2-D DCT basis functions are shown in Fig.6.10. Any 8 by 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 $\boldsymbol{B}_{kl}$ is weighted by the kl-th DCT coefficients $X[k,l]$, which can be obtained by the forward transform:

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{C}_0^T \\ \vdots \\ \boldsymbol{c}_{M-1}^T \end{bmatrix} \boldsymbol{x}[\boldsymbol{c}_0, \cdots, \boldsymbol{c}_{N-1}] \tag{6.95}$$

and the kl-th coefficient $X[k,l]$ is the projection of the 2-D signal $\boldsymbol{x}$ onto the kl-th basis function, i.e., their inner product:

$$\begin{aligned} X[k,l] &= \boldsymbol{c}_k^T \begin{bmatrix} x[0,0] & \cdots & x[0,N-1] \\ \vdots & \ddots & \vdots \\ x[M-1,0] & \cdots & x[M-1,N-1] \end{bmatrix} \boldsymbol{c}_l \\ &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m,n] B_{kl}[m,n] = < \boldsymbol{x}, \boldsymbol{B}_{kl} > \end{aligned} \tag{6.96}$$

Same as in the 2-D DFT case (Eq.4.214), the coefficient $X[k,l]$ can be found as the projection of the signal $\boldsymbol{x}$ onto the kl-th DCT basis function $\boldsymbol{B}_{kl}$.

Example 6.8: An image and its DCT spectrum are shown in Fig. 6.11. Different from the complex DFT, the DCT is a real transform and a 2-D DCT spectrum is a array of real elements representing the magnitudes of the frequency components, unlike a 2-D DFT spectrum which contains both the real and imaginary parts, representing the magnitudes and phases for the frequency components.

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.6.12 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 higher than a cut-off frequency, i.e., farther away from the DC component (top-left corner of the spectrum) than a distance correspond-

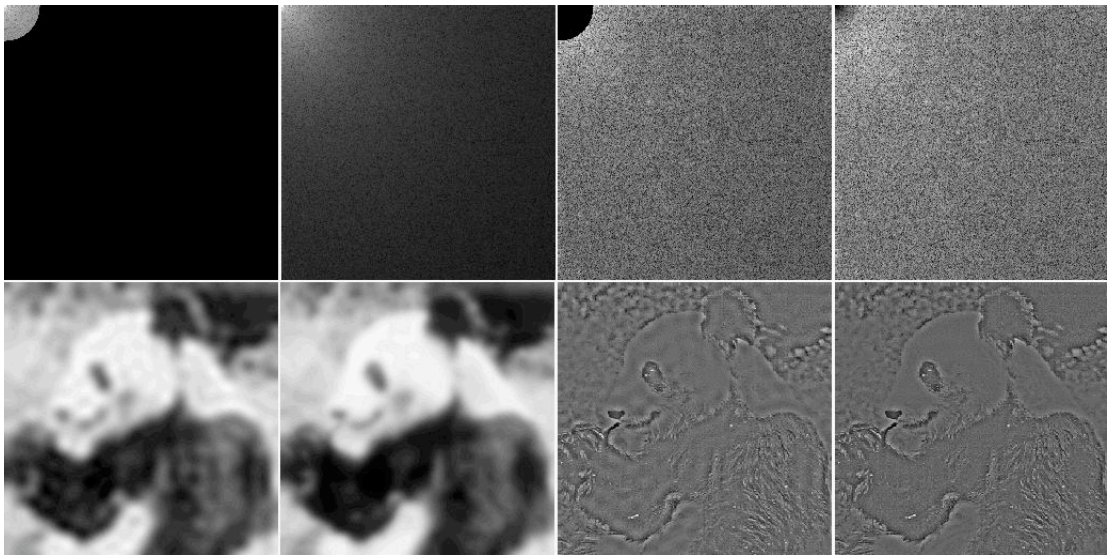

Figure 6.12 LP and HP filtering of an image
Similar to the Fourier transform, DCT also suffers from the ringing artifacts caused by the ideal filters (first and third), which can be avoided by the smooth Butterworth filter.

ing to the cut-off frequency, are suppressed to zero with all other components unchanged. The modified spectrum and the resulting low-pass filtered image after 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 6.9: The example shown in Fig. 6.13 illustrates why the DCT can also be used for data compression. In this particular 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.

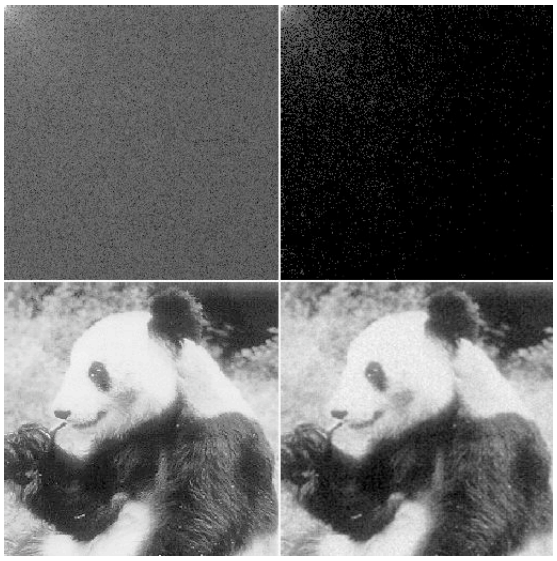

Figure 6.13 Image Compression based on DCT
An image and its DCT spectrum (left) and the reconstructed image (right) based on 10% of the coefficients but 99.6% of the total energy.

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 (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. For this reason, DCT is widely used in image compression, most noticeably in the image compression standards, such as JPEG (http://en.wikipedia.org/wiki/JPEG).

7 The Walsh-Hadamard, Slant and Haar Transforms

7.1 The Walsh-Hadamard Transform

The Walsh-Hadamard Transform (WHT) is yet another real orthogonal transform which can also be closely related to the discrete cosine transform (DCT), although they are defined totally differently.

7.1.1 Hadamard Matrix

Let us first consider an operation between two matrices. The *Kronecker product* of two matrices $\boldsymbol{A} = [a_{ij}]_{m\times n}$ and $\boldsymbol{B} = [b_{ij}]_{k\times l}$ is defined as

$$\boldsymbol{A}\otimes\boldsymbol{B} = \begin{bmatrix} a_{11}\boldsymbol{B} & \cdots & a_{1n}\boldsymbol{B} \\ \cdots & \cdots & \cdots \\ a_{m1}\boldsymbol{B} & \cdots & a_{mn}\boldsymbol{B} \end{bmatrix}_{mk\times nl} \tag{7.1}$$

In general, $\boldsymbol{A}\otimes\boldsymbol{B} \neq \boldsymbol{B}\otimes\boldsymbol{A}$. Now the *Hadamard Matrix* is defined recursively as:

$$\boldsymbol{H}_1 = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{7.2}$$

$$\boldsymbol{H}_n = \boldsymbol{H}_1\otimes\boldsymbol{H}_{n-1} = \frac{1}{\sqrt{2}}\begin{bmatrix} \boldsymbol{H}_{n-1} & \boldsymbol{H}_{n-1} \\ \boldsymbol{H}_{n-1} & -\boldsymbol{H}_{n-1} \end{bmatrix} \tag{7.3}$$

Note that the dimensionality of matrix $\boldsymbol{H}_n$ is 2^n by 2^n. For example, when $n = 2$, we have

$$\boldsymbol{H}_2 = \boldsymbol{H}_1\otimes\boldsymbol{H}_1 = \frac{1}{\sqrt{2}}\begin{bmatrix} \boldsymbol{H}_1 & \boldsymbol{H}_1 \\ \boldsymbol{H}_1 & -\boldsymbol{H}_1 \end{bmatrix} = \frac{1}{\sqrt{4}}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \tag{7.4}$$

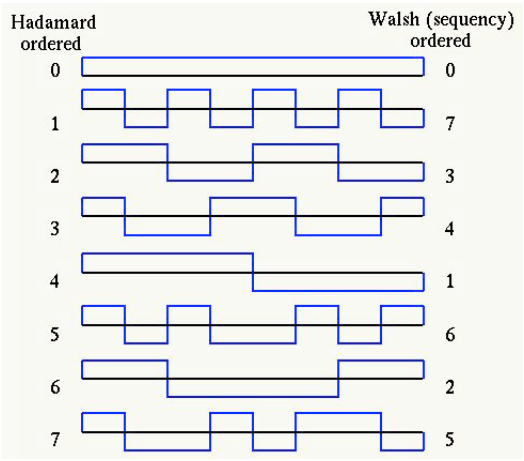

Figure 7.1 The basis functions for the WHT

and if $n = 3$, we have

$$\boldsymbol{H}_3 = \boldsymbol{H}_1 \otimes \boldsymbol{H}_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} \boldsymbol{H}_2 & \boldsymbol{H}_2 \\ \boldsymbol{H}_2 & -\boldsymbol{H}_2 \end{bmatrix} = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix} \begin{matrix} 0\ 0 \\ 1\ 7 \\ 2\ 3 \\ 3\ 4 \\ 4\ 1 \\ 5\ 6 \\ 6\ 2 \\ 7\ 5 \end{matrix} \tag{7.5}$$

The first column to the right of the array is the index numbers k of the $N = 8$ rows, and the second column represents the *sequency* s, the number of zero-crossings or sign changes in each row. As can be seen, the index numbers corresponding to sequencies $s = 0, 1, 2, 3, 4, 5, 6, 7$ are $k = 0, 4, 6, 2, 3, 7, 5, 1$, respectively.

Similar to frequency, sequency also measures the rate of changes in a signal. But, different from frequency, sequency can be used to measure non-periodical signals as well as periodic ones.

Alternatively, a Hadamard matrix $\boldsymbol{H}$ can also be defined in terms of its element $h[k, m]$ in the kth row and mth column as below (for simplicity, the scaling factor $1/\sqrt{N}$ is neglected for now):

$$h[k, m] = (-1)^{\sum_{i=0}^{n-1} k_i m_i} = \prod_{i=0}^{n-1} (-1)^{k_i m_i} = h[m, k] \qquad (k, m = 0, 1, \cdots, N-1) \tag{7.6}$$

where

$$k = \sum_{i=0}^{n-1} k_i 2^i = (k_{n-1}k_{n-2}\cdots k_1 k_0)_2 \quad (k_i = 0, 1) \tag{7.7}$$

$$m = \sum_{i=0}^{n-1} m_i 2^i = (m_{n-1}m_{n-2}\cdots m_1 m_0)_2 \quad (m_i = 0, 1) \tag{7.8}$$

i.e., $(k_{n-1}k_{n-2}\cdots k_1k_0)_2$ and $(m_{n-1}m_{n-2}\cdots m_1m_0)_2$ 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 $h[k, l]$ in row $k = 2 = (010)_2$ and column $m = 3 = (011)_2$ of $\boldsymbol{H}_3$ is $(-1)^{0+1+0} = -1$.

It is easy to show that this alternative definition of the Hadamard matrix is actually the same as the previous recursive definition given in Eqs. 7.2 and 7.3. 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_0m_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., $h[0, 0] = h[0, 1] = h[1, 0] = (-1)^{k_0m_0} = (-1)^0 = 1$ and $h[1, 1] = (-1)^{k_0m_0} = (-1)^1 = -1$. This is actually Eq. 7.2.

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 $\boldsymbol{H}_n$. The first three quadrants (upper-left, upper-right and lower-left) corresponding to $k_{n-1}m_{n-1} = 0$ are therefore identical to $\boldsymbol{H}_{n-1}$, while the lower-right quadrant corresponding to $k_{n-1}m_{n-1} = 1$ is the negation of $\boldsymbol{H}_{n-1}$. This is the recursion in Eq. 7.3.

Obviously $\boldsymbol{H}$ is real and symmetric, and we can easily show that it is also orthogonal:

$$\boldsymbol{H} = \boldsymbol{H}^* = \boldsymbol{H}^T = \boldsymbol{H}^{-1} \tag{7.9}$$

To do so, we first note that $\boldsymbol{H}_1\boldsymbol{H}_1 = \boldsymbol{I}$. Next we assume $\boldsymbol{H}_{n-1}\boldsymbol{H}_{n-1} = \boldsymbol{I}_{n-1}$, and consider

$$\begin{aligned}\boldsymbol{H}_n\boldsymbol{H}_n &= \frac{1}{\sqrt{2}}\begin{bmatrix}\boldsymbol{H}_{n-1} & \boldsymbol{H}_{n-1}\\ \boldsymbol{H}_{n-1} & -\boldsymbol{H}_{n-1}\end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix}\boldsymbol{H}_{n-1} & \boldsymbol{H}_{n-1}\\ \boldsymbol{H}_{n-1} & -\boldsymbol{H}_{n-1}\end{bmatrix}\\ &= \frac{1}{2}\begin{bmatrix}2\boldsymbol{H}_{n-1}\boldsymbol{H}_{n-1} & \boldsymbol{0}\\ \boldsymbol{0} & 2\boldsymbol{H}_{n-1}\boldsymbol{H}_{n-1}\end{bmatrix} = \begin{bmatrix}\boldsymbol{I}_{n-1} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{I}_{n-1}\end{bmatrix} = \boldsymbol{I}_n\end{aligned} \tag{7.10}$$

therefore the matrix is orthogonal for all n:

$$\boldsymbol{H} = \boldsymbol{H}^{-1} \tag{7.11}$$

7.1.2 Hadamard Ordered Walsh-Hadamard Transform (WHT_h)

The orthogonal Hadamard matrix can be written in terms of its columns:

$$\boldsymbol{H} = [\boldsymbol{h}_0, \cdots, \boldsymbol{h}_{N-1}] \tag{7.12}$$

As these N vectors are orthonormal

$$< \boldsymbol{h}_m, \boldsymbol{h}_n >= \boldsymbol{h}_m^T \boldsymbol{h}_n = \delta[m-n] \tag{7.13}$$

they form a complete basis that spans the N-dimensional vector space, and the Hadamard matrix $\boldsymbol{H}$ can be used to define an orthogonal transform, called Hadamard ordered Walsh-Hadamard transform (WHT_h):

$$\begin{cases} \boldsymbol{X} = \boldsymbol{H}\boldsymbol{x} & \text{(forward)} \\ \boldsymbol{x} = \boldsymbol{H}\boldsymbol{X} & \text{(inverse)} \end{cases} \tag{7.14}$$

Here $\boldsymbol{x} = [x[0], \cdots, x[N-1]]^T$ is an N-point signal vector and $\boldsymbol{X} = X[0], \cdots, X[N-1]]^T$ is its WHT spectrum vectors. Note that, interestingly, as $\boldsymbol{H}^{-1} = \boldsymbol{H}$, 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 written as:

$$\boldsymbol{x} = [\boldsymbol{h}_0, \cdots, \boldsymbol{h}_{N-1}] \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \sum_{k=0}^{N-1} X[k]\boldsymbol{h}_k \tag{7.15}$$

i.e., the signal vector is expressed as a linear combination of the N basis vectors $\boldsymbol{h}_k$ weighted by the WHT coefficients $X[k]$ $(k = 0, \cdots, N-1)$, which can be written by the forward WHT:

$$\boldsymbol{X} = \boldsymbol{H}\boldsymbol{x} = [\boldsymbol{h}_0, \cdots, \boldsymbol{h}_{N-1}]\boldsymbol{x} \tag{7.16}$$

or in component form:

$$X[k] =< \boldsymbol{x}, \boldsymbol{h}_k >= \boldsymbol{h}_k^T \boldsymbol{x}, \qquad (k = 0, \cdots, N-1) \tag{7.17}$$

i.e., the coefficient $X[k]$ is the projection of the signal vector $\boldsymbol{x}$ onto the kth basis vector $\boldsymbol{h}_k$, which can also be written as

$$X[k] = \sum_{m=0}^{N-1} h[k,m]x[m] = \sum_{m=0}^{N-1} x[m] \prod_{i=0}^{n-1} (-1)^{m_i k_i} \tag{7.18}$$

7.1.3 Fast Walsh-Hadamard Transform Algorithm

The complexity of WHT implemented as a matrix multiplication $\boldsymbol{X} = \boldsymbol{H}\boldsymbol{x}$ is $O(N^2)$. However, similar to the FFT algorithm, we can also derive a fast WHT algorithm with complexity of $O(Nlog_2N)$ as shown below. We assume $n = 3$ and

$N = 2^n = 8$, and write the WHT_h of an 8-point signal $\boldsymbol{x}$ as:

$$\boldsymbol{X} = \boldsymbol{H}_3\boldsymbol{x} = \begin{bmatrix} X[0] \\ \vdots \\ X[3] \\ X[4] \\ \vdots \\ X[7] \end{bmatrix} = \begin{bmatrix} \boldsymbol{H}_2 & \boldsymbol{H}_2 \\ \boldsymbol{H}_2 & -\boldsymbol{H}_2 \end{bmatrix} \begin{bmatrix} x[0] \\ \vdots \\ x[3] \\ x[4] \\ \vdots \\ x[7] \end{bmatrix} \tag{7.19}$$

This equation can be separated into two parts. The first half of vector $\boldsymbol{X}$ can be obtained as

$$\begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ X[3] \end{bmatrix} = \boldsymbol{H}_2 \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix} + \boldsymbol{H}_2 \begin{bmatrix} x[4] \\ x[5] \\ x[6] \\ x[7] \end{bmatrix} = \boldsymbol{H}_2 \begin{bmatrix} x_1[0] \\ x_1[1] \\ x_1[2] \\ x_1[3] \end{bmatrix} \tag{7.20}$$

where we have defined

$$x_1[i] = x[i] + x[i+4] \quad (i = 0, \cdots, 3) \tag{7.21}$$

Similarly the second half of vector $\boldsymbol{X}$ can be obtained as

$$\begin{bmatrix} X[4] \\ X[5] \\ X[6] \\ X[7] \end{bmatrix} = \boldsymbol{H}_2 \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \end{bmatrix} - \boldsymbol{H}_2 \begin{bmatrix} x[4] \\ x[5] \\ x[6] \\ x[7] \end{bmatrix} = \boldsymbol{H}_2 \begin{bmatrix} x_1[4] \\ x_1[5] \\ x_1[6] \\ x_1[7] \end{bmatrix} \tag{7.22}$$

where we have defined

$$x_1[i+4] = x[i] - x[i+4] \quad (i = 0, \cdots, 3) \tag{7.23}$$

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. 7.20 as:

$$\begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ X[3] \end{bmatrix} = \begin{bmatrix} \boldsymbol{H}_1 & \boldsymbol{H}_1 \\ \boldsymbol{H}_1 & -\boldsymbol{H}_1 \end{bmatrix} \begin{bmatrix} x_1[0] \\ x_1[1] \\ x_1[2] \\ x_1[3] \end{bmatrix} \tag{7.24}$$

which can again be separated into two halves. The first half is

$$\begin{bmatrix} X[0] \\ X[1] \end{bmatrix} = \boldsymbol{H}_1 \begin{bmatrix} x_1[0] \\ x_1[1] \end{bmatrix} + \boldsymbol{H}_1 \begin{bmatrix} x_1[2] \\ x_1[3] \end{bmatrix} = \boldsymbol{H}_1 \begin{bmatrix} x_2[0] \\ x_2[1] \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_2[0] \\ x_2[1] \end{bmatrix} = \begin{bmatrix} x_2[0] + x_2[1] \\ x_2[0] - x_2[1] \end{bmatrix} \tag{7.25}$$

where

$$x_2[i] = x_1[i] + x_1[i+2] \quad (i = 0, 1) \tag{7.26}$$

and

$$X[0] = x_2[0] + x_2[1], \qquad X[1] = x_2[0] - x_2[1] \tag{7.27}$$

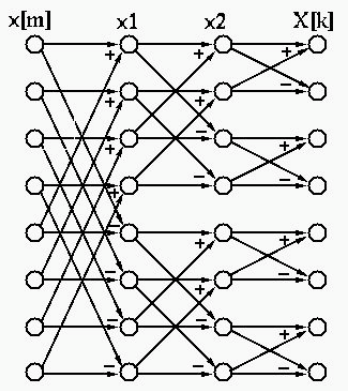

Figure 7.2 The fast WHT algorithm

The second half is

$$\begin{bmatrix} X[2] \\ X[3] \end{bmatrix} = \boldsymbol{H}_1 \begin{bmatrix} x_1[0] \\ x_1[1] \end{bmatrix} - \boldsymbol{H}_1 \begin{bmatrix} x_1[2] \\ x_1[3] \end{bmatrix} = \boldsymbol{H}_1 \begin{bmatrix} x_2[2] \\ x_2[3] \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x_2[2] \\ x_2[3] \end{bmatrix} = \begin{bmatrix} x_2[2] + x_2[3] \\ x_2[2] - x_2[3] \end{bmatrix} \tag{7.28}$$

where

$$x_2[i+2] = x_1[i] - x_1[i+2] \quad (i = 0, 1) \tag{7.29}$$

and

$$X[2] = x_2[2] + x_2[3], \qquad X[3] = x_2[2] - x_2[3] \tag{7.30}$$

Similarly the coefficients $X[4]$ through $X[7]$ in the second half of the transform in Eq. 7.22 can be obtained by the same process. Summarizing the above steps of Equations 7.21, 7.23, 7.26, 7.27, 7.29, 7.30, we get the fast WHT algorithm as illustrated in Fig. 7.2.

7.1.4 Sequency Ordered Walsh-Hadamard Matrix (WHT_w)

The rows in the WHT matrix $\boldsymbol{x}$ are not arranged in order of their sequencies, while it makes better physical sense if the elements of the WHT spectrum $\boldsymbol{X} = [X[0], X[1], \cdots, X[N-1]]^T$ are arranged according to their sequencies so that they represent different components contained in the signal in a low-to-high order, such as in the Fourier transform. To do so, we can re-order the rows (or columns) of the Hadamard matrix H according to their sequencies. We first consider the conversion of a given sequency number s into the corresponding row index number k in Hadamard order, which can be down in the following three steps:

1. represent s in binary form:

$$s = (s_{n-1} \cdots s_0)_2 = \sum_{i=0}^{n-1} s_i 2^i \tag{7.31}$$

2. convert this n-bit binary number to an n-bit Gray code:

$$g = (g_{n-1} \cdots g_0)_2, \quad \text{where} \quad g_i = s_i \oplus s_{i+1} \ \ (i = 0, \cdots, n-1) \tag{7.32}$$

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:

$$k_i = g_{n-1-i} = s_{n-1-i} \oplus s_{n-i} \tag{7.33}$$

Now row index k can be obtained:

$$k = (k_{n-1}\, k_{n-2} \cdots k_1\, k_0)_2 = \sum_{i=0}^{n-1} s_{n-1-i} \oplus s_{n-i} 2^i = \sum_{j=0}^{n-1} s_j \oplus s_{j+1} 2^{n-1-j} \tag{7.34}$$

where $j = n - 1 - i$ or equivalently $i = n - 1 - j$.

For example, when $n = log_2 N = log_2 8 = 3$, 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

(7.35)

Now the sequency-ordered (also called Walsh-ordered) Walsh-Hadamard matrix can be obtained as

$$\boldsymbol{H}_w = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix} \begin{matrix} 0\ 0 \\ 1\ 4 \\ 2\ 6 \\ 3\ 2 \\ 4\ 3 \\ 5\ 7 \\ 6\ 5 \\ 7\ 1 \end{matrix} \tag{7.36}$$

Here a subscript w is included to indicate the row vectors of this matrix $\boldsymbol{H}$ 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: $\boldsymbol{H}_h^T = \boldsymbol{H}_w$.

Now the sequency-ordered Walsh-Hadamard transform (WHT_w) can be carried out as

$$\boldsymbol{X} = \boldsymbol{H}_w \boldsymbol{x} \tag{7.37}$$

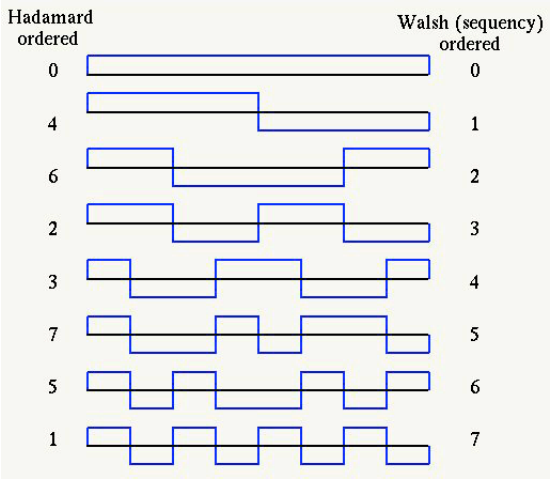

Figure 7.3 The basis functions for the WHT (sequency ordered)

or in component form:

$$X[k] = \sum_{m=0}^{N-1} h_w[k,m]x[m] \tag{7.38}$$

where $h_w[k,m]$ is the element in the kth row and nth column of $\boldsymbol{H}_w$.

7.1.5 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 $\boldsymbol{X}$ as shown above. Alternatively, we can use the following fast WHT_w directly with better efficiency.

Similar to the WHT shown in Eq.7.18, the sequency ordered WHT of $x[m]$ can be represented as:

$$\begin{aligned} X[k] = \sum_{m=0}^{N-1} h_w[k,m]x[m] &= \sum_{m=0}^{N-1} x[m] \prod_{j=0}^{n-1} (-1)^{(k_{n-1-j}+k_{n-j})m_j} \\ &= \sum_{m=0}^{N-1} x[m] \prod_{i=0}^{n-1} (-1)^{(k_i+k_{i+1})m_{n-1-i}} \end{aligned} \tag{7.39}$$

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.7.34). 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 we 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:

$$m = \sum_{i=0}^{n-1} m_i 2^i = 4m_2 + 2m_1 + m_0, \quad k = \sum_{i=0}^{n-1} k_i 2^i = 4k_2 + 2k_1 + k_0 \tag{7.40}$$

Here $k_n = k_3 = 0$ is defined to be zero. 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:

$$x_0[4m_0 + 2m_1 + m_2] = x[4m_2 + 2m_1 + m_0] \qquad \text{for } m = 0, 1, \cdots 7 \tag{7.41}$$

  Now Eq.7.39 can be written as:

$$\begin{aligned} X[k] &= \sum_{m_2=0}^{1} \sum_{m_1=0}^{1} \sum_{m_0=0}^{1} x_0[4m_0 + 2m_1 + m_2] \prod_{i=0}^{2} (-1)^{(k_i + k_{i+1}) m_{n-1-i}} \\ &= \sum_{l_0=0}^{1} \sum_{l_1=0}^{1} \sum_{l_2=0}^{1} x_0[4l_2 + 2l_1 + l_0] \prod_{i=0}^{2} (-1)^{(k_i + k_{i+1}) l_i} \end{aligned} \tag{7.42}$$

  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

$$\begin{aligned} X[k] &= \sum_{l_0=0}^{1} \sum_{l_1=0}^{1} \prod_{i=0}^{1} (-1)^{(k_i + k_{i+1}) l_i} [x_0[2l_1 + l_0] + (-1)^{k_2 + k_3} x_0[4 + 2l_1 + l_0]] \\ &= \sum_{l_0=0}^{1} \sum_{l_1=0}^{1} \prod_{i=0}^{1} (-1)^{(k_i + k_{i+1}) l_i} x_1[4k_2 + 2l_1 + l_0] \end{aligned} \tag{7.43}$$

  where x_1 is defined as

$$x_1[4k_2 + 2l_1 + l_0] = x_0[2l_1 + l_0] + (-1)^{k_2 + k_3} x_0[4 + 2l_1 + l_0] \tag{7.44}$$

- Again, expanding the 2nd summation into two terms for $l_1 = 0$ and $l_1 = 1$, we get

$$\begin{aligned} X[k] &= \sum_{l_0=0}^{1} (-1)^{(k_i + k_{i+1}) l_0} [x_1[4k_2 + l_0] + (-1)^{k_1 + k_2} x_1[4k_2 + 2 + l_0]] \\ &= \sum_{l_0=0}^{1} (-1)^{(k_i + k_{i+1}) l_0} x_2[4k_2 + 2k_1 + m_0] \end{aligned} \tag{7.45}$$

  where x_2 is defined as

$$x_2[4k_2 + 2k_1 + l_0] = x_1[4k_2 + l_0] + (-1)^{k_1 + k_2} x_1[4k_2 + 2 + l_0] \tag{7.46}$$

- Finally, expanding the 1st summation into two terms for $l_0 = 0$ and $l_0 = 1$, we have

$$X[k] = x_2[4k_2 + 2k_1] + (-1)^{k_0 + k_1} x_2[4k_2 + 2k_1 + 1] \tag{7.47}$$

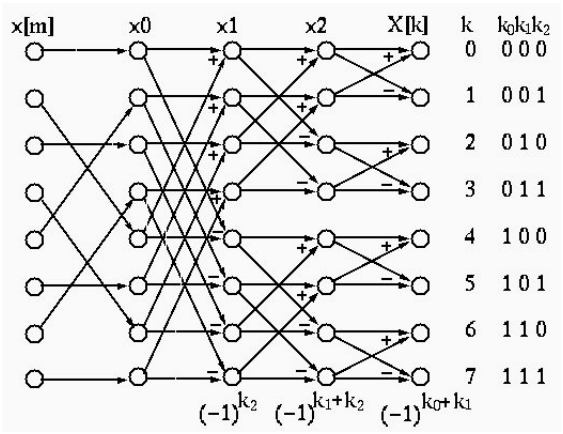

Figure 7.4 The fast WHT algorithm (sequency ordered)

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.7.4. 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 $x[m]$ $(m = 0, \cdots, N-1)$ and converts it to WHT coefficients $X[k]$ $(k = 0, \cdots, N-1)$, which are Hadamard ordered if the argument sequency 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;
```

```
      }
      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 Haramard 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 7.1: The sequency ordered WHT of an 8-point signal vector $\boldsymbol{x} = [0, 0, 2, 3, 4, 0, 0, 0]^T$ can be obtained by matrix multiplication:

$$\boldsymbol{X} = \boldsymbol{H}_w \boldsymbol{x} = [3.18, 0.35, -3.18, -0.35, 1.77, -1.06, -1.77, 1.06]^T \tag{7.48}$$

where $\boldsymbol{H}_w$ is given in Eq.7.36. The inverse transform (which is identical to the forward transform as $\boldsymbol{H}_w^{-1} = \boldsymbol{H}_w$) represents the signal vector as a linear combination of a set of square waves of different sequencies:

$$\boldsymbol{x} = \boldsymbol{H}_w \boldsymbol{X} = [h_0, \cdots, h_7]\boldsymbol{X} = \sum_{n=0}^{7} X[n]\boldsymbol{h}_n[0, 0, 2, 3, 4, 0, 0, 0]^T \tag{7.49}$$

This example is illustrated in Fig.7.5.

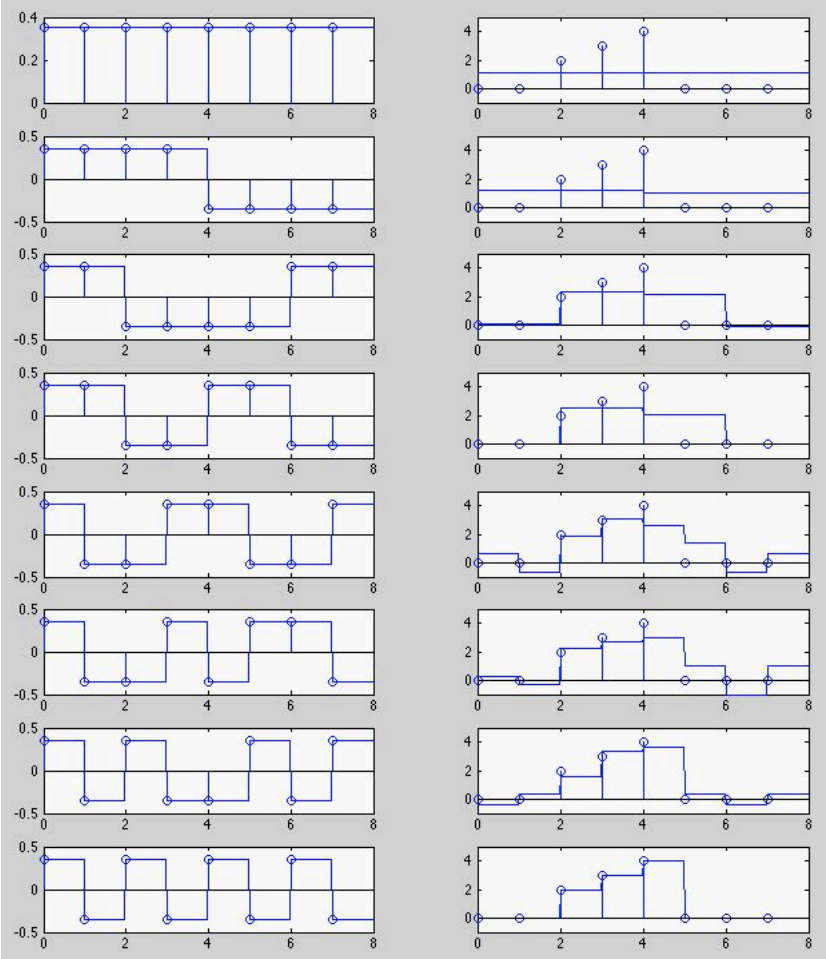

Figure 7.5 The WHT of a 8-point signal
The left column shows the 8 basis WHT functions (both continuous and discrete), while the right column shows how a signal can be reconstructed by the inverse WHT (Eq.7.49) as a linear combination of these basis functions weighted by WHT coefficients obtained by the forward WHT (Eq.7.48). The plots on the right show the reconstructed signal using progressively more components of higher sequencies (from DC component alone to all 8 sequency components).

7.2 The Slant Transform

7.2.1 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 defined the same as $\boldsymbol{H}_1$ for the Hadamard matrix (Eq.7.2):

$$\boldsymbol{S}_1 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{7.50}$$

The recursive definition for matrix $\boldsymbol{S}_n$ of size $N = 2^n$ is:

$$\boldsymbol{S}_n = \boldsymbol{R}_n \left[\boldsymbol{S}_1 \otimes \boldsymbol{S}_{n-1} \right] = \frac{1}{\sqrt{2}} \boldsymbol{R}_n \begin{bmatrix} \boldsymbol{S}_{n-1} & \boldsymbol{S}_{n-1} \\ \boldsymbol{S}_{n-1} & -\boldsymbol{S}_{n-1} \end{bmatrix} \tag{7.51}$$

where $\boldsymbol{R}_n$ is a size $N = 2^n$ rotation matrix by which the $N/4$-th row and $N/2$-th row are rotated by an angle θ_n:

$$\boldsymbol{R}_n = \begin{bmatrix} 1 & & & & & & & & \\ & \ddots & & & & & & & \\ & & 1 & & & & & & \\ & & & \cos\theta_n & & & -\sin\theta_n & & \\ & & & & 1 & & & & \\ & & & & & \ddots & & & \\ & & & & & & 1 & & \\ & & & \sin\theta_n & & & \cos\theta_n & & \\ & & & & & & & 1 & \\ & & & & & & & & \ddots \\ & & & & & & & & & 1 \end{bmatrix} \begin{matrix} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{matrix} \begin{matrix} \\ \\ \\ \\ \\ \\ \\ (2^{n-2} = N/4)\text{th row} \\ \\ \\ \\ (2^{n-1} = N/2)\text{th row} \end{matrix} \tag{7.52}$$

where

$$\begin{aligned} \cos\theta_n &= \left(\frac{2^{2n-2} - 1}{2^{2n} - 1} \right)^{1/2} = \sqrt{\frac{3N^2}{4N^2 - 4}} \\ \sin\theta_n &= \left(\frac{2^{2n-2} - 2^{2n-2}}{2^{2n} - 1} \right)^{1/2} = \sqrt{\frac{N^2 - 4}{4N^2 - 4}} \end{aligned} \tag{7.53}$$

Note that indeed we have $\sin_n^2 + \cos_n^2 = 1$. For example, when $n = 1,\ 2, 3$ we have:

n	N	$\cos\theta_n$	$\sin\theta_n$
1	2	0	1
2	4	$1/\sqrt{5}$	$2/\sqrt{5}$
3	8	$\sqrt{5/21}$	$4/\sqrt{21}$

The N row vectors of the identity matrix $\boldsymbol{I}_n$ of size $N = 2^n$ can be considered as a set of N standard basis vectors $\boldsymbol{e}_k$ $(k = 0, \cdots, N-1)$ of an N-D vector space. Similarly, the N vectors of matrix $\boldsymbol{R}_n$ also form a set of basis vectors, of which $N - 2$ are the same standard vectors as those in $\boldsymbol{I}_n$, except rows number $N/4$ and number $N/2$ which are rotated by an angle θ_n. As rotation is a unitary transformation, the rotation matrix is unitary, i.e.,

$$\boldsymbol{R}_n^T \boldsymbol{R}_n = \boldsymbol{I}_n$$

Without the rotation, i.e., if $\theta_n = 0$ and $\boldsymbol{R}_n = \boldsymbol{I}_n$, Eq.7.51 for slant matrix becomes the same as Eq.7.3 for Hadamard matrix.

The slant transform matrix $\boldsymbol{S}_n$ is obviously real but not symmetric, and we can also show that it is orthogonal:

$$\boldsymbol{S}_n^T = \boldsymbol{S}_n^{-1}, \quad \text{i.e.,} \quad \boldsymbol{S}_n^T \boldsymbol{S}_n = \boldsymbol{I}_n \tag{7.54}$$

Proof:

This is a proof by induction. First, when $n = 1$, it is trivial to show that $\boldsymbol{R}_1$ is orthogonal:

$$\boldsymbol{S}_1^T \boldsymbol{S}_1 = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \boldsymbol{I}_1$$

Next, we assume $\boldsymbol{S}_{n-1}^T \boldsymbol{S}_{n-1} = \boldsymbol{I}_{n-1}$, and consider:

$$\begin{aligned} \boldsymbol{S}_n^T \boldsymbol{S}_n &= \frac{1}{2}\begin{bmatrix} \boldsymbol{S}_{n-1} & \boldsymbol{S}_{n-1} \\ \boldsymbol{S}_{n-1} & -\boldsymbol{S}_{n-1} \end{bmatrix}^T \boldsymbol{R}_n^T \boldsymbol{R} \begin{bmatrix} \boldsymbol{S}_{n-1} & \boldsymbol{S}_{n-1} \\ \boldsymbol{S}_{n-1} & -\boldsymbol{S}_{n-1} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} \boldsymbol{S}_{n-1}^T & \boldsymbol{S}_{n-1}^T \\ \boldsymbol{S}_{n-1}^T & -\boldsymbol{S}_{n-1}^T \end{bmatrix}\begin{bmatrix} \boldsymbol{S}_{n-1} & \boldsymbol{S}_{n-1} \\ \boldsymbol{S}_{n-1} & -\boldsymbol{S}_{n-1} \end{bmatrix} \\ &= \frac{1}{2}\begin{bmatrix} 2\boldsymbol{S}_{n-1}^T \boldsymbol{S}_{n-1} & \boldsymbol{0}_{n-1} \\ \boldsymbol{0}_{n-1} & 2\boldsymbol{S}_{n-1}^T \boldsymbol{S}_{n-1} \end{bmatrix} = \begin{bmatrix} \boldsymbol{I}_{n-1} & \boldsymbol{0}_{n-1} \\ \boldsymbol{0}_{n-1} & \boldsymbol{I}_{n-1} \end{bmatrix} = \boldsymbol{I}_n \end{aligned} \tag{7.55}$$

As a slant matrix $\boldsymbol{S}_n$ is closely related to a Hadamard matrix $\boldsymbol{H}_n$, the sequencies of their corresponding rows are the same. The same re-ordering method given in Eq. 7.35 can be used to rearrange the rows in $\boldsymbol{S}_n$ in ascending order of their sequencies. Based on the recursive definition in Eq.7.51, and after conversion to sequency order, the slant matrices of the next two levels for $n = 2$ and $n = 3$ can generate:

$$\boldsymbol{S}_2 = \frac{1}{2}\begin{bmatrix} 1.00 & 1.00 & 1.00 & 1.00 \\ 1.34 & 0.45 & -0.45 & -1.34 \\ 1.00 & -1.00 & -1.00 & 1.00 \\ 0.45 & -1.34 & 1.34 & -0.45 \end{bmatrix}$$

and

$$\boldsymbol{S}_3 = \frac{1}{\sqrt{8}}\begin{bmatrix} 1.00 & 1.00 & 1.00 & 1.00 & 1.00 & 1.00 & 1.00 & 1.00 \\ 1.53 & 1.09 & 0.65 & 0.22 & -0.22 & -0.65 & -1.09 & -1.53 \\ 1.34 & 0.45 & -0.45 & -1.34 & -1.34 & -0.45 & 0.45 & 1.34 \\ 0.68 & -0.10 & -0.88 & -1.66 & 1.66 & 0.88 & 0.10 & -0.68 \\ 1.00 & -1.00 & -1.00 & 1.00 & 1.00 & -1.00 & -1.00 & 1.00 \\ 1.00 & -1.00 & -1.00 & 1.00 & -1.00 & 1.00 & 1.00 & -1.00 \\ 0.45 & -1.34 & 1.34 & -0.45 & -0.45 & 1.34 & -1.34 & 0.45 \\ 0.45 & -1.34 & 1.34 & -0.45 & 0.45 & -1.34 & 1.34 & -0.45 \end{bmatrix} \tag{7.56}$$

We can make the following observations of the slant matrix:

- In particular, the second row with sequency of 1 has a negative linear slop, thereby the name "slant" matrix.

- Unlike the Walsh-Hademard matrix, the slant matrix $\boldsymbol{S}^T \neq \boldsymbol{S}$ is not symmetric;
- Like the Walsh-Hadamard matrix, the sequency of the row and column vectors of the slant matrix increases from 0 of the first row/column to the last one; However, we prefer to treat the *row* vectors as the orthogonal basis vectors that span a vector space, i.e,

$$\boldsymbol{S}_n = \begin{bmatrix} \boldsymbol{s}_0^T \\ \vdots \\ \boldsymbol{s}_{N-1}^T \end{bmatrix}$$

7.2.2 Fast Slant Transform

Given an orthogonal matrix $\boldsymbol{S}_n$, an orthogonal transform of an N-D vector $\boldsymbol{x}$ can be defined as:

$$\boldsymbol{X} = \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \boldsymbol{S}\boldsymbol{x} = \begin{bmatrix} \boldsymbol{s}_0^T \\ \vdots \\ \boldsymbol{s}_{N-1}^T \end{bmatrix} \boldsymbol{x}, \tag{7.57}$$

or in component form:

$$X[k] = \boldsymbol{s}_k^T \boldsymbol{x} = < \boldsymbol{s}_k, \boldsymbol{x} >$$

i.e., $X[k]$ is the projection of the signal vector $\boldsymbol{x}$ onto the kth basis vector $\boldsymbol{s}_k$. The inverse transform reconstructs the signal from its transform coefficients:

$$\boldsymbol{x} = \boldsymbol{S}^T \boldsymbol{X} = [\boldsymbol{s}_0, \cdots, \boldsymbol{s}_{N-1}] \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \sum_{k=0}^{N-1} X[k] \boldsymbol{s}_k \tag{7.58}$$

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 $\boldsymbol{x}$ of size $N = 2^3 = 8$ is:

$$\boldsymbol{X} = \boldsymbol{S}_3 \boldsymbol{x} = \frac{1}{\sqrt{2}} \boldsymbol{R}_3 \begin{bmatrix} \boldsymbol{S}_2 & \boldsymbol{S}_2 \\ \boldsymbol{S}_2 & -\boldsymbol{S}_2 \end{bmatrix} \begin{bmatrix} x[0] \\ \vdots \\ x[3] \\ x[4] \\ \vdots \\ x[7] \end{bmatrix} = \frac{1}{\sqrt{2}} \boldsymbol{R}_3 \begin{bmatrix} \boldsymbol{S}_2 \boldsymbol{x}_1 \\ \boldsymbol{S}_2 \boldsymbol{x}_2 \end{bmatrix} \tag{7.59}$$

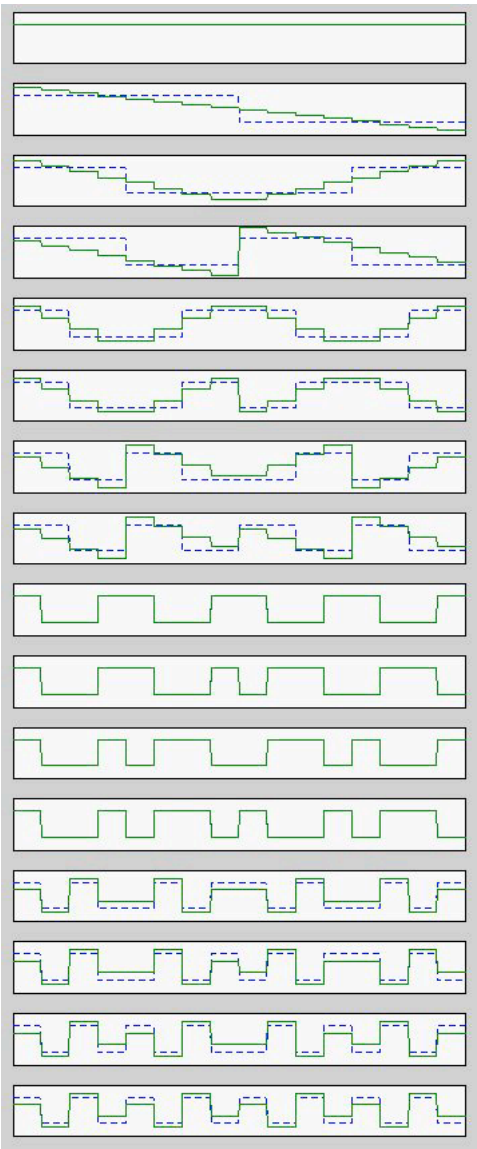

Figure 7.6 A comparison of slant (solid lines) and Hadamard (dashed lines) matrices

where

$$\boldsymbol{x}_1 = \begin{bmatrix} x[0] \\ \vdots \\ x[3] \end{bmatrix} + \begin{bmatrix} x[4] \\ \vdots \\ x[7] \end{bmatrix}, \qquad \boldsymbol{x}_2 = \begin{bmatrix} x[0] \\ \vdots \\ x[3] \end{bmatrix} - \begin{bmatrix} x[4] \\ \vdots \\ x[7] \end{bmatrix}$$

We see that an 8-point slant transform is converted into two 4-point slant transforms, each of which can be in turn converted to two 2-point transforms. This recursive process is illustrated in the diagram in Fig.7.7. The three nested boxes

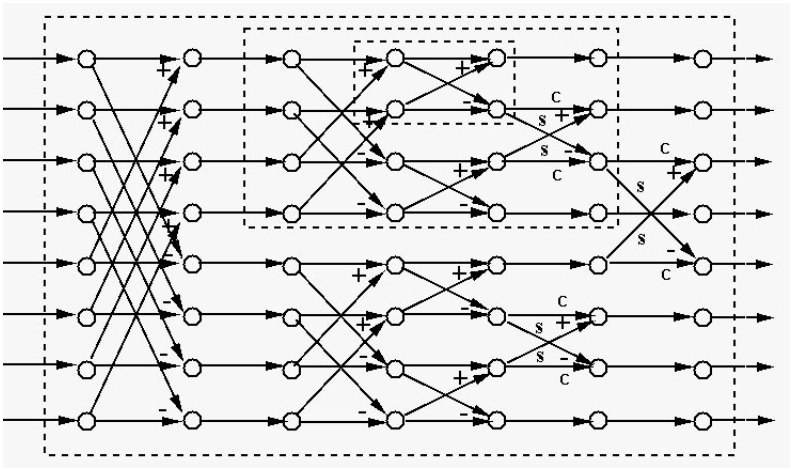

Figure 7.7 A recursive algorithm of fast slant transform
The three nested boxes (dashed line) are for 8, 4 and 2-point transforms, respectively. Letter c and s represents $\cos\theta_n$ and $\sin\theta_n$ for the rotation for each of the transforms (except for $N = 2$).

(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 WHT algorithm shown in Fig.7.4, 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,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 {
    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 7.2: The sequency ordered WHT of an 8-point signal vector $\boldsymbol{x} = [0, 0, 2, 3, 4, 0, 0, 0]^T$ can be obtained by matrix multiplication:

$$\boldsymbol{X} = \boldsymbol{S}_3\boldsymbol{x} = [3.18, 0.39, -3.64, -0.03, 1.77, -1.06, -0.16, 1.11]^T \tag{7.60}$$

where $\boldsymbol{S}_3$ is given in Eq.7.56. The inverse transform will bring the original signal $\boldsymbol{x}$ back:

$$\boldsymbol{S}_3^{-1}\boldsymbol{X} = \boldsymbol{S}_3^T\boldsymbol{X} = \boldsymbol{x} = [0, 0, 2, 3, 4, 0, 0, 0]^T \tag{7.61}$$

7.3 The Haar Transform

7.3.1 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 in the sense that it is also a special type of the wavelet transforms to be discussed in a later chapter.

The family of Haar functions $h_n(t),\ (n = 0, 1, 2, \cdots)$ are defined on the interval $0 \le t \le 1$. Except $h_0(t) = 1$ defined as a constant 1, the shape of the nth function $h_n(t)$ for $n > 0$ are determined by two parameters p and q, which are related to k by:

$$n = 2^p + q - 1 \tag{7.62}$$

For any given $n > 0$, p and q are uniquely determined so that 2^p is the largest power of 2 contained in n, i.e., $2^p < n$ or $0 \le p < log_2 k$, and $q - 1$ is the remainder, i.e., $q - 1 = n - 2^p$ or $1 \le q \le 2^p$. For example, the values of p and q corresponding to $n = 1, \cdots, 15$ are shown in the table:

n	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	1	1	2	1	2	3	4	1	2	3	4	5	6	7	8

(7.63)

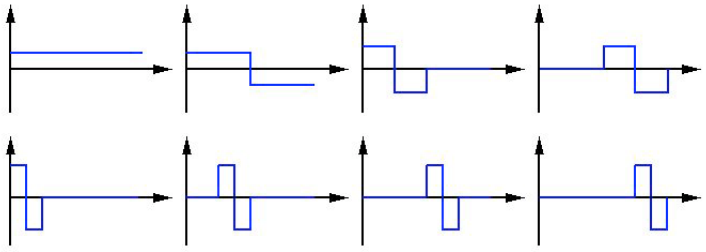

Figure 7.8 The 8 basis functions for the Haar transform

Now the family of Haar functions can be defined in terms of p and q as:

- When $n = 0$:

$$h_0(t) = 1, \qquad (0 \leq t < 1) \tag{7.64}$$

- When $n > 0$,

$$h_k(t) = \begin{cases} 2^{p/2} & (q-1)/2^p \leq t < (q-0.5)/2^p \\ -2^{p/2} & (q-0.5)/2^p \leq t < q/2^p \\ 0 & \text{otherwise} \end{cases} \tag{7.65}$$

The first $N = 8$ Haar functions are shown in Fig. 7.8. We see that the Haar functions $h_n(t)$ for all $n > 0$ 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 $n = 3$, then $p = 1$, $q = 2$, and we have

$$h_3(t) = \begin{cases} \sqrt{2} & 0.5 \leq t < 0.75 \\ -\sqrt{2} & 0.75 \leq t < 1 \\ 0 & 0 \leq t < 0.5 \end{cases} \tag{7.66}$$

These Haar functions are obviously orthonormal:

$$< h_n(t), h_\nu(t) >= \int_0^1 h_n(t) h_\nu(t) dt = \delta[n-\nu] = \begin{cases} 1 & n = \nu \\ 0 & n \neq \nu \end{cases} \tag{7.67}$$

and they can be used as the basis functions to span a function space over $0 \leq t < 1$. A signal function $x(t)$ in this space can be expressed as a linear combination of these Haar functions:

$$x(t) = \sum_{n=0}^{\infty} X[n] h_n(t) \tag{7.68}$$

where the nth coefficient $X[n]$ can be obtained as the projection of $x(t)$ onto the nth basis function $h_n(t)$:

$$X[n] =< x(t), h_n(t) >= \int_0^1 x(t) h_n(t) dt \tag{7.69}$$

This is the continuous Haar transform of the signal $x(t)$. Let us further consider what each coefficient $X[n]$ represents. First, when $n = 0$, the coefficient

$$X[0] = \int_0^1 x(t)h_0(t)dt = \int_0^1 x(t)dt \tag{7.70}$$

represents the average or DC component of the signal, same as all orthogonal transforms discussed before. Next, each of the coefficients $X[n]$ for $n > 0$ represents two specific aspects of the signal characteristics, determined by p and q, respectively:

- certain detailed feature of the signal, in the form of the difference between two consecutive segments of the signal, at different time scales
- and when the detailed feature occurs in time.

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 in the second half of its duration.

In light of these two characteristics, especially the second one, it is interesting to compare the Haar transform with all the orthogonal transforms discussed before, including Fourier transform, cosine transform, Walsh-Hadamard transform. What all of these transforms, including 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 frequency/sequency/scales. However, none of these transforms is able to indicate when in time such details occur, except the Haar transform, which can represent not only the details of different scales, but also their temporal positions. It is this feature that distinguishes the Haar transform from all of other orthogonal transforms, and for this reason, the Haar transform is also a special form of the wavelet transform to be discussed later.

7.3.2 Discrete Haar Transform (DHT)

The discrete Haar transform (DHT) is defined based on the family of Haar functions. Specifically, by sampling each of the first N Haar functions $h_n(t)$ $(n = 0, \cdots, N-1)$ at time moments $t = m/N$ $(m = 0, 1, 2, \cdots, N-1)$, we get N orthogonal vectors. Moreover, if a scaling factor $1/\sqrt{N}$ is included, these vectors become orthonormal:

$$< \boldsymbol{h}_n, \boldsymbol{h}_\nu >= \boldsymbol{h}_n^T \boldsymbol{h}_\nu = \delta[m-n] \tag{7.71}$$

These N orthonormal vectors form a basis that spans the N-dimensional vector space, and they form an N by N DHT matrix $\boldsymbol{H}$ (not to be confused with the

WHT matrix):

$$\boldsymbol{H} = [\boldsymbol{h}_0, \cdots, \boldsymbol{h}_{N-1}], \quad \text{or} \quad \boldsymbol{H}^T = \begin{bmatrix} \boldsymbol{h}_0^T \\ \vdots \\ \boldsymbol{h}_{N-1}^T \end{bmatrix} \tag{7.72}$$

which is obviously real and orthogonal:

$$\boldsymbol{H} = \boldsymbol{H}^*, \quad \boldsymbol{H}^{-1} = \boldsymbol{H}^T, \quad \text{i.e.} \quad \boldsymbol{H}^T\boldsymbol{H} = \boldsymbol{I} \tag{7.73}$$

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:

$$\boldsymbol{H}_1^T = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 0.71 & 0.71 \\ 0.71 & -0.71 \end{bmatrix} \tag{7.74}$$

  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 $N = 4$

$$\boldsymbol{H}_2^T = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix} = \begin{bmatrix} 0.50 & 0.50 & 0.50 & 0.50 \\ 0.50 & 0.50 & -0.50 & -0.50 \\ 0.71 & -0.71 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.71 & -0.71 \end{bmatrix} \tag{7.75}$$

  In comparison, the transform matrices $\boldsymbol{C}$ and $\boldsymbol{H}_w$ for the other two real transforms DCT and WHT are also listed below:

$$\boldsymbol{C}^T = \begin{bmatrix} 0.50 & 0.50 & 0.50 & 0.50 \\ 0.65 & 0.27 & -0.27 & -0.65 \\ 0.50 & -0.50 & -0.50 & 0.50 \\ 0.27 & -0.65 & 0.65 & -0.27 \end{bmatrix}, \quad \boldsymbol{H}_w^T = \begin{bmatrix} 0.50 & 0.50 & 0.50 & 0.50 \\ 0.50 & 0.50 & -0.50 & -0.50 \\ 0.50 & -0.50 & -0.50 & 0.50 \\ 0.50 & -0.50 & 0.50 & -0.50 \end{bmatrix} \tag{7.76}$$

  Here a subscript w is included in the WHT matrix $\boldsymbol{H}_w$ to tell it apart from the DHT matrix $\boldsymbol{H}$. We see that all three matrices have identical first rows representing the DC component of the signal, and the elements of their second rows have the same polarities, although different values, representing the difference between the the first and second halves of the signal. However, the third and forth rows are quite different. In the case of DCT and WHT, they represent progressively higher frequency/sequency components in the signal, but in the case of DHT, these rows represent the same level of details in signal variation, and their different temporal locations (either in the first or second half), at a finer scale than the second row.

- When $N = 8$, we have

$$\boldsymbol{H}_3^T = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{bmatrix} \begin{matrix} \varphi_0(t) \\ \psi_0(t) \\ \psi_{1,0}(t) \\ \psi_{1,1}(t) \\ \psi_{2,0}(t) \\ \psi_{2,1}(t) \\ \psi_{2,2}(t) \\ \psi_{2,3}(t) \end{matrix} \tag{7.77}$$

  It is obvious that the additional four rows represent signal variations and their temporal positions at a finer still scale than the previous two rows. Note that each row is also labeled as a function ($\varphi(t)$ for the first row and $\psi(t)$ 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 $\boldsymbol{x} = [x[0], \cdots, x[N-1]]^T]$ can be expressed as a linear combination of the column vectors $\boldsymbol{h}_n$ $(n = 0, \cdots, N-1)$ of the DHT matrix $\boldsymbol{H}$:

$$\boldsymbol{x} = \boldsymbol{H}\boldsymbol{X} = [\boldsymbol{h}_0, \cdots, \boldsymbol{h}_{N-1}] \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \sum_{n=0}^{N-1} X[n]\boldsymbol{h}_n \tag{7.78}$$

This is the inverse discrete Haar transform (IDHT), where $X[n]$ is the coefficient for the nth vector $\boldsymbol{h}_n$, which can be obtained as the projection of the signal vector $\boldsymbol{x}$ onto the nth basis vector $\boldsymbol{h}_n$:

$$X[n] = < \boldsymbol{x}, \boldsymbol{h}_n > = \boldsymbol{h}_n^T \boldsymbol{x} \qquad (n = 0, 1, \cdots, N-1) \tag{7.79}$$

or in matrix form:

$$\boldsymbol{X} = \boldsymbol{H}^{-1}\boldsymbol{x} = \boldsymbol{H}^T\boldsymbol{x} = \begin{bmatrix} \boldsymbol{h}_0^T \\ \vdots \\ \boldsymbol{h}_{N-1}^T \end{bmatrix} \boldsymbol{x} \tag{7.80}$$

This is the forward discrete Haar transform (DHT), which can also be obtained by pre-multiplying $\boldsymbol{H}^{-1}$ on both sides of the IDHT equation above. The DHT pair can be written as:

$$\begin{cases} \boldsymbol{X} = \boldsymbol{H}^T\boldsymbol{x} & \text{(forward)} \\ \boldsymbol{x} = \boldsymbol{H}\boldsymbol{X} & \text{(inverse)} \end{cases} \tag{7.81}$$

Example 7.3: The Haar transform coefficients of an 8-point signal $\boldsymbol{x} = [0, 0, 2, 3, 4, 0, 0, 0]^T$ can be obtained by the DHT as:

$$\boldsymbol{X} = \boldsymbol{H}^T\boldsymbol{x} = [3.18, 0.35, -2.50, 2.0, 0.0, -0.71, 2.83, 0.0]^T \tag{7.82}$$

where the 8-point Haar transform matrix is given in Eq.7.77. Here, same as in DCT, WHT and ST, $X[0] = 3.18$ and $X[1] = 0.35$ represent respectively the average and the 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, $X[4], \cdots, X[7]$ 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.7.78.

7.3.3 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.7.9 for the for an $N = 8$ point DHT transforms. The forward transform $\boldsymbol{X} = \boldsymbol{H}_3^T \boldsymbol{x}$ can be written in matrix form as:

$$\begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ X[3] \\ X[4] \\ X[5] \\ X[6] \\ X[7] \end{bmatrix} = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \\ x[4] \\ x[5] \\ x[6] \\ x[7] \end{bmatrix}$$

$$= \begin{bmatrix} (1 & 1 & 1 & 1 & 1 & 1 & 1 & 1) / \sqrt{2}^3 \\ (1 & 1 & 1 & 1 & -1 & -1 & -1 & -1) / \sqrt{2}^3 \\ (1 & 1 & -1 & -1 & 0 & 0 & 0 & 0) / \sqrt{2}^2 \\ (0 & 0 & 0 & 0 & 1 & 1 & -1 & -1) / \sqrt{2}^2 \\ (1 & -1 & 0 & 0 & 0 & 0 & 0 & 0) / \sqrt{2} \\ (0 & 0 & 1 & -1 & 0 & 0 & 0 & 0) / \sqrt{2} \\ (0 & 0 & 0 & 0 & 1 & -1 & 0 & 0) / \sqrt{2} \\ (0 & 0 & 0 & 0 & 0 & 0 & 1 & -1) / \sqrt{2} \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \\ x[4] \\ x[5] \\ x[6] \\ x[7] \end{bmatrix} \tag{7.83}$$

By inspection of this matrix multiplication of this forward transform, we see that each of the last four coefficients $X[4], \cdots, X[7]$ in the second half of vector $\boldsymbol{X}$ 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 $X[3]$

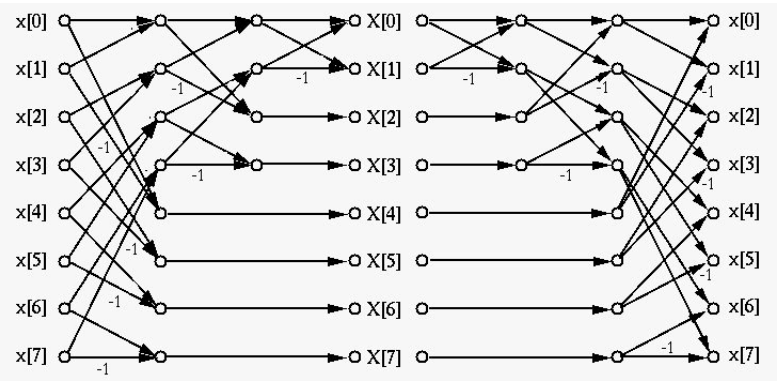

Figure 7.9 The fast Haar transform algorithm
The forward DHT transform shown on the left of the diagram converts the signal $\boldsymbol{x}$ to it DHT coefficients $\boldsymbol{X}$ in the middle, which is then inverse transformed back to time domain by the IDHT shown on the right of the diagram.

of the first half of $\boldsymbol{X}$ can 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.7.9, 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 $X[0], \cdots, X[7]$ 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 $\boldsymbol{x} = \boldsymbol{H}_3\boldsymbol{X}$ can also be written in matrix form:

$$\begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \\ x[4] \\ x[5] \\ x[6] \\ x[7] \end{bmatrix} = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & \sqrt{2} & 0 & 2 & 0 & 0 & 0 \\ 1 & 1 & \sqrt{2} & 0 & -2 & 0 & 0 & 0 \\ 1 & 1 & -\sqrt{2} & 0 & 0 & 2 & 0 & 0 \\ 1 & 1 & -\sqrt{2} & 0 & 0 & -2 & 0 & 0 \\ 1 & -1 & 0 & \sqrt{2} & 0 & 0 & 2 & 0 \\ 1 & -1 & 0 & \sqrt{2} & 0 & 0 & -2 & 0 \\ 1 & -1 & 0 & -\sqrt{2} & 2 & 0 & 0 & 2 \\ 1 & -1 & 0 & -\sqrt{2} & 0 & 0 & 0 & -2 \end{bmatrix} \begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ X[3] \\ X[4] \\ X[5] \\ X[6] \\ X[7] \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & -1 & 0 & 0 & 0 \\ 1 & 1 & -1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & -1 & 0 & 0 & -1 & 0 & 0 \\ 1 & -1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & -1 & 0 & 1 & 0 & 0 & -1 & 0 \\ 1 & -1 & 0 & -1 & 0 & 0 & 0 & 1 \\ 1 & -1 & 0 & -1 & 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} X[0]/\sqrt{2}^3 \\ X[1]/\sqrt{2}^3 \\ X[2]/\sqrt{2}^2 \\ X[3]/\sqrt{2}^2 \\ X[4]/\sqrt{2} \\ X[5]/\sqrt{2} \\ X[6]/\sqrt{2} \\ X[7]/\sqrt{2} \end{bmatrix} \tag{7.84}$$

By inspection we see that this matrix multiplication can also be carried out in $\log_2 8 = 3$ stages as shown on the right of the diagram in Fig.7.9. Again, the output of each stage needs to be divided by $\sqrt{2}$.

Following this example for $N = 8$, we see that in general an N-point DHT can be carried out in $\log_2 N$ stages, and the number of operations for each stage is always half of that of the previous stage (i.e., $N, N/2, \cdots, 2$), therefore the total number of operations for the DHT is

$$\frac{N}{1} + \frac{N}{2} + \frac{N}{4} + \frac{N}{8} + \cdots + 2 = N \sum_{k=0}^{n-1} \left(\frac{1}{2}\right)^k = 2N(1 - \frac{1}{2^n}) < 2N \qquad (7.85)$$

i.e., the computational complexity of this DHT algorithm is $O(N)$, much more efficient compared to the typical complexity of $O(N \log_2 N)$ for most of other transforms such as FFT, DCT and WHT. 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);
}
```

7.3.4 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 later. 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 $h[n]$:

$$x'[n] = x[n] * h[n] = \sum_m h[m]x[n-m] \tag{7.86}$$

In particular, for the Haar transform, we consider four different 2-point convolution kernels:

- $h_0[0] = h_0[1] = 1/\sqrt{2}$
- $h_1[0] = 1/\sqrt{2}, \;\; h_1[1] = -1/\sqrt{2}$
- $g_0[0] = g_0[1] = 1/\sqrt{2}$
- $g_1[0] = -1/\sqrt{2}, \;\; h_1[1] = 1/\sqrt{2}$

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, if the kernel is h_0 (or g_0), we have

$$y[n] = x[n] * h_0[n] = \sum_{m=0}^{1} h_0[m]x[n-m] = \frac{1}{\sqrt{2}}(x[n-1] + x[n]) \tag{7.87}$$

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 some low frequencies). On the other hand, if the kernel is h_1, then

$$y[n] = x[n] * h_1[n] = \sum_{m=0}^{1} h_0[m]x[n-m] = \frac{1}{\sqrt{2}}(x[n-1] - x[n]) \tag{7.88}$$

This can be considered as a highpass filter as the output $y[n]$ represents the difference of the two consecutive data points (corresponding to some high frequencies). Finally, if the kernel is g_1, the convolution is also a highpass filter:

$$y[n] = x[n] * g_1[n] = \frac{1}{\sqrt{2}}(x[n] - x[n-1]) = -x[n] * h_1[n] \tag{7.89}$$

Due to the convolution theorem of Z-transform, these convolutions can also be represented as multiplications in Z-domain:

$$Y(z) = H_i(z)X(z), \quad Y(z) = G_i(z)X(z), \qquad (i = 0, 1) \tag{7.90}$$

Now the forward transform of the fast DHT shown on the left of Fig. 7.9 can be considered as a recursion of the following two operations:

- Operation 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 D (difference or detail): a highpass filter implemented as $y[n] = x[n] * h_1[n]$, also followed by downsampling.

For example, operation A applied to a set of 8-point sequence $x[0], \cdots, x[7]$ 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 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 A and D is applied first to the N-point signal $x[n]$ $(n = 0, \cdots, N-1)$, and then recursively to the output of operation A in the previous recursion. As the data size is reduced by half by 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.7.10.

The inverse transform of the fast algorithm (right half of Fig. 7.9) can also be viewed as a recursion of two operations:

- Operation 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 D: a highpass filtered by $y[n] = x[n] * g_1[n]$, applied to the upsampled input data.

For example, when operation 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 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 A will be applied to this 2-point sequence, while operation 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 $x[0], \cdots, x[N-1]$ are reconstructed. This is the filter bank implementation of the IDHT, as illustrated on the right of Fig.7.10.

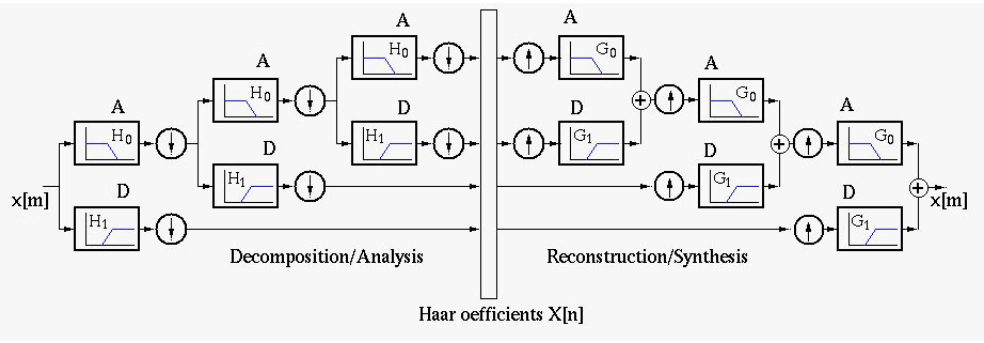

Figure 7.10 Filter bank implementation of DHT
Here H_0 and $G_0(z)$ are lowpass filters and H_1 and G_1 are highpass filters. The up and down arrows represent upsampling and downsampling, respectively.

7.4 Two-dimensional Transforms

Same as the discrete Fourier and cosine transforms, all three of the transform methods discussed above can also be applied to a 2-D signal $x[m,n]$ $(m=0,\cdots,M-1, n=0,\cdots,N-1)$, 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 for the Walsh-Hadamard, slant and Haar transforms by a generic orthogonal matrix $\boldsymbol{A}$. The forward and inverse 2-D transform of a 2-D signal are defined respectively as:

$$\begin{cases} \boldsymbol{X}_{M\times N} = \boldsymbol{A}_M^T \boldsymbol{x}_{M\times N} \boldsymbol{A}_N & \text{(forward)} \\ \boldsymbol{x}_{M\times N} = \boldsymbol{A}_M \boldsymbol{X}_{M\times N} \boldsymbol{A}_N^T & \text{(inverse)} \end{cases} \tag{7.91}$$

where $\boldsymbol{x}$ and $\boldsymbol{X}$ are respectively the 2-D M by N signal matrix and its transform coefficient matrix, and $\boldsymbol{A}_N$ and $\boldsymbol{A}_N$ are respectively M by M matrix for the column transforms and N by N matrix for the row transforms. The inverse transform (second equation) expresses the given 2-D signal $\boldsymbol{x}$ as a linear combination of a set of N^2 2-D basis functions:

$$\begin{aligned} \boldsymbol{x} &= [\boldsymbol{a}_0,\cdots,\boldsymbol{a}_{M-1}] \begin{bmatrix} X[0,0] & \cdots & X[0,N-1] \\ \vdots & \ddots & \vdots \\ X[M-1,0] & \cdots & X[M-1,N-1] \end{bmatrix} \begin{bmatrix} \boldsymbol{a}_0^T \\ \vdots \\ \boldsymbol{a}_{N-1}^T \end{bmatrix} \\ &= \sum_{k=0}^{M-1}\sum_{l=0}^{N-1} X[k,l]\boldsymbol{a}_k\boldsymbol{a}_l^T = \sum_{k=0}^{M-1}\sum_{l=0}^{N-1} X[k,l]\boldsymbol{B}_{kl} \end{aligned} \tag{7.92}$$

where $\boldsymbol{B}_{kl} = \boldsymbol{a}_k\boldsymbol{a}_l^T$ is the kl-th 2-D (M by N) basis function, weighted by the corresponding coefficient $X[k,l]$. Also, same as in the cases of DFT (Eq.4.214) and DCT (Eq.6.96), this coefficient can be obtained as the projection (inner

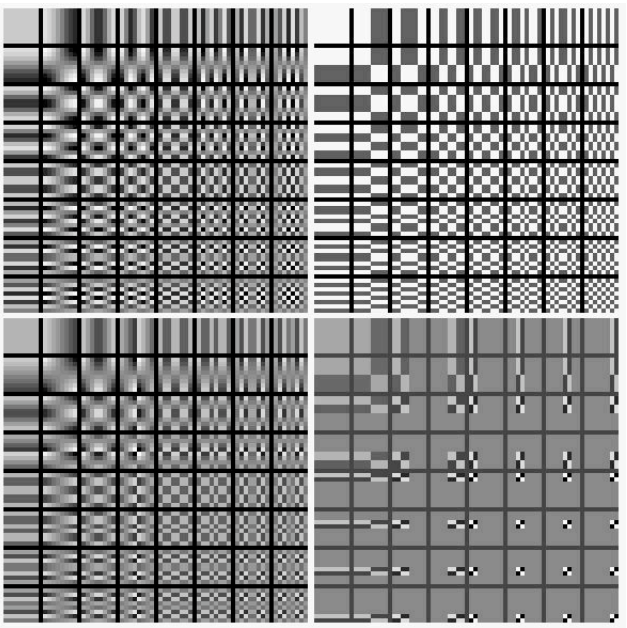

Figure 7.11 The basis functions for different 2-D orthogonal transforms
DCT (top left), WHT (top right), ST (lower left), and DHT (lower right). In all cases the DC component is at the top-left corner, and the farther away from the corner, the higher frequency/sequency contents or scales of details are represented. Specially, the spatial positions, as well as different levels of scales, are also represented in the Haar basis.

product) of $\boldsymbol{x}$ onto the kl-th basis function $\boldsymbol{B}_{kl}$:

$$X[k,l] = \boldsymbol{a}_k^T \begin{bmatrix} x[0,0] & \cdots & x[0,N-1] \\ \vdots & \ddots & \vdots \\ x[M-1,0] & \cdots & x[M-1,N-1] \end{bmatrix} \boldsymbol{a}_l$$
$$= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x[m,n] B_{kl}[m,n] = < \boldsymbol{x}, \boldsymbol{B}_{kl} > \tag{7.93}$$

We see that this coefficient is the projection of the 2-D signal $\boldsymbol{x}$ onto the 2-D basis function. When $M = N = 8$, the $8 \times 8 = 64$ such 2-D basis functions corresponding to Walsh-Hadamard, slant, and Haar are shown in Fig.7.11. For comparison, the basis functions corresponding basis functions for discrete cosine transform are also shown.

All of these transform methods can be used for filtering. Fig.7.12 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 components. In the low-pass filtering

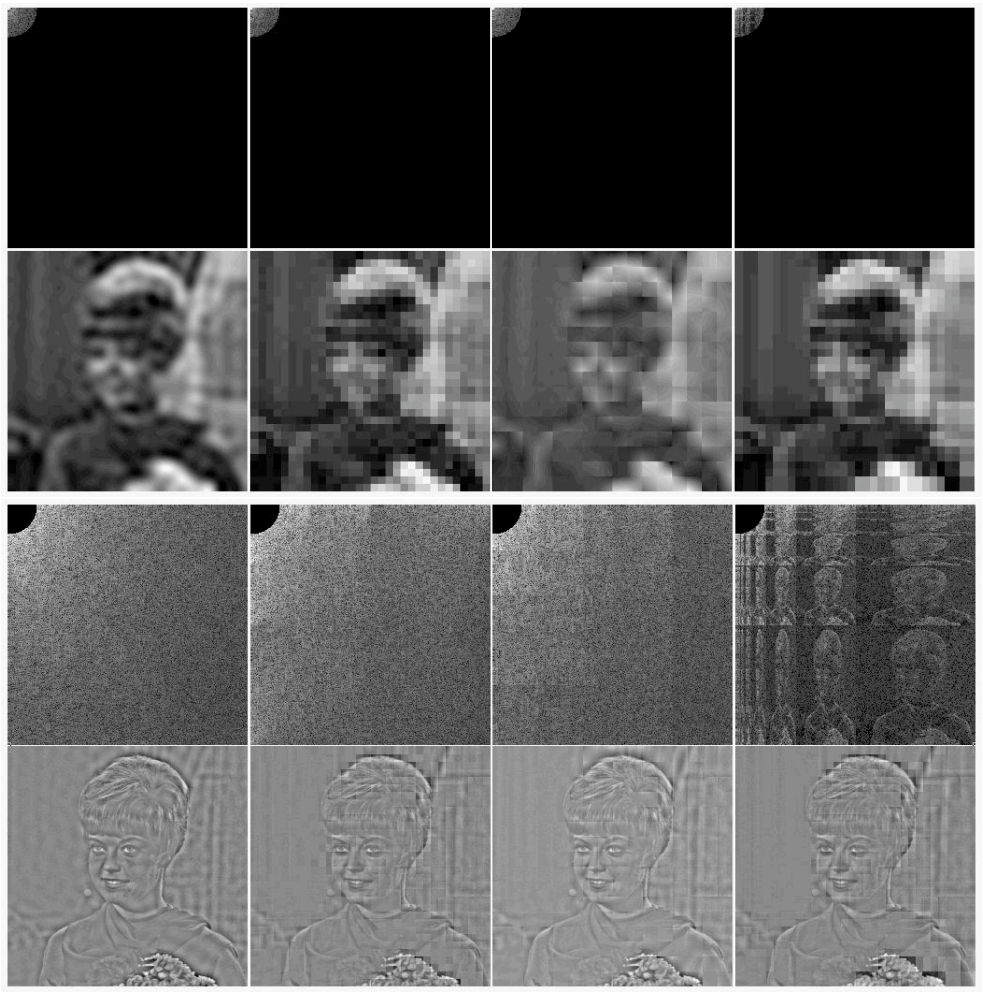

Figure 7.12 Low-pass and high-pass filtering based on different 2-D transforms
The spectrum of each of the transform methods (from left to right: DCT, WHT, ST and DHT) after filtering is given in the first (low-pass) and third (high-pass) row, and the corresponding filtered image is given directly below each spectrum.

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. Therefore all of these transform methods lend themselves to data compression, same as the Fourier transform.

On the other hand, these transform methods use different basis functions, which may be suitable for different types of signals. Most obviously, like the DFT, the DCT has sinusoidal basis functions and is therefore suitable for rep-

resenting 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, if the signals are square wave like in nature, then the WHT may be more suitable to use, as the signal may be most effectively represented by a small subset of the basis functions, so that 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.7.11, is a constant representing the DC component of the 2-D signal. However, the rest of the basis functions are quite different. For example, the lower-right quarter of the image shows the last 16 basis functions representing not only the details in the signal at the highest scale level, 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. It is this unique characteristic that makes the Haar transform also a special case of the wavelet transform, as we will see in a later chapter.

8 Karhunen-Loeve Transform and Principal Component Analysis

8.1 Stochastic Signal and Signal Correlation

8.1.1 Signals as Stochastic Processes

In all of our previous discussions, a 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, such as weather parameters (temperature, precipitation, etc.), are not deterministic, in the sense that multiple measurements of the same variable may be similar but not identical. While such probabilistic nature of these signals could be caused by some inevitable measurement errors, we also realize that many natural processes are affected by a large number of factors which are simply impossible to model precisely in terms of how they affect the variables of interest. Consequently the measured signals seem to be contaminated by some random noise.

The time signal $x(t)$ of such a non-deterministic variable can be considered as a *stochastic process* or *random process*, of which each time sample can be treated as a random variable with certain probability distribution. In this chapter, we will consider an spectial orthogonal transform that can be applied to stochastic signals, similar to the way all orthogonal transforms discussed previously are applied to deterministic signals, so that the subsequent signal processing and analysis can be carried out more effectively and conveniently.

First let us review the following concepts of a stochastic process $x(t)$.

- The *mean function* of $x(t)$ is the expectation of the stochastic process:

$$\mu_x(t) = \int x(t)p(x_t)dx = E[x(t)] \tag{8.1}$$

  where $p(x_t)$ is the probability density function of the variable $x(t)$. 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 $x(t)$. Therefore, without loss of generality, we can always assume that a given process $x(t)$ is centered with $\mu_x(t) = 0$.

- The *auto-covariance function* of $x(t)$ is defined as

$$Cov_x(t,\tau) = \sigma_x^2(t,\tau) = \int\int (x(t)-\mu_x(t))\ (\overline{x}(\tau)-\overline{\mu}_x(\tau))p(x_t,x_\tau)dt\ d\tau$$
$$= E[(x(t)-\mu_x(t))\ (\overline{x}(\tau)-\overline{\mu}_x(\tau))] = E[x(t)\overline{x}(\tau)] - \mu_x(t)\overline{\mu}_x(\tau)$$

  where $p(x_t, x_\tau)$ is the joint probability density function of $x(t)$ and $x(\tau)$. When $t=\tau$, the covariance $\sigma^2(t,t) = Cov_x(t) = Var_x(t) = E[|x(t)|^2]$ becomes the variance of the signal at t. Without loss of generality, we can always assume $x(t)$ to be centered with $\mu_x(t)=0$, and the covariance $\sigma_x^2(t,\tau)$ becomes:

$$\sigma_x^2(t,\tau) = E[x(t)\overline{x}(\tau)] =< x(t), x(\tau) > \tag{8.2}$$

  which can be considered as an inner product of the two variables $x(t)$ and $x(\tau)$ (Eq.2.27 in Chapter 2). In particular, if $\sigma_x^2(t,\tau) =< x(t), x(\tau) >= 0$, the two variables are 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 $\sigma_x(\tau)$:

$$r_x(t,\tau) = \frac{\sigma_x^2(t,\tau)}{\sqrt{\sigma_x^2(t)\ \sigma_x^2(\tau)}} = \frac{< x(t), x(\tau) >}{\sqrt{< x(t), x(t) >< x(\tau), x(\tau) >}} \tag{8.3}$$

  Due to the Cauchy-Schwarz inequality: $| < x, y > |^2 \leq < x, x >< y, y >$, we see that $|r_x(t,\tau)| \leq 1$, and $r_x(t,\tau) = 1$ if and only if $t=\tau$. This results indicates that the similarity between any two different variables $x(t)$ and $x(\tau)$ is always smaller than that of a variable $x(t)$ to itself, which is always 1 or one hundred percent.

Moreover, if the joint probability density function of the random process $x(t)$ does not change over time, then $x(t)$ is called a *stationary process*, and the following hold for any τ:

$$\mu_x(t) = \mu_x(t-\tau), \qquad \sigma_x^2(t,\tau) = \sigma_x^2(t-\tau), \qquad r_x(t,\tau) = r_x(t-\tau) \tag{8.4}$$

If these equations still hold while the joint density function may not be necessarily invariant over time, then $x(t)$ is called *weak-sense stationary* or *wide-sense stationarity (WSS)*.

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 $x[i] = x(t_i)$ $(i = 0, \cdots, N-1)$, which can be represented by a random vector $\boldsymbol{x} = [x[0], \cdots, x[N-1]]^T$. Correspondingly, the mean and autocovariance/autocorrelation functions above become a vector and matrix, respectively:

- The *mean vector* of a random vector $\boldsymbol{x}$ is its expectation:

$$\boldsymbol{\mu}_x = E(\boldsymbol{x}) = [\mu[0], \cdots, \mu[N-1]]^T \tag{8.5}$$

  where $\mu[i] = E(x[i])$ is the mean of $x[i]$ $(i = 0, \cdots, N-1)$. Also, without loss of generality, we can always assume $\boldsymbol{x}$ is centered with $\boldsymbol{\mu} = \mathbf{0}$.

- The *covariance matrix* of a random vector $\boldsymbol{x}$ is defined as:

$$\boldsymbol{\Sigma}_x = E[(\boldsymbol{x}-\boldsymbol{\mu}_x)(\boldsymbol{x}-\boldsymbol{\mu}_x)^*] = E[\boldsymbol{x}\boldsymbol{x}^*] - \boldsymbol{\mu}_x\boldsymbol{\mu}_x^* = \begin{bmatrix} \sigma_0^2 & \cdots & \sigma_{0(N-1)}^2 \\ \vdots & \ddots & \vdots \\ \sigma_{(N-1)0}^2 & \cdots & \sigma_{N-1}^2 \end{bmatrix} \tag{8.6}$$

  where the element σ_{ij}^2 is the covariance of two $x[i]$ and $x[j]$:

$$\sigma_{ij}^2 = E[(x[i]-\mu[i])(\overline{x}[j]-\overline{\mu}[j])] = E(x[i]\overline{x}[j]) - \mu[i]\overline{\mu}[j], \quad (i,j=0,\cdots,N-1) \tag{8.7}$$

  When $\boldsymbol{\mu}_x = \boldsymbol{0}$, we have $\sigma_{ij} = E(x[i]\overline{x}[j]) =< x[i], x[j] >$. The ith component on the main diagonal is the variance $\sigma_i^2 = E[|x[i]-\mu[i]|^2]$ of the ith variable $x[i]$. Note that this covariance matrix $\boldsymbol{\Sigma}_x^* = \boldsymbol{\Sigma}_x$ is Hermitian and positive definite.
- The *correlation coefficient* between two random variables $x[i]$ and $x[j]$ is defined as the covariance σ_{ij}^2 normalized by σ_i and σ_j:

$$r_{ij} = \frac{\sigma_{ij}^2}{\sqrt{\sigma_i^2\ \sigma_j^2}} = \frac{< x[i], x[j] >}{\sqrt{< x[i], x[i] >< x[j], x[j] >}}, \quad (i,j=0,\cdots,N-1) \tag{8.8}$$

  which measures the similarity between the two variables. In matrix form, we have the *correlation matrix* composed of all correlation coefficients:

$$\boldsymbol{R}_x = \begin{bmatrix} r_0 & \cdots & r_{0(N-1)} \\ \vdots & \ddots & \vdots \\ r_{(N-1)0} & \cdots & r_{N-1} \end{bmatrix} \tag{8.9}$$

  where $|r_{ij}| \leq 1$ for all $i \neq j$, and $r_{ii} = r_i$ along the main diagonal of $\boldsymbol{R}_x$ is 1.

The true mean vector $\boldsymbol{\mu}_x$ and covariance matrix $\boldsymbol{\Sigma}_x$ of a random vector $\boldsymbol{x}$ are difficult to obtain as they depend on the joint probability density function $p(\boldsymbol{x})$, 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 $\{\boldsymbol{x}_k, (k=1,\cdots,K)\}$ be a set of K samples of the N-D random vector $\boldsymbol{x}$, then the mean vector and covariance matrix can be estimated as:

$$\hat{\boldsymbol{\mu}}_x = \frac{1}{K}\sum_{k=1}^{K}\boldsymbol{x}_k, \quad \text{and} \quad \hat{\boldsymbol{\Sigma}}_x = \frac{1}{K}\sum_{k=1}^{K}(\boldsymbol{x}_k-\boldsymbol{\mu}_x)(\boldsymbol{x}_k-\boldsymbol{\mu}_x)^* \tag{8.10}$$

As we can always trivially subtract the mean vector from each of the K samples so that they all have zero mean (centered), we can assume $\boldsymbol{\mu}_x = 0$ without loss of generality. Moreover, we further define a K by N matrix $\boldsymbol{D} = [\boldsymbol{x}_1,\cdots,\boldsymbol{x}_K]^T$ composed of the K sample vectors of zero mean as its row vectors, then the estimated covariance matrix can be expressed as:

$$\hat{\boldsymbol{\Sigma}}_x = \frac{1}{K}[\boldsymbol{D}^T\boldsymbol{D}^*]_{N\times N} \tag{8.11}$$

8.1.2 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 variable associated with a certain physical system, a signal tend to be continuously and relatively evenly distributed in either time or space, in the sense that any two near by samples of such a temporal or spatial signal are likely to be highly correlated. For example, given the current temperature as the signal sample $x(t)$, one could predict with reasonable confidence the temperature $x(t+\tau)$ in the near future (small τ) as the next sample to be fairly similar, in contrast with the temperature in the far future (large τ). In other words, the two signal samples $x(t)$ and $x(t+\tau)$ are highly correlated, i.e., $\sigma_x^2(t, t+\tau)$ has a large value or $r_x(t, t+\tau)$ is close to 1.

This common sense 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 locally (strictly speaking, in a closed system), concentrated energy tends to disperse over time, and differences in physical quantities (temperature, pressure, or density) tend to even out. In this physical world, any disruption or discontinuity, typically associated with some kind of energy surge, is a relatively rare and unlikely event.

On the other hand, we also observe that when the spatial or temporal interval between two signal samples increases, their correlation tends to reduce. While predicting the future temperature $x(t+\tau)$ based on the current one $x(t)$, one would be less confident when τ becomes larger. The correlation between two signal samples will eventually be totally diminished when they are so far apart from each other that they are not related anymore.

The signal characteristics of local correlation is reflected in the correlation matrix $\boldsymbol{R}_x$ of the signal. All elements along the main diagonal take the maximum value 1, representing the maximal self-correlation of each signal sample, and all off-diagonal elements $r_{ij} < 1$ takes a smaller value representing the cross-correlation between two signal samples $x[i]$ and $x[j]$. Moreover, due to local correlation, elements r_{ij} close to the main diagonal (small $|i-j|$) tends to take large values (close to 1) than those which are farther away from the main diagonal (large $|i-j|$). If we consider the correlation matrix as a landscape, then there is a ridge along its main diagonal.

Based on this observation, the signal can be modeled by a stationary first order Markov process (a memoryless random process) with correlation $0 \leq r \leq 1$ between two consecutive samples. In general, the correlation between any two samples $x[i]$ and $x[j]$ is $r_{ij} = r^{|i-j|}$, i.e., the correlation will reduce exponentially as a function of the distance between two samples. The correlation matrix of this

Markov chain is:

$$\boldsymbol{R}_x = \begin{bmatrix} 1 & r & r^2 & \cdots & r^{N-1} \\ r & 1 & r & \cdots & r^{N-2} \\ r^2 & r & 1 & \cdots & r^{N-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r^{N-1} & r^{N-2} & r^{N-3} & \cdots & 1 \end{bmatrix}_{N\times N} \tag{8.12}$$

This is a Toeplitz matrix with all elements along each diagonal being the same. This first-order Markov chain model will be used later.

We next consider a set of simple examples to illustrate intuitively the correlation r_{xy} between two random variables x and y. In each of the following cases, we first assume an experiment concerning two random variables x and y is carried out $K = 3$ times with different outcomes as listed in the tables, and then calculate their correlation r_{xy} based on the estimated means and covariances:

$$\hat{\mu}_x = \frac{1}{K}\sum_{k=1}^{K} x^{(k)}, \qquad \hat{\mu}_y = \frac{1}{K}\sum_{k=1}^{K} y^{(k)} \tag{8.13}$$

$$\hat{\sigma}_{xy}^2 = \frac{1}{K}\sum_{k=1}^{K} x^{(k)}y^{(k)} - \hat{\mu}_x\hat{\mu}_y, \qquad \hat{r}_{xy} = \frac{\hat{\sigma}_{xy}^2}{\sqrt{\hat{\sigma}_x^2\,\hat{\sigma}_y^2}} \tag{8.14}$$

1.

k	1st	2nd	3rd
$x^{(k)}$	1	2	3
$y^{(k)}$	1	2	3

(8.15)

We have $\hat{\mu}_x = \hat{\mu}_y = 2$, $\hat{\sigma}_{xy}^2 = \hat{\sigma}_x^2 = \hat{\sigma}_y^2 = 2/3$ and

$$\hat{r}_{xy} = \frac{\hat{\sigma}_{xy}^2}{\sqrt{\hat{\sigma}_x^2\,\hat{\sigma}_y^2}} = \frac{2/3}{\sqrt{(2/3)^2}} = 1 \tag{8.16}$$

i.e., x and y are maximally correlated (Fig.8.1(a)).

2.

k	1st	2nd	3rd
$x^{(k)}$	2	4	6
$y^{(k)}$	3	6	9

(8.17)

These values of x and y are the scaled version of the previous ones by 2 and 3, respectively, and we have:

$$\hat{\mu}_x = 4, \quad \hat{\mu}_y = 6, \quad \hat{\sigma}_x^2 = 8/3, \quad \hat{\sigma}_y^2 = 6, \quad \hat{\sigma}_{xy}^2 = 4 \tag{8.18}$$

and

$$\hat{r}_{xy} = \frac{\hat{\sigma}_{xy}^2}{\sqrt{\hat{\sigma}_x^2\,\hat{\sigma}_y^2}} = 1 \tag{8.19}$$

We see that the two variables x and y are scaled differently but their correlation, the normalized covariance, remains the same without being affected by the variable scaling.

3.

k	1st	2nd	3rd
$x^{(k)}$	1	2	3
$y^{(k)}$	3	2	1

(8.20)

Same as before, we have $\hat{\mu}_x = \hat{\mu}_y = 2$ and $\hat{\sigma}_x^2 = \hat{\sigma}_y^2 = 2/3$, but $\hat{\sigma}_{xy}^2 = -2/3$. Then correlation becomes $\hat{r}_{xy} = -1$, indicating that the two variables are negatively or inversely correlated (Fig.8.1(b)).

4.

k	1st	2nd	3rd
$x^{(k)}$	1	2	3
$y^{(k)}$	2	2	2

(8.21)

We have $\hat{\mu}_x = \hat{\mu}_y = 2$, $\hat{\sigma}_{xy}^2 = 0$, and $\hat{r}_{xy} = 0$, indicating that the two variables are completely uncorrelated (Fig.8.1(c)).

5.

k	1st	2nd	3rd
$x^{(k)}$	2	2	2
$y^{(k)}$	1	2	3

(8.22)

Same as before, the two variables are completely uncorrelated (Fig.8.1(d)).

6.

k	1st	2nd	3rd	4th	5th
$x^{(k)}$	1	2	2	2	3
$y^{(k)}$	2	1	2	3	2

(8.23)

This is the combination of the outcomes of the previous two cases when the experiment is repeated $K = 5$ times. Now we still have $\hat{\mu}_x = \hat{\mu}_y = 2$, $\hat{\sigma}_{xy}^2 = 0$ and $\hat{r}_{xy} = 0$, indicating that the two variables are completely uncorrelated.

In the examples above the variances σ_x^2 and σ_y^2 represent the dynamic energy or information contained in the two variables x and y, and the covariance σ_{xy}^2 represents how much the two variables x and y are similar or in common, i.e., how much they are correlated. If $\sigma_{xy}^2 > 0$, they are positively correlated, but if $\sigma_{xy}^2 < 0$ they are negatively correlated, and if $\sigma_{xy}^2 = 0$, they are not correlated at all. Specifically, in the first case above, the two variables x and y contain the same amount energy $\sigma_x^2 = \sigma_y^2$, and they are maximally correlated with $r_{xy} = 1$, i.e., the information they carry are redundant. On the other hand, in cases 4 and 5, $r_{xy} = 0$, indicating the two variables are not correlated at all, i.e., there is no redundancy among the two variables. Also, as variables y in case 4 and x in case 5 contain zero energy and therefore carry no information, they can be omitted to reduce data without losing any information. Comparing these two situations

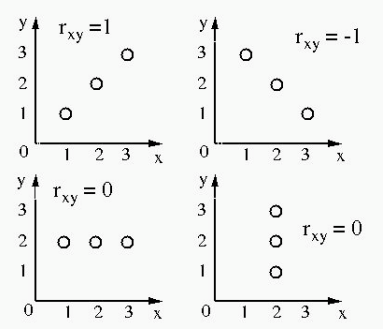

Figure 8.1 Different correlations between x and y

it becomes clear, from the signal processing point of view, that the latter cases 4 and 5 are much more preferred than the first one. In general we want to avoid high signal correlation and even energy distribution, it is therefore desirable to convert the given data in such a way that (1) the signal components are minimally correlated with little 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 information 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 in an optimal way.

8.2 Karhunen-Loeve theorem (KLT)

8.2.1 Continuous Karhunen-Loeve theorem (KLT)

As discussed previously, a deterministic time signal $x(t)$ can be represented by an orthogonal transform as a linear combination of a set of orthonormal basis functions:

$$\begin{cases} x(t) = \sum_k c_k \phi_k(t) \\ c_k = < x(t), \phi_k(t) > = \int x(t) \overline{\phi}_k(t) dt \end{cases} \tag{8.24}$$

Similarly, as a stochastic process, a random signal can also be represented in exactly the same form according to the Karhunen-Loeve theorem (Theorem 2.16). As discussed in section 2.5, the covariance $\sigma_x^2(t,\tau)$ of a centered stochastic process $x(t)$ is Hermitian kernel, and the associated integral operator is a self-adjoint and positive definite with real positive eigenvalues $\lambda_i > 0$ and orthogonal eigenfunctions $\phi_k(t)$:

$$< \phi_k(t), \phi_l(t) > = \int \phi_k(t) \overline{\phi}_l(t) dt = \delta[k-l] \tag{8.25}$$

Based on this result we obtained the Karhunen-Loeve theorem, which states that a stochastic process $x(t)$ can also be expressed as a linear combination of the

orthogonal basis functions $\phi_k(t)$ (Eq.2.322):

$$x(t) = \sum_{k=1}^{\infty} c_k \phi_k(t) \tag{8.26}$$

where the coefficients can be obtained as (Eq.2.323):

$$c_k =< x(t), \phi_k(t) >= \int x(t)\overline{\phi}_k(t)dt \tag{8.27}$$

These two equations for the series expansion of the stochastic signal $x(t)$ can also be considered as an orthogonal transform by which the time domain signal is converted into a set of coefficients c_k for the series in the transform domain. Although Eqs.8.26 and 8.27 appear to be identical to Eq.8.24, these two sets of equations are significantly different as the former is for deterministic signals while the latter is for random signals. The coefficients c_k in Eq.8.24 for a deterministic signal are constants, but the coefficients c_k in Eqs.8.26 and 8.27 are random variables. The random nature of a stochastic signal, when series expanded to become $x(t) = \sum_k c_k \phi_k(t)$, is reflected by the random coefficients c_k in the expansion. However, the orthogonal functions $\phi_k(t)$ of the expansion are deterministic, and they form a specific basis that spans the function space.

8.2.2 Discrete Karhunen-Loeve Transform

We now consider the discrete version of the KLT. When a stochastic process $x(t)$ is truncated and sampled, it becomes a random vector composed of N random variables $\boldsymbol{x} = [x[0], \cdots, x[N-1]]^T$. For convenience and without loss of generality, we will always assume in the following that the signal is centered with $\boldsymbol{\mu}_x = \mathbf{0}$, and its covariance matrix is $\boldsymbol{\Sigma}_x = E(\boldsymbol{x}\boldsymbol{x}^*)$ with its ij-th element being $\sigma_{ij}^2 = E(x[i]\overline{x}[j]) =< x[i], x[j] >$. As $\boldsymbol{\Sigma}_x$ is a positive definite Hermitian matrix its eigenvalues λ_i are real and positive, and its eigenvectors $\boldsymbol{\phi}_i$ form a set of orthogonal basis vectors that span the N-dimensional vector space. Any given N-D random vector in the space can therefore be represented as a linear combination of these basis vectors. This is the discrete KLT.

Let $\boldsymbol{\phi}_i$ $(i = 0, \cdots, N-1)$ be the eigenvector corresponding to the ith eigenvalue λ_i of the covariance matrix $\boldsymbol{\Sigma}_x$, i.e.,

$$\boldsymbol{\Sigma}_x \boldsymbol{\phi}_i = \lambda_i \boldsymbol{\phi}_i \qquad (i = 0, \cdots, N-1) \tag{8.28}$$

The matrix $\boldsymbol{\Phi} = [\boldsymbol{\phi}_0, \cdots, \boldsymbol{\phi}_{N-1}]$ formed by these N orthogonal eigenvectors is unitary, i.e., $\boldsymbol{\Phi}^{-1} = \boldsymbol{\Phi}^*$, or $\boldsymbol{\Phi}^*\boldsymbol{\Phi} = \boldsymbol{\Phi}\boldsymbol{\Phi}^* = \boldsymbol{I}$, and the N eigenequations in Eq.8.28 can be combined to become:

$$\boldsymbol{\Sigma}_x \boldsymbol{\Phi} = \boldsymbol{\Phi}\boldsymbol{\Lambda} \tag{8.29}$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix $\boldsymbol{\Lambda} = diag(\lambda_0, \cdots, \lambda_{N-1})$. If we pre-multiply $\boldsymbol{\Phi}^* = \boldsymbol{\Phi}^{-1}$ on both sides, the covariance matrix $\boldsymbol{\Sigma}_x$ is diagonalized:

$$\boldsymbol{\Phi}^*\boldsymbol{\Sigma}_x\boldsymbol{\Phi} = \boldsymbol{\Phi}^*\boldsymbol{\Phi}\boldsymbol{\Lambda} = \boldsymbol{\Lambda} \tag{8.30}$$

The discrete Karhunen-Loeve Transform of a given random signal vector $\boldsymbol{x}$ can now be defined as:

$$\boldsymbol{X} = \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \boldsymbol{\Phi}^* \boldsymbol{x} = \begin{bmatrix} \boldsymbol{\phi}_0^* \\ \vdots \\ \boldsymbol{\phi}_{N-1}^* \end{bmatrix} \boldsymbol{x} \tag{8.31}$$

where the ith component $X[i]$ of the vector $\boldsymbol{X}$ in transform domain is the projection of $\boldsymbol{x}$ onto the ith basis vector $\boldsymbol{\phi}_i$:

$$X[i] = \boldsymbol{\phi}_i^* \boldsymbol{x} = < \boldsymbol{x}, \boldsymbol{\phi}_i > \tag{8.32}$$

Pre-multiplying $\boldsymbol{\Phi}$ on both sides of equation 8.31, we get the inverse KLT transform:

$$\boldsymbol{x} = \boldsymbol{\Phi} \boldsymbol{X} = [\boldsymbol{\phi}_0, \cdots, \boldsymbol{\phi}_{N-1}] \begin{bmatrix} X[0] \\ \vdots \\ X[N-1] \end{bmatrix} = \sum_{i=0}^{N-1} X[i] \boldsymbol{\phi}_i \tag{8.33}$$

Eqs. 8.33 and 8.31 can be rewritten as a pair of the discrete KLT transform:

$$\begin{cases} \boldsymbol{X} = \boldsymbol{\Phi}^* \boldsymbol{x} \\ \boldsymbol{x} = \boldsymbol{\Phi}\ \boldsymbol{X} \end{cases} \tag{8.34}$$

The first equation is the forward transform that gives the random coefficient $X[i]$ as the projection of the random vector $\boldsymbol{x}$ onto the ith deterministic basis vector $\boldsymbol{\phi}_i$ $(i = 0, \cdots, N-1)$, while the second equation is the inverse transform that represents the random vector $\boldsymbol{x}$ as a linear combination of the N eigenvectors $\boldsymbol{\phi}_i$ $(i = 0, \cdots, N-1)$ of $\boldsymbol{\Sigma}_x$ weighted by the random coefficients $X[i]$. Note that Eqs. 8.33 and 8.31 for the discrete KLT correspond to Eqs.8.26 and 8.27 for the continuous KLT.

8.2.3 The Optimality of the KLT

As discussed previously, all orthogonal transforms considered in previous chapters (e.g., DFT, DCT, WHT, Haar transform, etc.) 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 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 given the value of a time sample $x[m]$ of the signal one could predict the value of the next sample $x[n+1]$ to be similar, this is certainly no longer the case in frequency domain in which knowing the value of one frequency component $X[n]$ would provide little information regarding the value of the next frequency component $X[n+1]$. Such tendencies are generally true for all other transform methods.

Now we will show that as far as the properties of signal decorrelation and energy compaction are concerned, the KLT is the optimal transform as

1. The KLT *completely* decorrelates any given signal and
2. The KLT *maximally* compacts the signal energy.

The first property is simply due to the definition of the KLT transform by which the covariance matrix $\mathbf{\Sigma}_X$ of the resulting vector $\boldsymbol{X} = \mathbf{\Phi}^* \boldsymbol{x}$ is diagonalized (Eq.8.30):

$$\mathbf{\Sigma}_X = E(\boldsymbol{X}\boldsymbol{X}^*) = E[(\mathbf{\Phi}^*\boldsymbol{x})(\mathbf{\Phi}^*\boldsymbol{x})^*] = \mathbf{\Phi}^* E(\boldsymbol{x}\boldsymbol{x}^*)\mathbf{\Phi} = \mathbf{\Phi}^* \mathbf{\Sigma}_x \mathbf{\Phi} = \mathbf{\Lambda} \tag{8.35}$$

After KLT the covariance $\mathbf{\Sigma}_X = \mathbf{\Lambda}$ becomes a diagonal matrix with all off-diagonal element $E(X[i]\overline{X}[j]) =< X[i], X[j] >= 0$, indicating that the correlation between any two different components $X[i]$ and $X[j]$ $(i \neq j)$ is indeed completely decorrelated, i.e., the signal components become orthogonal to each other. Also, the trace of the covariance matrix remains the same:

$$tr\mathbf{\Sigma}_X = tr(\mathbf{\Phi}^*\mathbf{\Sigma}_x\mathbf{\Phi}) = tr(\mathbf{\Phi}^*\mathbf{\Phi}\mathbf{\Sigma}_x) = tr\mathbf{\Sigma}_x \tag{8.36}$$

(recall $tr(\boldsymbol{A}\boldsymbol{B}) = tr(\boldsymbol{B}\boldsymbol{A})$), indicating that the total signal energy is conserved by the KLT:

$$\sum_{i=0}^{N-1} E(|x[i]|^2) = \sum_{i=0}^{N-1} E(|X[i]|^2) = \sum_{i=0}^{N-1} \lambda_i \tag{8.37}$$

This result corresponds to the Parseval's identity for the deterministic signals indicating that the signal energy is conserved by any orthogonal transform.

Next we prove the second property of the KLT, i.e., it redistributes the energy contained in all N signal components in such a way that the energy is optimally compacted into a minimum number of components. Let $\boldsymbol{A} = [\boldsymbol{a}_0, \cdots, \boldsymbol{a}_{N-1}] = (\boldsymbol{A}^*)^{-1}$ be an arbitrary unitary matrix, based on which an orthogonal transform can be defined as $\boldsymbol{X} = \boldsymbol{A}^*\boldsymbol{x}$, with the ith element of $\boldsymbol{X}$ being $X[i] = \boldsymbol{a}_i^*\boldsymbol{x}$. The energy contained in the first $M < N$ components after this transform is the sum of the first M elements along the main diagonal of $\mathbf{\Sigma}_X$:

$$\mathcal{E}_M(\boldsymbol{A}) = \sum_{i=0}^{M-1} E(|X[i]|^2) \tag{8.38}$$

We now prove the second property above by showing that $\mathcal{E}_M(\boldsymbol{A})$ is maximized if and only if the transform matrix is $\boldsymbol{A} = \mathbf{\Phi}$ for the KLT, i.e.,

$$\mathcal{E}_M(\mathbf{\Phi}) \geq \mathcal{E}_M(\boldsymbol{A}) \tag{8.39}$$

We first rewrite the expression for $\mathcal{E}_E(\boldsymbol{A})$ above as:

$$\begin{aligned}\mathcal{E}_M(\boldsymbol{A}) &= \sum_{i=0}^{M-1} E(|X[i]|^2) = \sum_{i=0}^{M-1} E(|\boldsymbol{a}_i^*\boldsymbol{x}|^2) = \sum_{i=0}^{M-1} E[(\boldsymbol{a}_i^*\boldsymbol{x})\ (\boldsymbol{x}^*\boldsymbol{a}_i)^*] \\ &= \sum_{i=0}^{M-1} E(\boldsymbol{a}_i^*\boldsymbol{x}\ \boldsymbol{x}^*\boldsymbol{a}_i) = \sum_{i=0}^{M-1} \boldsymbol{a}_i^* E(\boldsymbol{x}\boldsymbol{x}^*)\boldsymbol{a}_i = \sum_{i=0}^{M-1} \boldsymbol{a}_i^* \mathbf{\Sigma}_x \boldsymbol{a}_i\end{aligned}$$

The task of finding the optimal matrix $\boldsymbol{A}$ that maximizes $\mathcal{E}_M(\boldsymbol{A})$ can now be formulated as a constrained optimization problem:

$$\begin{aligned}&\mathcal{E}_M(\boldsymbol{A}) = \sum_{i=0}^{M-1} \boldsymbol{a}_i^* \boldsymbol{\Sigma}_x \boldsymbol{a}_i \rightarrow max \\ &\text{subject to:} \quad \boldsymbol{a}_i^* \boldsymbol{a}_i = 1 \quad (i = 0, \cdots, M-1)\end{aligned} \tag{8.40}$$

Here the constraint $\boldsymbol{a}_i^* \boldsymbol{a}_i = 1$ is to guarantee that $\boldsymbol{A}$ is indeed an orthogonal matrix with orthonormal column vectors. This problem can be solved using the method of Lagrange multipliers. Specifically, we set the following partial derivative with respect to $\boldsymbol{a}_j$ to zero and solve the resulting equation for $\boldsymbol{a}_j$ $(j = 0 \cdots, M-1)$:

$$\begin{aligned}&\frac{\partial}{\partial \boldsymbol{a}_j}[\mathcal{E}_M(\boldsymbol{A}) - \sum_{i=0}^{M-1} \lambda_i(\boldsymbol{a}_i^* \boldsymbol{a}_i - 1)] = \frac{\partial}{\partial \boldsymbol{a}_j}[\sum_{i=0}^{M-1} (\boldsymbol{a}_i^* \boldsymbol{\Sigma}_x \boldsymbol{a}_i - \lambda_i \boldsymbol{a}_i^* \boldsymbol{a}_i + \lambda_i)] \\ &= \frac{\partial}{\partial \boldsymbol{a}_j}[\boldsymbol{a}_j^* \boldsymbol{\Sigma}_x \boldsymbol{a}_j - \lambda_j \boldsymbol{a}_j^* \boldsymbol{a}_j] = 2\boldsymbol{\Sigma}_x \boldsymbol{a}_j - 2\lambda_j \boldsymbol{a}_j = 0\end{aligned} \tag{8.41}$$

The second to the last equal sign is due to the derivative of a scalar function $f(\boldsymbol{a})$ with respect to its vector argument $\boldsymbol{a}$, see appendix A). This result indicates $\boldsymbol{a}_j$ must be the eigenvector of $\boldsymbol{\Sigma}_x$:

$$\boldsymbol{\Sigma}_x \boldsymbol{a}_j = \lambda_j \boldsymbol{a}_j \qquad (j = 0, \cdots, M-1) \tag{8.42}$$

Comparing this to Eq.8.28, we see that $\boldsymbol{a}_j = \boldsymbol{\phi}_i$, i.e., we have thus proved that the optimal transform matrix is indeed the KLT matrix $\boldsymbol{\Phi}$. The energy contained in the M components is

$$\mathcal{E}_M(\boldsymbol{\Phi}) = \sum_{i=0}^{M-1} \boldsymbol{\phi}_i^* \boldsymbol{\Sigma}_x \boldsymbol{\phi}_i = \sum_{i=0}^{M-1} \lambda_i \tag{8.43}$$

where the ith eigenvalue λ_i represents the average energy contained in the ith component of the signal. We see that this $\mathcal{E}_M(\boldsymbol{\Phi})$ can be maximized if we choose the M eigenvectors $\boldsymbol{\phi}_i$ corresponding to the M largest eigenvalues. The percentage of energy kept in the M components is $\sum_{i=0}^{M-1} \lambda_i / \sum_{i=0}^{N-1} \lambda_i$.

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 in various applications such as information extraction and data compression. The signal components $X[i]$ 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 $\boldsymbol{\mu}_x$ of the given random signal vector $\boldsymbol{x}$. Subtract $\boldsymbol{\mu}_x$ from $\boldsymbol{x}$ so that it becomes centered with zero mean.
2. Estimate the covariance matrix $\boldsymbol{\Sigma}_x$ of the centered signal.

3. Find all N eigenvalues and sort then in descending order:

$$\lambda_0 \geq \cdots \geq \lambda_{N-1} \tag{8.44}$$

4. Determine a reduced dimensionality $M < N$ so that the percentage of energy contained $\sum_{i=0}^{M-1} \lambda_i / \sum_{i=0}^{N-1} \lambda_i$ 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 $\mathbf{\Sigma}_x$:

$$\mathbf{\Phi}_M = [\boldsymbol{\phi}_0, \cdots, \boldsymbol{\phi}_{M-1}]_{N \times M} \tag{8.45}$$

and carry out the KLT based on this $\mathbf{\Phi}_M$:

$$\boldsymbol{X}_M = \begin{bmatrix} X[0] \\ \vdots \\ X[M-1] \end{bmatrix}_{M \times 1} = \mathbf{\Phi}_M^* \boldsymbol{x} = \begin{bmatrix} \boldsymbol{\phi}_0^* \\ \vdots \\ \boldsymbol{\phi}_{M-1}^* \end{bmatrix}_{M \times N} \begin{bmatrix} x[0] \\ \vdots \\ x[N-1] \end{bmatrix}_{N \times 1} \tag{8.46}$$

where the kth element is $X[i] = \boldsymbol{\phi}_i^* \boldsymbol{x} =< \boldsymbol{x}, \boldsymbol{\phi}_i >$. As the dimensionality M of $\boldsymbol{X}$ is less than the dimensionality N of $\boldsymbol{x}$, data compression is achieved. This is a lossy compression with the error representing the percentage of information lost: $\sum_{i=M}^{N-1} \lambda_i / \sum_{i=0}^{N-1} \lambda_i$. But as these λ_i'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):

$$\hat{\boldsymbol{x}} = \mathbf{\Phi}_M \boldsymbol{X}_M = \mathbf{\Phi}_M \mathbf{\Phi}_M^* \boldsymbol{x} \tag{8.47}$$

or in component form:

$$\hat{\boldsymbol{x}} = \begin{bmatrix} \hat{x}[0] \\ \vdots \\ \hat{x}[N-1] \end{bmatrix} = \begin{bmatrix} \boldsymbol{\phi}_0 \cdots \boldsymbol{\phi}_{M-1} \end{bmatrix} \begin{bmatrix} X[0] \\ \vdots \\ X[M-1] \end{bmatrix} = \sum_{k=0}^{M-1} X[k] \boldsymbol{\phi}_k \tag{8.48}$$

$$= \begin{bmatrix} \boldsymbol{\phi}_0 \cdots \boldsymbol{\phi}_{M-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}_0^* \\ \vdots \\ \boldsymbol{\phi}_{M-1}^* \end{bmatrix} \boldsymbol{x} = \left[\sum_{k=0}^{M-1} \boldsymbol{\phi}_k \boldsymbol{\phi}_k^* \right]_{N \times N} \boldsymbol{x} \tag{8.49}$$

Here Eq.8.48 indicates that $\hat{\boldsymbol{x}}$ is a linear combination of the first M of the N eigenvectors that span the N-D space, while Eq.8.49 indicates that $\hat{\boldsymbol{x}}$ is a linear transformation of $\boldsymbol{x}$ by an $N \times N$ matrix formed as the sum of the M outer products $\boldsymbol{\phi}_k \boldsymbol{\phi}_k^*$ $(k = 0, \cdots, M-1)$. In particular when $M = N$, this matrix becomes $\mathbf{\Phi}_N \mathbf{\Phi}_N^* = \boldsymbol{I}_{N \times N}$ and $\hat{\boldsymbol{x}} = \boldsymbol{x}$ is a perfect reconstruction.

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 $\mathbf{\Phi} = [\boldsymbol{\phi}_0, \cdots, \boldsymbol{\phi}_{N-1}]$ is composed of the eigenvectors of the covariance matrix $\mathbf{\Sigma}_x$ of the signal $\boldsymbol{x}$, 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 many 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.

8.2.4 Geometric Interpretation of KLT

Assume the N random variables in a signal vector $\boldsymbol{x} = [x[0], \cdots, x[N-1]]^T$ have a normal joint probability density:

$$p(\boldsymbol{x}) = N(\boldsymbol{x}, \boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x) = \frac{1}{(2\pi)^{N/2}} |\boldsymbol{\Sigma}_x|^{1/2} exp[-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_x)^T \boldsymbol{\Sigma}_x^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_x)] \tag{8.50}$$

As always, we can assume $\boldsymbol{\mu}_x = \mathbf{0}$ without loss of generality. The shape of this normal distribution in the N-dimensional space can be represented by an iso-value hyper-surface in the space determined by:

$$N(\boldsymbol{x}, \boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x) = c \tag{8.51}$$

where the constant is chosen to be $c = (2\pi)^{-N/2} |\boldsymbol{\Sigma}_x|^{1/2} e^{-1/2}$ so that:

$$(\boldsymbol{x} - \boldsymbol{\mu}_x)^T \boldsymbol{\Sigma}_x^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_x) = \boldsymbol{x}^T \boldsymbol{\Sigma}_x^{-1} \boldsymbol{x} = 1 \tag{8.52}$$

In particular, when $N = 2$, this equation becomes:

$$\boldsymbol{x}^T \boldsymbol{\Sigma}_x^{-1} \boldsymbol{x} = [x[0], x[1]] \begin{bmatrix} a & b/2 \\ b/2 & c \end{bmatrix} \begin{bmatrix} x[0] \\ x[1]] \end{bmatrix} = ax^2[0] + bx[0]x[1] + cx^2[1] = 1 \tag{8.53}$$

where we have assumed:

$$\boldsymbol{\Sigma}_x^{-1} = \begin{bmatrix} a & b/2 \\ b/2 & c \end{bmatrix} \tag{8.54}$$

As $\boldsymbol{\Sigma}_x$ is positive definite and so is $\boldsymbol{\Sigma}_x^{-1}$, we have $\left|\boldsymbol{\Sigma}_x^{-1}\right| = ac - b^2/4 > 0$ and the above quadratic equation represents an ellipse (instead of other quadratic curves such as hyperbola or parabola) centered at the origin (or at $\boldsymbol{\mu}_x$ if it is not zero). In general when $N > 2$, the equation $N(\boldsymbol{x}, \boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x) = 1$ represents a hyper-ellipsoid in the N-D space. The spatial distribution of this ellipsoid is determined by $\boldsymbol{\Sigma}_x$. When $\boldsymbol{x}$ is completely decorrelated by KLT $\boldsymbol{X} = \boldsymbol{\Phi}^T \boldsymbol{x}$, the covariance matrix

becomes diagonalized:

$$\boldsymbol{\Sigma}_X = \boldsymbol{\Lambda} = \begin{bmatrix} \lambda_0 & 0 & \cdots & 0 \\ 0 & \lambda_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_{N-1} \end{bmatrix} = \begin{bmatrix} \sigma_X^2[0] & 0 & \cdots & 0 \\ 0 & \sigma_X^2[1] & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_X^2[N-1] \end{bmatrix} \tag{8.55}$$

and the quadratic equation becomes:

$$\boldsymbol{X}^T \boldsymbol{\Sigma}_X^{-1} \boldsymbol{X} = \sum_{i=0}^{N-1} \frac{X^2[i]}{\lambda_i} = \sum_{i=0}^{N-1} \left(\frac{X[i]}{\sigma_X[i]} \right)^2 \tag{8.56}$$

This equation represents a standard hyper-ellipsoid in the N-D space. In other words, the KLT transform $\boldsymbol{X} = \boldsymbol{\Phi}^* \boldsymbol{x}$ rotates the coordinate system in such a way that the semi-principal axes of the hyper-ellipsoid associated with the normal distribution of $\boldsymbol{x}$ are in parallel with $\boldsymbol{\phi}_i$ $(i = 0, \cdots, N-1)$, the basis vectors of the new coordinate system. Moreover, the ith semi-principal axis is equal to the square root of the corresponding eigenvalue $\sqrt{\lambda_i}$.

The 2-D KLT is illustrated in Fig.8.2. A given signal $\boldsymbol{x} = [x[0], x[1]]^T$ is originally represented under the standard basis vectors $\boldsymbol{e}_0$ and $\boldsymbol{e}_1$:

$$\boldsymbol{x} = \begin{bmatrix} x[0] \\ x[1] \end{bmatrix} = x[0]\boldsymbol{e}_0 + x[1]\boldsymbol{e}_1 = x[0] \begin{bmatrix} 1 \\ 0 \end{bmatrix} + x[2] \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{8.57}$$

We see that the two components $x[0]$ and $x[1]$ are maximally correlated with $r_{01} = 1$ and equal energy $\sigma_x^2[0] = \sigma_x^2[1]$, i.e., the energy is evenly distributed among both components. The KLT of the signal is a process of three stages: (1) subtract the mean $\boldsymbol{\mu}_x$ from $\boldsymbol{x}$ so that it is centered, (2) carry out the rotation $\boldsymbol{X} = \boldsymbol{\Phi}^* \boldsymbol{x}$, and (3) add back the mean vector in the rotated space $\boldsymbol{\mu}_X = \boldsymbol{\Phi}^* \boldsymbol{\mu}_x$. After the KLT rotation, the signal is represented as $\boldsymbol{X} = \boldsymbol{\Phi}^* \boldsymbol{x}$:

$$\boldsymbol{x} = \boldsymbol{\Phi} \boldsymbol{X} = [\boldsymbol{\phi}_0 \ \boldsymbol{\phi}_1] \begin{bmatrix} X[0] \\ X[1] \end{bmatrix} = X[0]\boldsymbol{\phi}_0 + X[1]\boldsymbol{\phi}_1 \tag{8.58}$$

Now the signal is represented by two new basis vectors $\boldsymbol{\phi}_0$ and $\boldsymbol{\phi}_1$, which are just rotated version of $\boldsymbol{e}_0$ and $\boldsymbol{e}_1$. In this space spanned by $\boldsymbol{\phi}_0$ and $\boldsymbol{\phi}_1$, the ellipse representing the joint probability density $p(\boldsymbol{x})$ becomes standardized with major semi-axis $\lambda_0 = \sigma_X^2[0]$ and minor semi-axes $\lambda_1 = \sigma_X^2[1]$, in parallel with the new basis vectors $\boldsymbol{\phi}_0$ and $\boldsymbol{\phi}_1$, respectively. We see that the two components $X[0]$ and $X[1]$ are completely decorrelated with $r_{01} = 0$, and $\lambda_0 > \lambda_1$ indicating that the energy is maximally compacted into $X[0]$ while $X[1]$ contains minimal energy. We see 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.

8.2.5 Comparison with Other Orthogonal Transforms

To illustrate the optimality of the KLT transform in terms of the two desirable properties of signal decorrelation and energy compaction discussed above, we

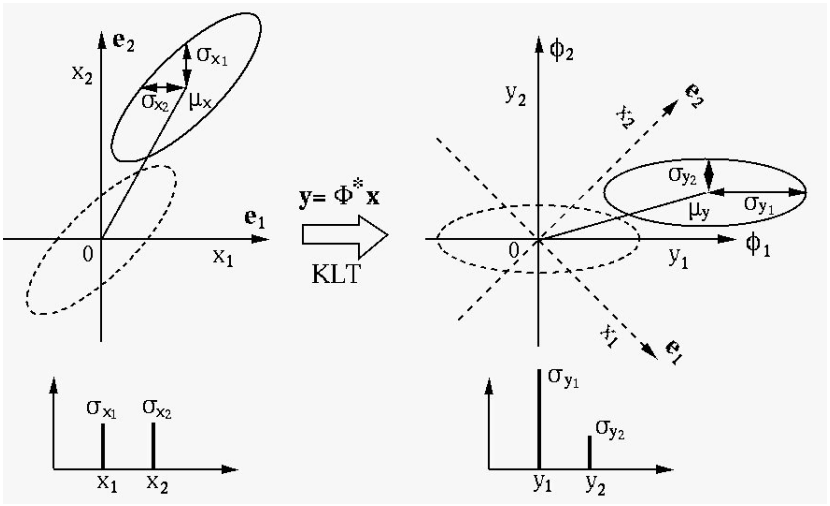

Figure 8.2 Geometric interpretation of KLT

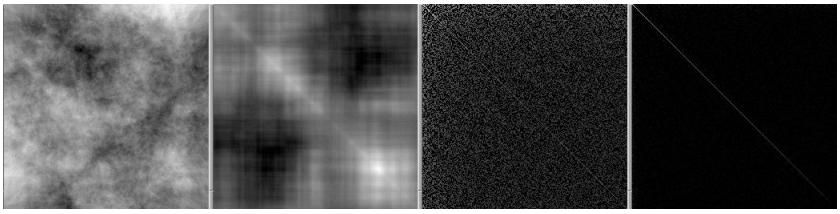

Figure 8.3 An image of clouds and covariance matrices after various transforms

compare the performance of the KLT with a set of other orthogonal transforms considered in previous chapters including identity transform (no transform), discrete Fourier transform DFT, discrete cosine transform DCT, and Walsh-Hadamard transform WHT, in the following examples.

Each row of a 256×256 image of clouds (left panel in Fig.8.3) is treated as an instantiation of a random vector $\boldsymbol{x}$ (with 256 components). Different orthogonal transforms $\boldsymbol{X} = \boldsymbol{A}^* \boldsymbol{x}$ are carried out and the corresponding covariance matrices $\boldsymbol{\Sigma}_X$ are obtained and compared to see how well each transform decorrelates the signal and compacts its energy. Fig.8.3 shows the original image (left panel) together with three covariance matrices corresponding to identify transform IT, DCT, and KLT. As the behaviors of DFT and WHT are very similar to that of DCT, they are not considered here. The pixel intensities of the images for covariance matrices are rescaled by a mapping $y = x^{0.3}$ so that those low values can still be seen.

In the second panel of Fig.8.3 showing the covariance matrix of the original signal without any transform, there exist quite a lot bright areas off the main diagonal, indicating that many signal components are highly correlated ($\sigma_{ij}^2 > 0$). In the third panel of Fig.8.3 showing the covariance matrix after a DCT, the

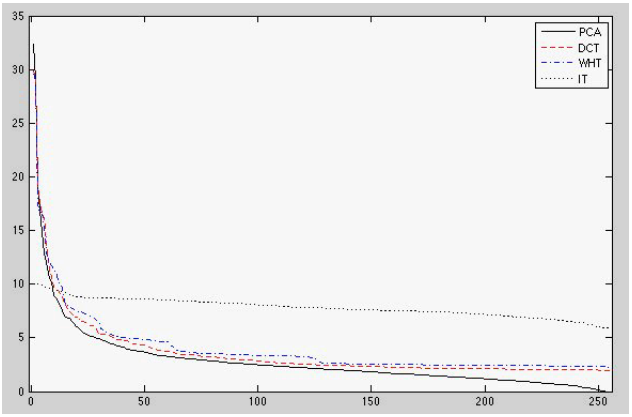

Figure 8.4 Signal energy distribution after various transforms

values of the off-diagonal elements are much reduced, indicating that the signal components are significantly decorrelated. Finally, in the last panel of the figure showing the covariance matrix after the KLT, all off-diagonal elements are 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 main diagonal, which is gradually reduced from top-left to bottom-right. This effect is more clearly shown in Fig.8.4 where the energy distribution among the N elements is plotted. The flat curve is the original energy distribution (no transform), indicating the energy is pretty evenly distributed without any transform. The remaining curves for energy distribution after the DCT, WHT and KLT all show some very steep descent (high on the left and low on the right), indicating that the signal energy is mostly concentrated in a small number of signal components. In particular, the curve with the steepest descent corresponds to the KLT with optimal energy compaction.

The effect of energy compaction is also illustrated by the table below, showing the number of components needed in order to keep certain percentage of the total signal energy (information) in data compression. For example, if it is tolerable to lose 5% of the signal energy/information, out of the total 256 components we need to keep 230 without any transform, 22 after DCT, but only 13 after KLT. In the optimal case of KLT, we can achieve a compression rate of $13/256 \approx 0.05$, i.e., 5% of the components contain keep 95% of the signal energy/information.

Percentage:	90%	95%	99%	100%
no transform:	209	230	250	256
DCT:	10	22	97	256
KLT:	7	13	55	256

Next we apply the same analysis process to a different image of sands, shown in the left panel of Fig.8.5. As the color of a grain of sand is irrelevant to that of

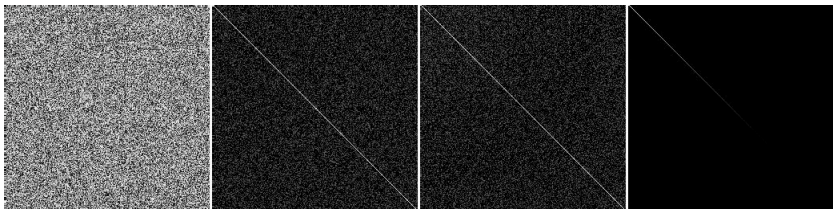

Figure 8.5 Image of sands and covariance matrices after various transforms

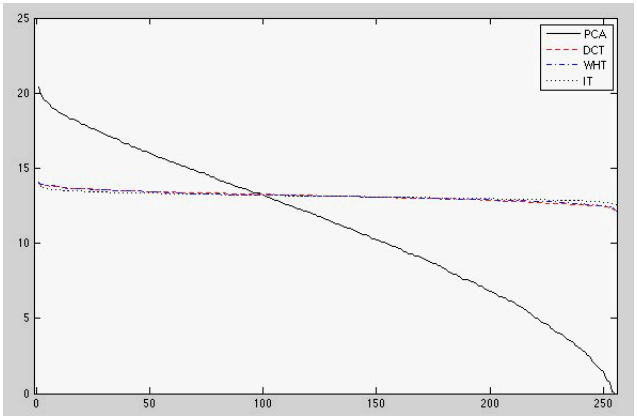

Figure 8.6 Signal energy distribution after various transforms

the neighboring grains, the texture of the sand is drastically different from that of the clouds in the previous case, and the pixel values are much less correlated in comparison to the pixels in the image of clouds. Again, the row vectors of the image are treated as different instantiations of a random vector and its covariance matrix is shown in the second panel. We see that all off-diagonal elements have very low values, indicating the pixels are hardly correlated. The signal correlation after the DCT is approximately the same, as indicated by signal covariance matrix after the DCT shown in the third panel. However, after the KLT the signal is again completely decorrelated as indicated by the diagonal covariance matrix shown in the last panel of the figure.

The energy distribution plots shown in Fig.8.6 indicate that DCT does not make much improvement in term of energy compaction, compared to the original signal (the two very similar flat plots), but KLT still maximally compact the energy, as shown by the curve high on the left low on the right.

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 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.

8.2.6 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 $\boldsymbol{R}$ (Eq.8.12). Specifically, we will show that when the correlation r of Markov process approaches $r = 1$, 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 $\boldsymbol{R}$, and (2), let $r \to 1$, and show that the KLT matrix approaches the DCT matrix.

The KLT matrix is the eigenvector matrix $\boldsymbol{\Phi}$ of the Toeplitz $\boldsymbol{R}$ which can be obtained by solving the eigenvalue problem:

$$\boldsymbol{R\Phi} = \boldsymbol{\Phi\Lambda}, \quad \text{i.e.,} \quad \boldsymbol{\Phi}^T\boldsymbol{R\Phi} = \boldsymbol{\Lambda} \tag{8.59}$$

As $\boldsymbol{R}$ is symmetric (self-adjoint), all λ_n are real and all $\boldsymbol{\phi}_n$ are orthogonal. It can be shown [1] that $\boldsymbol{\Phi}$ and $\boldsymbol{\Lambda}$ for a Toeplitz correlation matrix $\boldsymbol{R}$ take the following forms:

- The nth eigenvalue is:

$$\lambda_n = \frac{1-r}{1-2r\cos\omega_n + r^2}, \quad (n = 0, \cdots, N-1) \tag{8.60}$$

[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

- The mth element ϕ_{mn} of the nth eigenvector $\boldsymbol{\phi}_n = [\cdots, \phi_{mn}, \cdots]^T$ is:

$$\phi_{mn} = \left(\frac{2}{N+\lambda_n}\right)^{1/2} \sin\left(\omega_n(m - \frac{N-1}{2}) + (n+1)\frac{\pi}{2}\right), \quad (0 \le m, n \le N-1) \tag{8.61}$$

- In the above, ω_n $(n = 0, \cdots N-1)$ are the N real roots of the following equation:

$$\tan(N\omega) = -\frac{(1-r^2)\sin\omega}{(1+r^2)\cos\omega - 2r} \tag{8.62}$$

The proof for these expressions is lengthy and therefore omitted here.

Next we consider the three expressions given above when $r \to 1$. First, Eq.8.62 simply becomes:

$$\tan(N\omega) = \frac{0}{2(1-\cos\omega)} = 0 \tag{8.63}$$

Solving this for ω we get:

$$\omega_n = n\pi/N \tag{8.64}$$

However, note that when $n = 0$, $\omega_0 = 0$ and $\cos\omega_0 = 1$, and Eq.8.63 becomes an indeterminate form $0/0$. But applying L'Hopital's rule twice yields:

$$\lim_{\omega\to 0} \tan(N\omega) = \lim_{\omega\to 0} \frac{0}{2\cos\omega} = 0 \tag{8.65}$$

i.e., $\omega_0 = 0$ is still a valid root for Eq.8.62. Having found $\omega_n = n\pi/N$ for all $0 \le n \le N-1$, we can further find the eigenvalues λ_n in Eq.8.60 when $r \to 1$. For $n > 0$, $\omega_n \neq 0$ and $\cos\omega_n \neq 1$, we have:

$$\lambda_n = \lim_{r\to 1} \frac{1-r}{1-2r\cos\omega_n + r^2} = 0, \quad (1 \le n \le N-1) \tag{8.66}$$

We also get $\lambda_0 = N$ by noting that the second equation in Eq.8.59 is a similarity transformation of $\boldsymbol{R}$ which conserves its trace:

$$tr\boldsymbol{R} = N = tr\boldsymbol{\Lambda} = \sum_{n=0}^{N-1} \lambda_n = \lambda_0 \tag{8.67}$$

We can now find the elements ϕ_{mn} in the eigenvector $\boldsymbol{\phi}_n$. For all $n > 0$, we have $\lambda_n = 0$ and $\omega_n = n\pi/N$, Eq.8.61 becomes:

$$\begin{aligned}\phi_{mn} &= \sqrt{\frac{2}{N}} \sin\left(\frac{n\pi}{N}(m - \frac{N-1}{2}) + (n+1)\frac{\pi}{2}\right) = \sqrt{\frac{2}{N}} \sin\left(\frac{n\pi}{2N}(2m+1) + \frac{\pi}{2}\right) \\ &= \sqrt{\frac{2}{N}} \cos\left(\frac{n\pi}{2N}(2m+1)\right), \quad (0 \le m \le N-1, 1 \le n \le N-1)\end{aligned} \tag{8.68}$$

When $n = 0$, $\omega_0 = 0$ and $\lambda_0 = N$, and Eq.8.61 becomes:

$$\boldsymbol{\phi}_{m0} = \sqrt{\frac{1}{N}} \sin\left(\frac{\pi}{2}\right) = \sqrt{\frac{1}{N}}, \quad (0 \le m \le N-1) \tag{8.69}$$

This is the DCT transform matrix derived in section 6.2.3, and we can therefore conclude that the KLT of a first order Markov process approaches the DCT when $r \to 1$.

However, we also note that at the limit of $r = 1$, all elements of the correlation matrix $\boldsymbol{R}$ become 1 and its eigenvectors are no longer unique. While the column vectors of the DCT matrix are indeed its eigenvectors as shown above, so are the column vectors of any other orthogonal transform matrix $\boldsymbol{A}$ considered in the previous chapters (e.g., DFT, WHT, etc.), i.e.,

$$\boldsymbol{A}^T \boldsymbol{R} \boldsymbol{A} = \boldsymbol{\Lambda} = diag[N, 0, \cdots, 0] \tag{8.70}$$

where $\boldsymbol{A} = [\boldsymbol{a}_0, \cdots, \boldsymbol{a}_{N-1}]$ and the first column $\boldsymbol{a}_0$ is composed of N constant $1/\sqrt{N}$ (representing the DC component). Note that as all columns are orthogonal, all other columns $\boldsymbol{a}_n$ $(n \neq 0)$ sum up to zero:

$$< \boldsymbol{a}_n, \boldsymbol{a}_0 >= \boldsymbol{a}_n^T \boldsymbol{a}_0 = \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} a[m, n] = 0, \qquad (1 \leq n \neq N-1) \tag{8.71}$$

Now we see that the mnth element of the matrix in Eq.8.70 is zero:

$$\boldsymbol{a}_m^T \boldsymbol{R} \boldsymbol{a}_n = \boldsymbol{a}_m^T \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix} \boldsymbol{a}_n = 0, \qquad (m \neq 0, n \neq 0) \tag{8.72}$$

except when $m = n = 0$, the top-left element is

$$\boldsymbol{a}_0^T \boldsymbol{R} \boldsymbol{a}_0 = \frac{1}{N} [1, 1, \cdots, 1] \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} = N \tag{8.73}$$

As an example, Fig.8.7 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 vectors match the DCT very closely and the similarity will increase when r approaches 1. Also note that as the eigenvector of $\boldsymbol{R}$, a KLT vector $\boldsymbol{\phi}_n$ can have either a positive or negative sign, 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 be the same.

The result above has important significance. As most signals of interest in practice are likely to be locally correlated and can therefore 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 $\boldsymbol{\Sigma}_x$ corresponding to the eigenvalues arranged in descending order, they are actually arranged according the energy contained in signal components (represented by the eigenvalues).

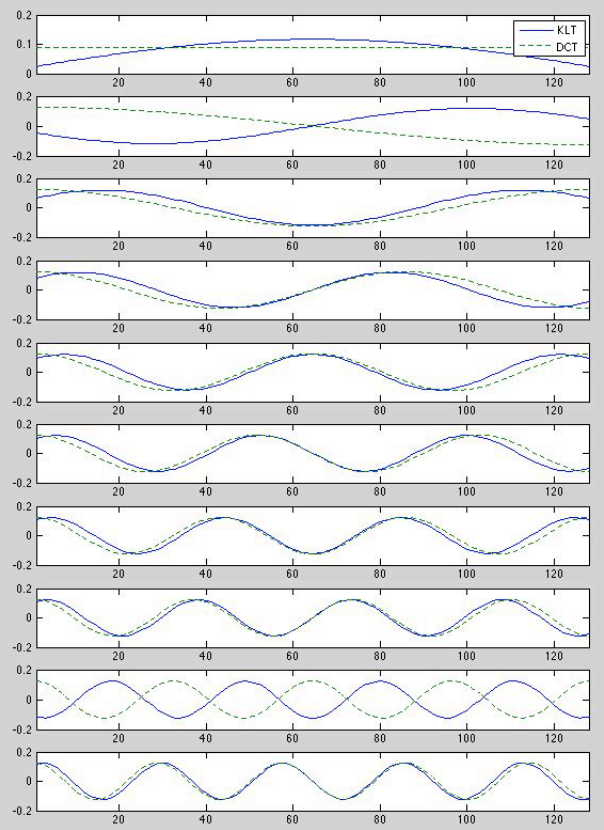

Figure 8.7 Comparison of the first 8 basis vectors of the DCT and KLT of 1st order Markov process

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 reconsider a dataset of annual temperatures in Los Angeles area collected over the period of 1878-1997, shown in the top panel of Fig.8.8. 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 main 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 main 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.8.9, 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.
- 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 $\boldsymbol{\phi}_0$ corresponding to the largest eigenvalue is approximated by the first row of the DCT matrix composed of all constants, representing the first principal component $y_0 =< \boldsymbol{x}, \boldsymbol{\phi}_0 >= \boldsymbol{\phi}_0^* \boldsymbol{x}$ is the average (DC component) of all elements in signal $\boldsymbol{x}$.

8.3 Applications of the KLT Transform

8.3.1 Image processing and analysis

The KLT can be carried out on a set of N $m \times n$ 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 vector is defined. We can treat each of the $K = m \times n$ pixels of the N images as a sample of an N-D random vector, the N images form a K by N matrix $\boldsymbol{D}$ each of whose row is for such an N-D random vector, whose covariance matrix can be estimated

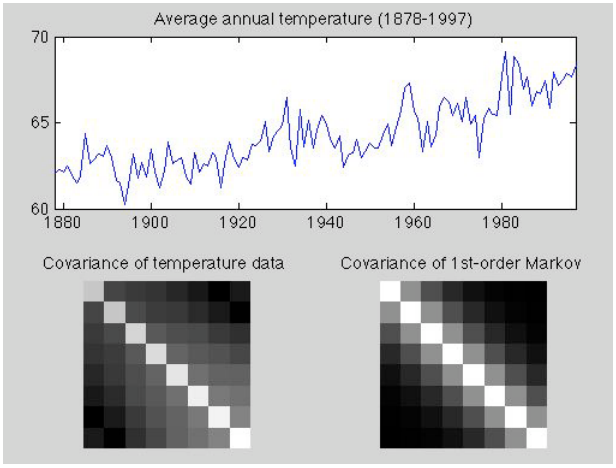

Figure 8.8 Covariances of natural signal and 1st order Markov chain

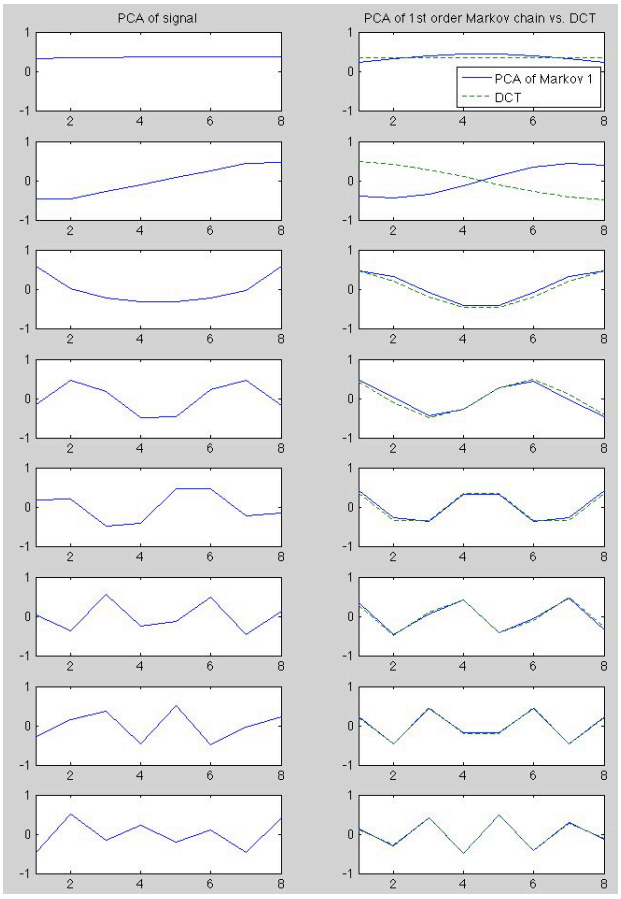

Figure 8.9 KLT of signal (left) compared with KLT of Markov model and DCT (right)

as (Eq.8.11):

$$\hat{\boldsymbol{\Sigma}}_x = \frac{1}{K}[\boldsymbol{D}^T\boldsymbol{D}]_{N\times N} \tag{8.74}$$

Alternatively, each image can be converted into a $K = m \times n$ dimensional vector by concatenating the rows (or columns). Each of these N vectors from the N images can be treated as a sample of a K-D random vector, represented by a columns of $\boldsymbol{D}$ defined above, or a row of $\boldsymbol{D}^T$, and the covariance matrix can be estimated as:

$$\hat{\boldsymbol{\Sigma}}_x = \frac{1}{N}[\boldsymbol{D}\boldsymbol{D}^T]_{K\times K} \tag{8.75}$$

We now show that the eigenvalue problems of these two different treatments are equivalent. We first assume the eigenequations for $\boldsymbol{D}^T\boldsymbol{D}$ and $\boldsymbol{D}^T\boldsymbol{D}$ are:

$$\boldsymbol{D}^T\boldsymbol{D}\boldsymbol{\phi} = \lambda\boldsymbol{\phi}, \qquad \boldsymbol{D}\boldsymbol{D}^T\boldsymbol{\psi} = \mu\boldsymbol{\psi} \tag{8.76}$$

Pre-multiplying $\boldsymbol{D}^T$ on both sides of the second equation we get:

$$\boldsymbol{D}^T\boldsymbol{D}[\boldsymbol{D}^T\boldsymbol{\psi}] = \mu[\boldsymbol{D}^T\boldsymbol{\psi}] \tag{8.77}$$

which is actually the same as the first eigenequation with the same eigenvalue $\mu = \lambda$, and the same eigenvector $\boldsymbol{D}^T\boldsymbol{\psi} = \boldsymbol{\phi}$ (when both sides are normalized). Although the dimensionalities of $\boldsymbol{D}^T\boldsymbol{D}$ and $\boldsymbol{D}^T\boldsymbol{D}$ are respectively N and K, the two matrices must have the same rank $min(N, K)$ and therefore the same number of non-zero eigenvalues. Consequently, the KLT can be carried out based on either $\boldsymbol{D}^T\boldsymbol{D}$ or $\boldsymbol{D}^T\boldsymbol{D}$ with essentially the same effects in terms of signal correlation and energy compaction. In the likely case where the number of image pixels is greater than the number of images, i.e., $K = mn > N$, we will take the first approach above to treat the same pixel (e.g., ith row and jth column) of the N images as a sample of the N-D random signal vector and carry out the KLT based on the N by N covariance matrix $\boldsymbol{\Sigma}_x = \boldsymbol{D}^T\boldsymbol{D}/K$. Each of the K pixels represented by a vector $\boldsymbol{x}$ is then transformed to a new vector $\boldsymbol{X} = \boldsymbol{\Phi}^*\boldsymbol{x}$ corresponding to the same pixel of a set of N *eigen-images*, as illustrated in Fig.8.10. Due to the nature of the KLT, most of the energy/information contained in the N images, representing the variations among the 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.

Consider, as an example, a sequence of N=8 frames from a video of a moving escalator shown on top of Fig.8.11. The covariance matrix and the energy distribution plot both before and after the KLT are shown in Fig.8.12. We see that due to the local correlation of the video frames, the covariance matrix (left) indeed closely resemble the correlation matrix $\boldsymbol{R}$ of a first order Markov process (bottom right in Fig.8.9) (consequently the KLT basis is very much similar to the DCT basis as shown in Eq.8.13). The covariance after the KLT (middle)

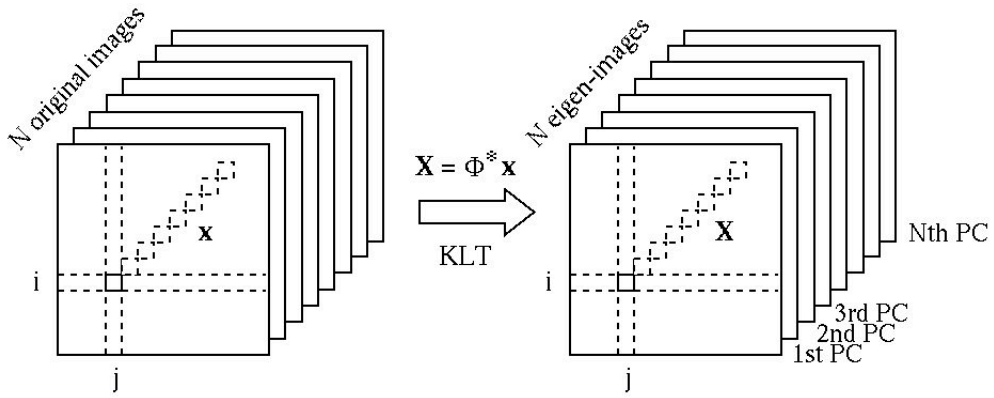

Figure 8.10 KLT of a set of images

is completely decorrelated and its energy highly compacted. The comparison of the energy distribution before and after the KLT is shown in the plots (right). Also, the eigen-images after the KLT are shown in the bottom half of Fig.8.11. It is interesting to observe that the first principal component represents mostly the static scene of the video frames, while the subsequent eigen-images represent mostly the motion in the video, the variation between the frames. For example, the motion of the people on the escalator is mostly reflected by the first few eigen-images following the first one, while the motion of the escalator steps is mostly reflected in the subsequent eigen-images.

As another example, consider a set of N=20 face images shown on top in Fig.8.14 (credit to AT&T Laboratories Cambridge). The KLT is carried out on these images to obtain the eigen-images, called in this case *eigen-faces* (middle). 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 the eigenfaces represent some other features with progressively less significance. The table below shows the percentage of energy contained in each component:

# of components	1	2	3	4	5	6	7	8	9	10
% energy	48.5	11.6	6.1	4.6	3.8	3.7	2.6	2.5	1.9	1.9
accumulative	48.5	60.1	66.2	70.8	74.6	78.3	81.0	83.5	85.4	87.3
	11	12	13	14	15	16	17	18	19	20
	1.8	1.6	1.5	1.4	1.3	1.2	1.1	1.1	0.9	0.8
	89.	90.7	92.2	93.6	94.9	96.1	97.2	98.2	99.2	100.0

Reconstructed faces using 95% of the total information are also shown in the figure. The method of eigenfaces is used in facial recognition and classification.

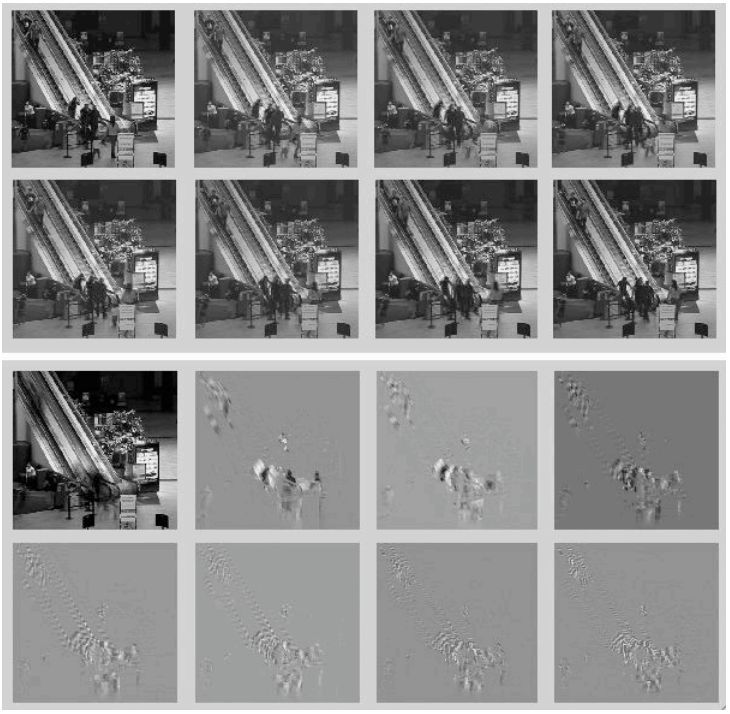

Figure 8.11 Video frames (top) and the eigen-images (bottom)

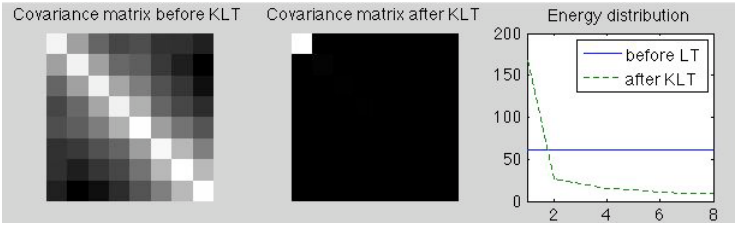

Figure 8.12 Covariance matrix before and after KLT
The covariance matrices before and after the KLT are shown as images (left and middle), while the energy distributions among the N components before and after the transform are also plotted (right).

8.3.2 Feature extraction for pattern recognition

In the field of machine learning, pattern recognition and classification 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 a pattern recognition/classification algorithm can be carried out in the N-D space, called *feature space*, in which all patterns reside. The classification

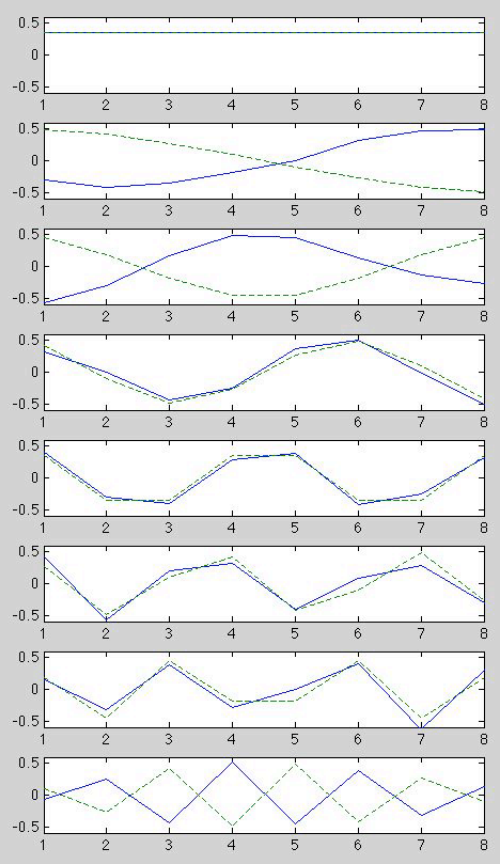

Figure 8.13 KLT basis vectors compared with DCT basis
The basis vectors of the KLT of the video frames closely resemble the DCT basis vectors.

is essentially the partitioning of the feature space into a set of regions each corresponding to one particular class. A given pattern is recognized as a member of 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 be obtained, an *unsupervised* algorithm has to be used. The KLT is an effective tool in the process called *feature extraction*, for extracting a set of pertinent features from the patterns to be classified.

For example, KLT is used in remote sensing, where the images of the surface of Earth or other planets such as Mars 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

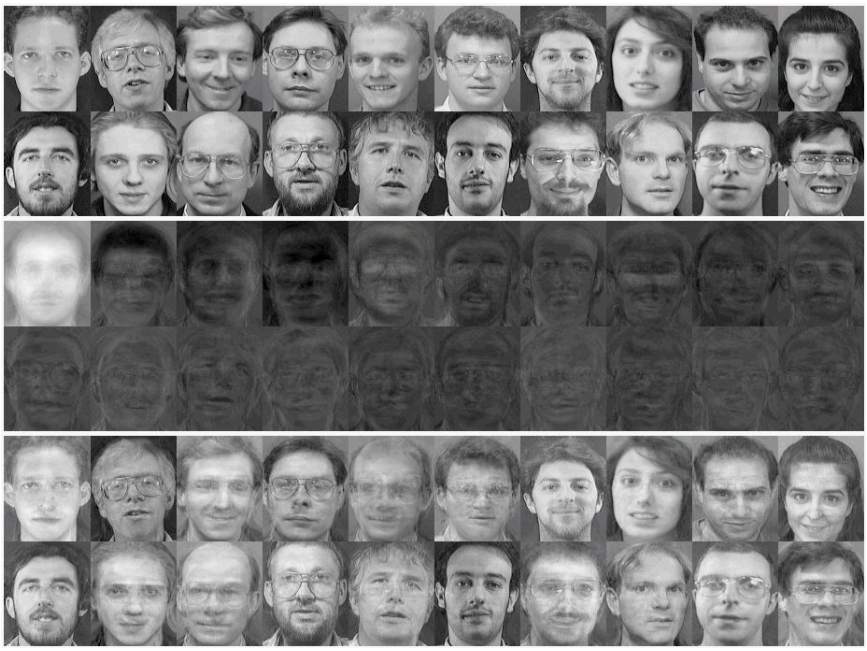

Figure 8.14 Original faces (top), Eigenfaces (middle), and Reconstructed faces (bottom)

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 hyper-spectral (N is up to 200 or greater). At each position in the image, a set of N pixel values each produced by one of the N sensors form an N-D vector in the feature space, representing the spectral signatures of the surface material. As different types of materials on the ground surface have different spectral signatures, a typical application of the multi or hyper-spectral image data is to classify all patterns in the N-D feature space, each for a pixel in the image, into different classes corresponding to different types of surface materials of interest. For hyper-spectral data with large N, the KLT can be used to reduce the dimensionality from N to $M << N$ without loss of essential information. Now the classification can be carried out in the M-D space corresponding to the first M eigen-images, thereby significantly reducing the computational complexity.

In many applications the patterns to be classified are given in image form, such as an object in an image. Extracting from the image a set of suitable features to represent the patterns may be a challenging task as it requires specific knowledge regarding the objects of interest. Alternatively, a more straightforward way of representing such image objects is simply to use all the pixels of the image. The shortcoming of this method, however, is that (1) the pixels are not particularly pertinent to the specific objects in the image, and (2) the dimensionality N,

the total number of pixels in the image, is typically unnecessarily high for the classification algorithm. In such case the KLT can be used.

Consider the feature extraction of a supervised classification problem for the recognition of hand-written characters, such as the 26 English letters or the 10 digits from 0 to 9, represented in image form by $N = 16 \times 16 = 256$ pixels. We assume a set of training samples with known classes is available based on which we need to come up with M features that represent the patterns of different classes most effectively. To so, we could use the KLT to extract a set of M¡N features from the N pixels, by first converting each image containing an object into an N-D vector by concatenating its rows (or columns) one after another, and then carrying out the KLT based on the covariance matrix of these vectors, independent of the classes of the objects they contain. However, we realize that the resulting features of this generic KLT are not specifically pertinent to the classification of the objects in the images, as they are extracted from all patterns without indiscriminating their classes.

Alternatively, to obtain M features most pertinent to the classification task by the KLT method, we can come up with a different covariance matrix containing the information directly reflecting the differences among the classes to be distinguished. We first let $\{\boldsymbol{x}_i^{(k)}, (i = 1, \cdots, n_k)\}$ be a set of n_k N-D vectors for the training samples of class k, where $k = 1, \cdots, K$ for all K classes. Based on these training samples we then define the following *scatter matrices*:

- *Scatter or covariance matrix* of class k for the variation or scatteredness within the class:

$$\boldsymbol{S}_k = \frac{1}{n_k} \sum_{i=1}^{n_k} (\boldsymbol{x}_i^{(k)} - \boldsymbol{m}_k)(\boldsymbol{x}_i^{(k)} - \boldsymbol{m}_k)^T, \qquad (k = 1, \cdots, K) \tag{8.78}$$

  where $\boldsymbol{m}_k$ is the mean vector of the kth class:

$$\boldsymbol{m}_k = \frac{1}{n_k} \sum_{i=1}^{n_k} \boldsymbol{x}_i^{(k)}, \qquad (k = 1, \cdots, K) \tag{8.79}$$

- *Within-class scatter matrix* for the average within-class scatteredness of all K classes:

$$\boldsymbol{S}_w = \sum_{k=1}^{K} p_k \boldsymbol{S}_k = \frac{1}{n} \sum_{k=1}^{K} n_k \boldsymbol{S}_k, \tag{8.80}$$

  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:

$$\boldsymbol{S}_b = \sum_{k=1}^{K} p_k (\boldsymbol{m}_k - \boldsymbol{m})(\boldsymbol{m}_k - \boldsymbol{m})^T, \tag{8.81}$$

where $\boldsymbol{m}$ is the mean vector of all n training samples of all K classes:

$$\boldsymbol{m} = \frac{1}{n}\sum_{\boldsymbol{x}} \boldsymbol{x} = \frac{1}{n}\sum_{k=1}^{k} n_k \frac{1}{n_k}\sum_{i=1}^{n_k} \boldsymbol{x}_i^{(k)} = \sum_{k=1}^{k} p_k \boldsymbol{m}_k \tag{8.82}$$

- *Total scatter or covariance matrix* for the total variation among all n samples of the K classes:

$$\begin{aligned}
\boldsymbol{S}_t &= \frac{1}{n}\sum_{\boldsymbol{x}} (\boldsymbol{x}-\boldsymbol{m})(\boldsymbol{x}-\boldsymbol{m})^T \\
&= \frac{1}{n}\sum_{k=1}^{K}\sum_{i=1}^{n_k} (\boldsymbol{x}_i^{(k)} - \boldsymbol{m}_k + \boldsymbol{m}_k - \boldsymbol{m})(\boldsymbol{x}^{(k)} - \boldsymbol{m}_k + \boldsymbol{m}_k - \boldsymbol{m})^T \\
&= \frac{1}{n}\sum_{k=1}^{K}\sum_{i=1}^{n_k} (\boldsymbol{x}_i^{(k)} - \boldsymbol{m}_k)(\boldsymbol{x}_i^{(k)} - \boldsymbol{m}_k)^T + \frac{1}{n}\sum_{k=1}^{K}\sum_{i=1}^{n_k} (\boldsymbol{m}_k - \boldsymbol{m})(\boldsymbol{m}_k - \boldsymbol{m})^T \\
&= \boldsymbol{S}_w + \boldsymbol{S}_b
\end{aligned} \tag{8.83}$$

The second to the last equal sign is due to the fact that

$$\sum_{k=1}^{K}\sum_{i=1}^{n_k} (\boldsymbol{x}_i^{(k)} - \boldsymbol{m}_k)(\boldsymbol{m}_k - \boldsymbol{m})^T = 0 \tag{8.84}$$

The equation $\boldsymbol{S}_t = \boldsymbol{S}_w + \boldsymbol{S}_b$ indicates that the total scatteredness $\boldsymbol{S}_t$ of the n samples is due to the contributions of the total within-class scatteredness $\boldsymbol{S}_w$ and the total between-class scatteredness $\boldsymbol{S}_b$, as one would intuitively expect. Now we can carry out the KLT based on the between-class scatter matrix $\boldsymbol{S}_b$, so that most of the energy or information specifically representing the separability of the K classes 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 $\boldsymbol{x}$ to the class with minimum distance $D(\boldsymbol{x}, \boldsymbol{m}_k)$ between its mean and the pattern $\boldsymbol{x}$:

$$\boldsymbol{x} \text{ belongs to class } k \text{ iff } \quad D(\boldsymbol{x}, \boldsymbol{m}_k) \le D(\boldsymbol{x}, \boldsymbol{m}_l), \qquad (l = 1, \cdots, K) \tag{8.85}$$

Moreover, it may be desirable to be able to visualize the data, for intuitive assessment of the distribution of the patterns in the feature space. However, visualization of 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 spatial distribution of the data points is conserved for visualization.

To illustrate the method, we now consider a specific example of classification of the 10 digits from 0 to 9, each written multiple times by different people, in the form of a 16 by 16 image, as shown in top-left panel of Fig.8.15. 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 $\boldsymbol{S}_b$ 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.8.15. 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 $\boldsymbol{S}_b$, 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 is then 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 the 9 components (Eq.8.47), as shown in 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.

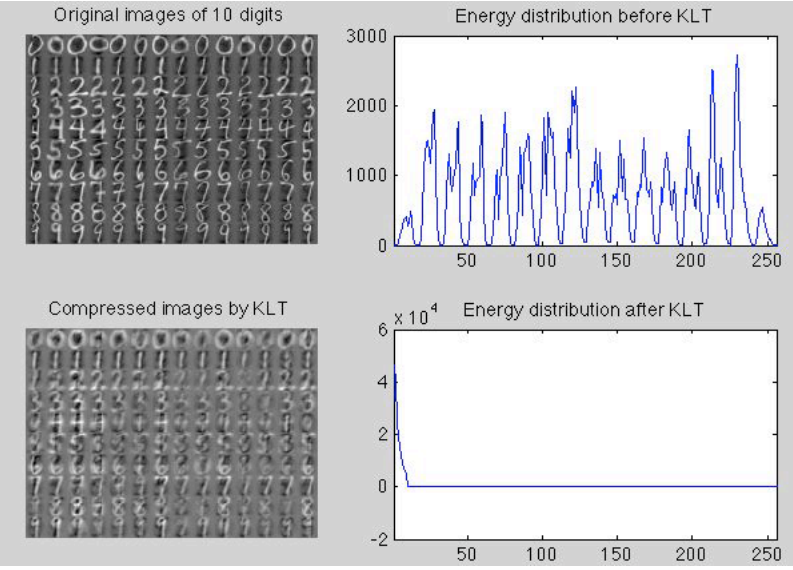

Figure 8.15 KLT of image pattern classification based on between-class scatter matrix

8.4 Singular Value Decomposition Transform

8.4.1 Singular Value Decomposition

The *singular value decomposition (SVD)* of an M by N matrix $\boldsymbol{A}$ of rank $R \leq min(M,N)$ is based on the following eigenvalue problems of an M by M symmetric matrix $\boldsymbol{A}\boldsymbol{A}^T$ and an N by N symmetric matrix $\boldsymbol{A}^T\boldsymbol{A}$:

$$\begin{aligned} \boldsymbol{A}\boldsymbol{A}^T\boldsymbol{u}_i &= \lambda_i \boldsymbol{u}_i, \qquad (i = 1, \cdots, M) \\ \boldsymbol{A}^T\boldsymbol{A}\boldsymbol{v}_i &= \lambda_i \boldsymbol{v}_i, \qquad (i = 1, \cdots, N) \end{aligned} \tag{8.86}$$

As both $\boldsymbol{A}\boldsymbol{A}^T$ and $\boldsymbol{A}^T\boldsymbol{A}$ are symmetric (self-adjoint), there eigenvalues λ_i are real and their eigenvector $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$ $(i = 1, \cdots, R)$ are orthogonal:

$$\boldsymbol{u}_i^T\boldsymbol{u}_j = \boldsymbol{v}_i^T\boldsymbol{v}_j = \delta[i-j] \tag{8.87}$$

and they form two orthogonal matrices $\boldsymbol{U} = [\boldsymbol{u}_1, \cdots, \boldsymbol{u}_N]_{M\times M}$ and $\boldsymbol{V} = [\boldsymbol{v}_1, \cdots, \boldsymbol{v}_N]_{N\times N}$ that satisfy:

$$\begin{aligned} \boldsymbol{U}\boldsymbol{U}^T &= \boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{I}_{M\times M} \\ \boldsymbol{V}\boldsymbol{V}^T &= \boldsymbol{V}^T\boldsymbol{V} = \boldsymbol{I}_{N\times N} \end{aligned} \tag{8.88}$$

The two $\boldsymbol{A}\boldsymbol{A}^T$ and $\boldsymbol{A}^T\boldsymbol{A}$ can be diagonalized by $\boldsymbol{U}$ and $\boldsymbol{V}$ respectively:

$$\begin{aligned} \boldsymbol{U}^T(\boldsymbol{A}\boldsymbol{A}^T)\boldsymbol{U} &= \boldsymbol{\Lambda}_{M\times M} = diag[\lambda_1, \cdots, \lambda_R, 0, \cdots, 0] \\ \boldsymbol{V}^T(\boldsymbol{A}^T\boldsymbol{A})\boldsymbol{V} &= \boldsymbol{\Lambda}_{N\times N} = diag[\lambda_1, \cdots, \lambda_R, 0, \cdots, 0] \end{aligned} \tag{8.89}$$

Note that as the rank of $\boldsymbol{A}$ is R, there exist only R non-zero eigenvalues.

The theorem of singular value decomposition states that any M by N matrix $\boldsymbol{A}$ can be diagonalized by $\boldsymbol{U}$ and $\boldsymbol{V}$:

$$\boldsymbol{U}^T\boldsymbol{A}V = \boldsymbol{\Lambda}^{1/2} = diag[\sqrt{\lambda_1}, \cdots, \sqrt{\lambda_R}, 0, \cdots, 0] = diag[s_1, \cdots, s_R, 0, \cdots, 0] \tag{8.90}$$

Here $s_i = \sqrt{\lambda_i}$ $(i = 1, \cdots, R)$ is defined as the ith *singular value* of matrix $\boldsymbol{A}$. This equation can be considered as the forward SVD transform. By pre-multiplying $\boldsymbol{U}$ and post-multiplying $\boldsymbol{V}^T$ on both sides of the equation above, we get inverse transform:

$$\boldsymbol{A} = \boldsymbol{U}\Lambda^{1/2}\boldsymbol{V}^T = \sum_{k=1}^{R} \sqrt{\lambda_i}[\boldsymbol{u}_k\boldsymbol{v}_k^T] = \sum_{k=1}^{R} s_k[\boldsymbol{u}_k\boldsymbol{v}_k^T] \tag{8.91}$$

by which represents the original matrix $\boldsymbol{A}$ is decomposed into a linear combination of R matrices $[\boldsymbol{u}_i\boldsymbol{v}_i^T]$ weighted by $\sqrt{\lambda_i}$ $(i = 1, \cdots, R)$. We can rewrite both the forward and inverse SVD transform as a pair:

$$\begin{cases} \boldsymbol{\Lambda}^{1/2} = \boldsymbol{U}^T\boldsymbol{A}V \\ \boldsymbol{A} = \boldsymbol{U}\Lambda^{1/2}\boldsymbol{V}^T \end{cases} \tag{8.92}$$

The matrix $\boldsymbol{A}$ can be considered as a 2-D signal, such as an image, which can be forward SVD transformed to obtain a set of coefficients $s_i = \sqrt{\lambda_i}$ for the

components $[\boldsymbol{u}_i \boldsymbol{v}_i^T]$ $(i = 1, \cdots, R)$, which can also be called eigen-images, and 2-D signal $\boldsymbol{A}$ 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 the M by N matrix $\boldsymbol{A}$ is simply the sum of the energy contained in each of its $M \times N$ elements a_{ij}, which is equal to the trace of either $\boldsymbol{A}\boldsymbol{A}^T$ and $\boldsymbol{A}^T\boldsymbol{A}$:

$$\mathcal{E} = \sum_{i=1}^{M}\sum_{j=1}^{N} |a_{ij}|^2 = tr(\boldsymbol{A}\boldsymbol{A}^T) = tr(\boldsymbol{A}^T\boldsymbol{A}) \tag{8.93}$$

Moreover, as trace is conserved by an orthogonal transform, we take trace on both sides of Eq.8.89 to get:

$$\begin{aligned} tr[\boldsymbol{U}^T(\boldsymbol{A}\boldsymbol{A}^T)\boldsymbol{U}] = tr(\boldsymbol{A}\boldsymbol{A}^T) = tr\boldsymbol{\Lambda} = \sum_{i=1}^{R}\lambda_i \\ tr[\boldsymbol{V}^T(\boldsymbol{A}^T\boldsymbol{A})\boldsymbol{V}] = tr(\boldsymbol{A}^T\boldsymbol{A}) = tr\boldsymbol{\Lambda} = \sum_{i=1}^{R}\lambda_i \end{aligned} \tag{8.94}$$

This result indicates that the energy contained in the signal $\boldsymbol{A}$ 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 no information is lost or generated, i.e., the signal information is conserved. If, for simplicity, we assume $M = N = R$, then the DOF of $\boldsymbol{A}_{N\times N}$ before the transform is N^2. In the transform domain, the signal is represented in terms of $\boldsymbol{U}$, $\boldsymbol{V}$, and $\boldsymbol{\lambda}$. The DOF of both $\boldsymbol{U}$ and $\boldsymbol{V}$ is $(N^2 - N)/2$ for the following reason. 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 $\boldsymbol{U}$ is:

$$(N-1) + (N-2) + \cdots + 1 = N(N-1)/2 = (N^2 - N)/2 \tag{8.95}$$

The same is true for $\boldsymbol{V}$. 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 $\boldsymbol{A}$ before the SVD transform.

8.4.2 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 image matrix $\boldsymbol{A}_{N\times N} = [\boldsymbol{a}_1, \cdots, \boldsymbol{a}_N]$ where $\boldsymbol{a}_i$ is the ith

column vector of $\boldsymbol{A}$. Image compression can be achieved by using only the first K eigen-images of $\boldsymbol{A}$:

$$\boldsymbol{A}_K = \sum_{i=1}^{K} \sqrt{\lambda_i} \boldsymbol{u}_i \boldsymbol{v}_i^T \tag{8.96}$$

The energy contained in $\boldsymbol{A}_k$ is:

$$\begin{aligned} tr\ [\boldsymbol{A}_K^T \boldsymbol{A}_K] &= tr[\sum_{i=1}^{K} \sqrt{\lambda_i} \boldsymbol{v}_i \boldsymbol{u}_i^T][\sum_{j=1}^{K} \sqrt{\lambda_j} \boldsymbol{u}_j \boldsymbol{v}_j^T] \\ &= tr[\sum_{i=1}^{K} (\sum_{j=1}^{K} \sqrt{\lambda_i}\sqrt{\lambda_j} \boldsymbol{v}_i \boldsymbol{u}_i^T \boldsymbol{u}_j \boldsymbol{v}_j^T)] = tr[\sum_{i=1}^{K} \lambda_i \boldsymbol{v}_i \boldsymbol{v}_i^T] \\ &= \sum_{i=1}^{K} \lambda_i tr\ [\boldsymbol{v}_i \boldsymbol{v}_i^T] = \sum_{i=1}^{K} \lambda_i \boldsymbol{v}_i^T \boldsymbol{v}_i = \sum_{i=1}^{K} \lambda_i \end{aligned}$$

The percentage of energy contained in the compressed image $\boldsymbol{A}_K$ is:

$$\sum_{i=1}^{K} \lambda_i / \sum_{i=1}^{R} \lambda_i \tag{8.97}$$

Obviously if we use the K components corresponding to the K largest eigenvalues, the energy contained in $\boldsymbol{A}_K$ is maximized.

Next we consider the compression rate in terms of the DOF of $\boldsymbol{A}_K$. The DOF in the K orthogonal vectors $\{\boldsymbol{u}_i \ \ i = 1, \cdots, K\}$ is:

$$(N-1) + (N-2) + \cdots + (N-K) = NK - K(K+1)/2 \tag{8.98}$$

The same is true for $\{\boldsymbol{v}_i \ \ i = 1, \cdots, K\}$. Including the DOF of K in $\{\lambda_i,\ i = 1, \cdots, K\}$, we get the total DOF:

$$2NK - K(K+1) + k = 2NK - K^2 \tag{8.99}$$

and the compression ratio is

$$\frac{2NK - K^2}{N^2} = \frac{2K}{N} - \frac{K^2}{N^2} \approx \frac{2K}{N} \tag{8.100}$$

We consider a specific example of the image of Lenna ($M = N = R = 128$) shown in Fig.8.16 (left) together with its SVD matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ (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.8.17. The reconstructed images using different number K of the SVD eigen-images are shown in Fig.8.18, where the top two rows show the SVD eigen-images (1st row) corresponding to the first (largest) 10 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 ($K = 10, 20, 30, \cdots$).

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.8.17. 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 8.16 Original image (left), matrices U (middle) and V (right)

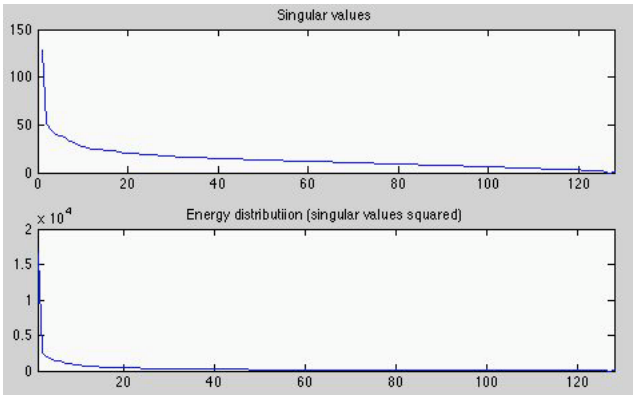

Figure 8.17 Singular values $s_i = \sqrt{\lambda_i}$ (top) and their energy distribution λ_i (bottom)

Figure 8.18 SVD components (top) and the corresponding partial reconstructions (bottom)

9 Continuous and Discrete-time Wavelet Transforms

9.1 Why Wavelet?

9.1.1 Short-time Fourier transform and Gabor transform

A time signal $x(t)$ contains the complete information in time domain, i.e., the amplitude of the signal at any given moment t. However, no information is explicitly available in $x(t)$ in terms of its frequency contents. On the other hand, the spectrum $X(f) = \mathcal{F}[x(t)]$ of the signal obtained by the Fourier transform (or any other orthogonal transform such as discrete cosine transform) is extracted from the entire time duration of the signal, it contains complete information in frequency domain in terms of the magnitudes and phases of the frequency component at any given frequency f, but there is no information explicitly available in the spectrum regarding the temporal characteristics of the signal, such as when in time certain frequency contents appear. 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.

To address this dilemma, the short-time Fourier transform (STFT), also called windowed Fourier transform, can be used. The signal $x(t)$ to be Fourier analyzed is first truncated by a time window function before it is transformed to the frequency domain. Now any characteristics appearing in the spectrum will be known to be from within this particular time window.

Let us first consider using a simple rectangular window with width T to truncate the signal:

$$w_r(t) = \begin{cases} 1 \; 0 < t < T \\ 0 \text{ otherwise} \end{cases} \tag{9.1}$$

If a particular segment $\tau < t < \tau + T$ of the signal $x(t)$ is of interest, the signal is first truncated by multiplication of the window $w_r(t)$ shifted by τ, and then Fourier transformed:

$$X_r(f,\tau) = \mathcal{F}[x(t)w_r(t-\tau)] = \int_{-\infty}^{\infty} x(t)w_r(t-\tau)e^{-j2\pi ft}dt = \int_{\tau}^{\tau+T} x(t)e^{-j2\pi ft}dt \tag{9.2}$$

Based on the time-shift and frequency convolution properties of the Fourier transform, the spectrum of this windowed signal can also be expressed as:

$$X_r(f,\tau) = X(f) * [W_r(f)e^{-2\pi f\tau}] \tag{9.3}$$

where $W_r(f) = \mathcal{F}[w_r(t)]$ is the Fourier transform of the rectangular window $w_r(t)$, a sinc function. While certain temporal locality can be achieved in the transform domain, the STFT spectrum $X_r(f)$ in frequency domain is severely distorted due to the convolution with the ringing sinc function $W_r(f) = \mathcal{F}[w_r(t)]$.

In order to overcome this drawback, a smooth window such as a bell-shaped Gaussian function can be used:

$$w_g(t) = e^{-\pi(t/\sigma)^2} \tag{9.4}$$

where the parameter σ controls the width of the window. Again the signal is first multiplied by a shifted version of the Gaussian window $w_g(t-\tau)$, and then Fourier transformed to get:

$$X_g(f,\tau) = \mathcal{F}[x(t)w_g(t-\tau)] = \int_{-\infty}^{\infty} x(t)e^{-(t-\tau)^2/\sigma^2}e^{-j2\pi ft}dt \tag{9.5}$$

This Fourier transform of the Gaussian windowed signal is called the *Gabor transform* of the signal, from which 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 the inverse transform:

$$\begin{aligned}
&\int_{-\infty}^{\infty} X_g(f,\tau)e^{j2\pi f\tau}df = \int_{-\infty}^{\infty}[\int_{-\infty}^{\infty} x(t)e^{-(t-\tau)^2/\sigma^2}e^{-j2\pi ft}dt\ ]\ e^{j2\pi f\tau}df \\
&= \int_{-\infty}^{\infty} x(t)e^{-(t-\tau)^2/\sigma^2}[\int_{-\infty}^{\infty} e^{-j2\pi ft}e^{j2\pi f\tau}df\ ]\ dt = \int_{-\infty}^{\infty} x(t)e^{-(t-\tau)^2/\sigma^2}\delta(t-\tau)dt \\
&= x(\tau)
\end{aligned} \tag{9.6}$$

Similar to the case of rectangular windowing in Eq.9.3, due to the frequency shift and the convolution properties of the Fourier transform, the Gabor spectrum in Eq.9.5 can also be written as:

$$X_g(f,\tau) = [W_g(f)e^{-j2\pi f\tau}] * X(f) \tag{9.7}$$

Different from the rectangular windowing, here the Fourier transform of the Gaussian window is also a Gaussian function (Eq. 3.149):

$$W_g(f) = \mathcal{F}[w_g(t)] = \sigma e^{-\pi(\sigma f)^2} \tag{9.8}$$

i.e., the spectrum $X(f)$ is now convolved with a smooth function $W_g(f)$, instead of a ringing sinc function $W_r(f)$ as in the previous case, therefore the Gabor spectrum $X_g(f)$ will not be distorted as severely as in the previous case.

9.1.2 The Heisenberg Uncertainty

The STFT methods in general, either with rectangular or Gaussian windowing, suffer from a more profound difficulty, namely, an increased time resolution will necessarily result in a decreased frequency resolution. The frequency resolution of STFT spectrum, either $X_r(f,\tau)$ or $X_g(f,\tau)$, is a blurred version of the true Fourier spectrum $X(f)$, due to the convolution in Eqs.9.3 or 9.7. For example, in the case of the Gabor transform, as the width of $W_g(f)$ $(1/\sigma)$ in frequency domain is inversely proportional to the width of $w_g(t)$ (σ) 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 be more clearly illustrated if we further assume the truncated signal $x(t)$ 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 discrete frequency components $X[k]$ $(k=0,\pm1,\pm2,\cdots)$, with a gap of $f_0=1/T$ between any two consecutive components $X[k]$ and $X[k+1]$. Obviously this discrete spectrum provides no information in the gaps. 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.

Another drawback the general STFT approach also suffers from is that the window width is fixed through out the analysis, independent of the specific signal being analyzed, while there may be a whole variety of different characteristics of varying time scales in the signal. For example, the signal may contain some random and sparse spikes, and bursts of rapid oscillation, which can be localized only if the time window used is very narrow. On the other hand, there may be some totally different features in the signal, such as some slow changing drifts and trends, which can be captured only if the time window has much wider width. Therefore it would be very difficult for the STFT method to detect and represent at the same time all such signal characteristics that are potentially of great importance and interest.

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. This effect is referred to as the *Heisenberg uncertainty*, as previously discussed in Chapter3 and clearly demonstrated by Eq.3.165.

If the temporal features of interest in a signal do not change much over time, i.e, the signal is stationary, then the Fourier transform is 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. In such cases, Fourier analysis is unable to detect them in terms of when such events has taken place, therefor unsuitable to describe or represent such signal features which may be of most interest.

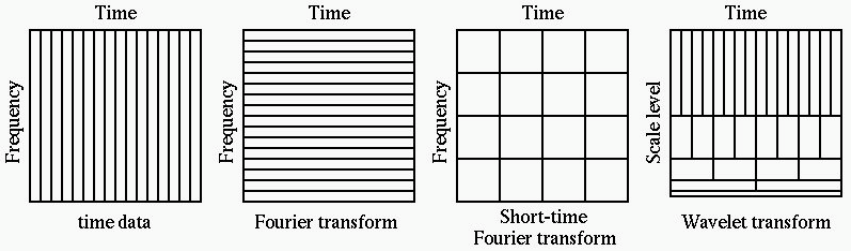

Figure 9.1 Comparisons of temporal and frequency locality in Fourier and wavelet transforms

In order to overcome this limitation 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. The wavelet transform 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. Different scale levels from lowest to the highest can reveal events of different scales.

The discussion above can be summarized by the *Heisenberg Box (Cell)* shown in Fig.9.1, which illustrates the issue of resolution or locality in both time and frequency in the Fourier transform, short-time Fourier transform and wavelet transform.

The first figure is the time signal with full time resolution but zero frequency resolution. The second is its Fourier spectrum with full frequency resolution but zero frequency resolution. The third one is the short-time Fourier transform with a fixed window size and inversely proportional resolutions in time and frequency domains. The last one is for the wavelet transform with varying scale levels and their corresponding time resolution, i.e., at a low scale level (less details corresponding to low frequencies) the window size a large, while at a high scale level (more 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.

9.2 Continuous-Time Wavelet Transform (CTWT)

9.2.1 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 $x(t)$ with a transform kernel function $\phi_f(t)$:

$$X(f) =< x(t), \phi_f(t) >= \int x(t)\overline{\phi}_f(t)dt \tag{9.9}$$

where the kernel function $\phi_f(t)$ represents the fth member of a family of complete basis functions that span the space in which the signal $x(t)$ exist. For example, in the case of Fourier transform, the kernel function is a complex exponential $\phi_f(t) = e^{j2\pi ft}$ with a parameter f representing a specific frequency. Similarly, the *continuous wavelet transform (CWT)* is also an integral transform based on a set of kernel functions, sometimes referred to as the *daughter wavelets*, all derived from a *mother wavelet* $\psi(t)$ that satisfies the following conditions:

- $\psi(t)$ should have a compact support, i.e., $\psi(t) \neq 0$ only inside a bounded range $a < t < b$.
- $\psi(t)$ has zero mean or zero DC component:

$$\int_{-\infty}^{\infty} \psi(t)dt = 0, \quad \text{i.e.} \quad \Psi(f)\big|_{f=0} = \Psi(0) = 0 \tag{9.10}$$

  where $\Psi(f) = \mathcal{F}[\psi(t)]$ is the Fourier transform of $\psi(t)$. In other words, the DC component of the mother wavelet is zero. This condition is needed in the future discussion of wavelet transforms.
- $\psi(t) \in \mathcal{L}^2$ is square integrable:

$$\int_{-\infty}^{\infty} |\psi(t)|^2 dt < \infty \tag{9.11}$$

- $\psi(t)$ can be normalized (same as all orthogonal transforms with normalized basis vectors):

$$||\psi(t)||^2 = \int_{-\infty}^{\infty} |\psi(t)|^2 dt = 1 \tag{9.12}$$

Qualitatively, a mother wavelet $\psi(t)$ has two properties. 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 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:

$$\psi_{s,\tau}(t) = \frac{1}{\sqrt{s}}\psi(\frac{t-\tau}{s}) \tag{9.13}$$

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 $\phi_f(t) = e^{j2\pi ft}$ of the Fourier transform with only one

parameter f for frequency, the CWT 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:

$$\begin{aligned} ||\psi_{s,\tau}(t)||^2 &= \int_{-\infty}^{\infty} |\psi_{s,\tau}(t)|^2 dt = \frac{1}{s}\int_{-\infty}^{\infty} |\psi_{s,\tau}(\frac{t-\tau}{s})|^2 dt \\ &= \int_{-\infty}^{\infty} \psi_{s,\tau}^2(t') s dt' = \int_{-\infty}^{\infty} \psi^2(t) dt = ||\psi(t)||^2 = 1 \end{aligned} \tag{9.14}$$

Here we have assumed $t' = (t-\tau)/s$ and therefore $dt' = dt/s$.

If we obtain the Fourier spectrum of the mother wavelet $\Psi(f) = \mathcal{F}[\psi(t)]$, the Fourier spectrum of each kernel function $\psi_{s,\tau}(t)$ can be found according to the time-shift and scaling properties of the Fourier transform: (Eqs.3.103, 3.102):

$$\Psi_{s,\tau}(f) = \mathcal{F}[\psi_{s,\tau}(t)] = \mathcal{F}[\frac{1}{\sqrt{s}}\psi(\frac{t-\tau}{s})] = \sqrt{s}\Psi(sf)e^{-j2\pi f\tau} \tag{9.15}$$

9.2.2 The forward and inverse wavelet transforms

First we consider the forward continuous wavelet transform. Given a mother wavelet $\psi(t)$ and all her daughter wavelets $\psi_{s,\tau}(t)$ for different s and τ, we can define the continuous wavelet transform (CWT) $X(s,\tau)$ of a time signal $x(t)$ as an integral transform: [1]

$$\begin{aligned} X(s,\tau) = \mathcal{W}[x(t)] &=< x(t), \psi_{s,\tau}(t) >= \int_{-\infty}^{\infty} x(t)\overline{\psi}_{s,\tau}(t)dt \\ &= \frac{1}{\sqrt{s}}\int_{-\infty}^{\infty} x(t)\overline{\psi}(\frac{t-\tau}{s})dt = x(\tau) \star \psi_{s,0}(\tau) \end{aligned} \tag{9.16}$$

As indicated by the last equality, the wavelet transform of $x(t)$ can also be considered as the correlation of the signal $x(t)$ and a wavelet function $\psi_{s,0}(t) = \psi(t/s)/\sqrt{s}$. Therefore, according to property of the correlation (Eq.3.108), the Fourier spectrum of the CWT $X(s,\tau)$ can be obtained in frequency domain as a multiplication:

$$\hat{X}(s,f) = \mathcal{F}[X(s,\tau)] = X(f)\,\overline{\Psi}_{s,0}(f), \tag{9.17}$$

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. Also, according to Eq.9.15, we

[1] Various notations have been used for the continuous wavelet transform of a signal $x(t)$, such $CWT_x(s,\tau)$ and $Wx(s,\tau)$. However, in this chapter a notation $X(s,\tau)$ is used in consistence with $X(f) = \mathcal{F}[x(t)]$ for the Fourier transform of $x(t)$.

have

$$\Psi_{s,0}(f) = \Psi_{s,\tau}(f)\big|_{\tau=0} = \sqrt{s}\Psi(sf)e^{-j2\pi f\tau}\big|_{\tau=0} = \sqrt{s}\Psi(sf) \tag{9.18}$$

Now the CWT can be readily found as the inverse Fourier transform of $\hat{X}(s,f)$:

$$X(s,\tau) = \mathcal{F}^{-1}[\hat{X}(s,f)] = \mathcal{F}^{-1}[X(f)\,\overline{\Psi}_{s,0}(f)] \tag{9.19}$$

Next we consider the inverse wavelet transform (ICWT) by which the original time function $x(t)$ can be reconstructed based on its CWT $X(s,\tau)$:

$$\begin{aligned} x(t) = \mathcal{W}^{-1}[X(s,\tau)] &= \frac{1}{C_\psi}\int_0^\infty \int_{-\infty}^\infty X(s,\tau)\psi_{s,\tau}(t)d\tau\ \frac{ds}{s^2} \\ &= \frac{1}{C_\psi}\int_0^\infty \frac{1}{\sqrt{s}}\int_{-\infty}^\infty X(s,\tau)\psi(\frac{t-\tau}{s})d\tau\ \frac{ds}{s^2} \end{aligned} \tag{9.20}$$

where C_ψ is defined as

$$C_\psi = \int_0^\infty \frac{|\Psi(f)|^2}{f}df < \infty \tag{9.21}$$

Eq.9.21 is referred to as the *admissibility condition*, which is necessary for the inverse CWT to exist. Note that in order for this condition to hold, we must have $\Psi(f)_{f=0} = \Psi(0) = 0$, i.e., the wavelet $\psi(t)$ has zero mean, as one of the conditions specified before (Eq.9.10). Consequently, Eq.9.19 will produce the same result for different $X(0)$ values (as it is always multiplied by $\Psi(0) = 0$). In other words, CWT is insensitive to the DC component $X(0)$ of the signal $x(t)$.

Now we prove the inverse CWT given in Eq.9.20. Multiplying $\Psi_{s,0}(f)$ on both sides of Eq.9.17, we get

$$\hat{X}(s,f)\Psi_{s,0}(f) = X(f)\,\overline{\Psi}_{s,0}(f)\Psi_{s,0}(f) = X(f)\,|\Psi_{s,0}(f)|^2 \tag{9.22}$$

and then divide both sides by s^2 and integrate with respect to s:

$$\int_0^\infty \hat{X}(s,f)\Psi_{s,0}(f)\frac{ds}{s^2} = X(f)\int_0^\infty |\Psi_{s,0}(f)|^2\frac{ds}{s^2} = X(f)\int_0^\infty \frac{|\Psi(sf)|^2}{s}ds \tag{9.23}$$

The last equal sign is due to Eq.9.18. We further consider the integral on the right-hand side:

$$\int_0^\infty \frac{|\Psi(sf)|^2}{s}ds = \int_0^\infty \frac{|\Psi(sf)|^2}{sf}d(sf) = \int_0^\infty \frac{|\Psi(f')|^2}{f'}df' = C_\psi \tag{9.24}$$

where we have assumed $f' = sf$, and the last equal sign is due to the definition of C_ψ in Eq.9.21. Now Eq.9.23 can be written as:

$$X(f) = \frac{1}{C_\psi}\int_0^\infty \hat{X}(s,f)\ \Psi_{s,0}(f)\frac{ds}{s^2} \tag{9.25}$$

Now the time signal can be obtained by taking the inverse Fourier transform on both sides:

$$\begin{aligned} x(t) = \mathcal{F}^{-1}[X(f)] &= \frac{1}{C_\psi}\int_0^\infty \mathcal{F}^{-1}[\hat{X}(s,f)\ \Psi_{s,0}(f)]\frac{ds}{s^2} \\ &= \frac{1}{C_\psi}\int_0^\infty X(s,t) * \psi_{s,0}(t)\frac{ds}{s^2} \\ &= \frac{1}{C_\psi}\int_0^\infty \frac{1}{\sqrt{s}}\int_{-\infty}^\infty X(s,\tau)\psi(\frac{t-\tau}{s})d\tau\frac{ds}{s^2} \end{aligned} \tag{9.26}$$

Here we have used the convolution theorem of the Fourier transform. The inverse wavelet transform given in Eq.9.20 is thereby proven.

As a side product, we note that the result of Eq.9.24 indicates a very interesting fact. For any given function $f(x)$, the integral of its scaled version $f(sx)/s$ over all scale s is a constant, over the entire domain of x. This result has some important significance, as we will see later in the discussion of the discrete-time wavelet transform.

In summary, the continuous wavelet transform in Eq.9.16 converts a 1-D continuous time signal $x(t)$ into a 2-D function $X(s,\tau)$ of two arguments s for scale and τ for translation, and by the inverse transform in Eq.9.20, the signal can be reconstructed based on its CWT coefficients $X(s,\tau)$. Although this pair of forward CWT and inverse CWT is similar to all previously discussed orthogonal transforms such as the Fourier transform, there are some essential differences. Most obviously, the Fourier spectrum $X(f)$ of a time signal $x(t)$ is a 1-D function of frequency f, but the CWT coefficient $X(s,\tau)$ is a 2-D function of two variables s and τ. Also, unlike the Fourier transform or any othter transforms, the CWT is not an orthogonal transform, as its kernel functions $\psi_{s,\tau}(t)$ are not orthogonal to each other, i.e., they do not form an orthonormal basis of the function space. We will consider in more details in terms of such differences between the CWT and other orthogonal transforms.

9.3 Properties of CTWT

- Linearity:

$$\mathcal{W}[ax(t) + by(t)] = a\mathcal{W}[x(t)] + b\mathcal{W}[y(t)] \tag{9.27}$$

  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. 9.16). Therefore due to the linearity of the inner product in the first variable, the wavelet transform is also linear.
- Time shift: If $\mathcal{W}[x(t)] = X(s,\tau)$, then $\mathcal{W}[x(t-t')] = X(s,\tau-t')$.

$$\mathcal{W}[x(t-t')] = \frac{1}{\sqrt{s}}\int_{-\infty}^\infty x(t-t')\overline{\psi}(\frac{t-\tau}{s})dt \tag{9.28}$$

Let $u = t - t'$, i.e., $t = u + t'$ and $dt = du$, the above becomes:

$$\mathcal{W}[x(t-t')] = \frac{1}{\sqrt{s}} \int_{-\infty}^{\infty} x(u)\overline{\psi}(\frac{u-(\tau-t')}{s})du = X(s, \tau - t') \tag{9.29}$$

- Time scaling: If $\mathcal{W}[x(t)] = X(s,\tau)$, then $\mathcal{W}[x(t/a)/\sqrt{a}] = X(s/a, \tau/a)$.

$$\mathcal{W}[x(t/a)/\sqrt{a}] = \frac{1}{\sqrt{as}} \int_{-\infty}^{\infty} x(t/a)\overline{\psi}(\frac{t-\tau}{s})dt \tag{9.30}$$

Let $u = t/a$, i.e., $t = au$ and $dt = adu$, the above becomes:

$$\begin{aligned}\mathcal{W}[x(t/a)/\sqrt{a}] &= \frac{a}{\sqrt{as}} \int_{-\infty}^{\infty} x(u)\overline{\psi}(\frac{au-\tau)}{s})du \\ &= \frac{1}{\sqrt{s/a}} \int_{-\infty}^{\infty} x(u)\overline{\psi}(\frac{u-\tau/a)}{s/a})du = X(s/a, \tau/a)\end{aligned} \tag{9.31}$$

- Localization Property:
  Consider the scaling and translation of a mother wavelet $\psi(t)$ in frequency domain as well in time domain. Assume its center is at $t = t_0$ and its width is Δt, and the center and width of its Fourier transform $\Psi(f)$ are f_0 and Δf. Then the center of the scaled and translated wavelet function $\psi_{s,\tau}$ (Eq. 9.13) is at $t = at_0 + \tau$ and its width is

$$\Delta t_{s,\tau} = s\Delta t \tag{9.32}$$

The Fourier transform of $\psi_{s,\tau}(t)$ is $\Psi_{s,\tau}(f) = \sqrt{s}\Psi(sf)e^{-j2\pi f\tau}$ and its center $f_{s,\tau}$ and width $\Delta f_{s,\tau}$ are:

$$\begin{aligned} f_{s,\tau} &= \frac{1}{2} f_0 \\ \Delta f_{s,\tau} &= \frac{1}{s}\Delta f \end{aligned} \tag{9.33}$$

We can now make two observations:

  - The product of the window widths of the wavelet function $\psi_{s,\tau}(t)$ in time and frequency domains is constant, independent of s and τ.

$$\Delta t_{s,\tau}\Delta f_{s,\tau} = s\Delta t\frac{1}{s}\Delta f = \Delta t\Delta f \tag{9.34}$$

  - If we consider a wavelet function $\psi_{s,\tau}(t)$ as a band-pass filter $\Psi_{s,\tau}(f)$ with a quality factor Q defined as the ratio of its bandwidth and the center frequency, then we have

$$Q = \frac{\Delta f}{f_0} \tag{9.35}$$

i.e., the quality factor Q of the filter is constant, independent of the scaling factor s.

- Multiplication theorem:

Let $X_1(f) = \mathcal{F}[x_1(t)]$ and $X_2(f) = \mathcal{F}[x_2(t)]$ be the Fourier transforms of $x_1(t)$ and $x_2(t)$, respectively. Then, according to Eq.9.19, we have:

$$X_i(s,\tau) = \sqrt{s}\int_{-\infty}^{\infty} X_i(f), \overline{\Psi}(sf)e^{j2\pi f\tau}df, \qquad (i=1,2) \tag{9.36}$$

We substitute these expressions into the inner product of $X_1(s,\tau)$ and $X_2(s,\tau)$ defined as:

$$\begin{aligned}
&< X_1(s,\tau), X_2(s,\tau) >= \int_0^{\infty}\int_{-\infty}^{\infty} X_1(s,\tau)\overline{X}_2(s,\tau)\, d\tau\, \frac{ds}{s^2} \\
&= \int_0^{\infty}\int_{-\infty}^{\infty}[\int_{-\infty}^{\infty} X_1(f)\sqrt{s}\overline{\Psi}(sf)e^{j2\pi f\tau}df]\,[\int_{-\infty}^{\infty}\overline{X}_2(f')\sqrt{s}\Psi(sf')e^{-j2\pi f'\tau}df']\, d\tau\, \frac{ds}{s^2} \\
&= \int_0^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}[X_1(f)\overline{X}_2(f')s\overline{\Psi}(sf)\Psi(sf')\int_{-\infty}^{\infty} e^{j2\pi(f-f')\tau}d\tau]\, df'\, df\, \frac{ds}{s^2} \\
&= \int_0^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} X_1(f)\overline{X}_2(f')s\overline{\Psi}(sf)\Psi(sf')\delta(f-f')df'\, df\, \frac{ds}{s^2} \\
&= \int_{-\infty}^{\infty} X_1(f)\overline{X}_2(f)[\int_0^{\infty}\frac{|\Psi(sf)|^2}{s}ds]\, df = C_\psi\int_{-\infty}^{\infty} X_1(f)\overline{X}_2(f)\, df
\end{aligned} \tag{9.37}$$

Note that the integral with respect to s inside the brackets is C_ψ. Now due to the multiplication theorem of the Fourier transform $< x_1(t), x_2(t) >=< X_1(f), X_2(f) >$, we have:

$$< X_1(s,\tau), X_2(s,\tau) >= C_\psi\int_{-\infty}^{\infty} X_1(f)\overline{X}_2(f)\, df = C_\psi < x_1(t), x_2(t) > \tag{9.38}$$

This is the multiplication theorem of the CWT. In particular, when $x_1(t) = x_2(t) = x(t)$, we have $< X(s,\tau), X(s,\tau) >= C_\psi < x(t), x(t) >$, i.e.,

$$\int_{-\infty}^{\infty}|x(t)|^2dt = \frac{1}{C_\psi}\int_{-\infty}^{\infty}|X(s,\tau)|^2d\tau\frac{ds}{s^2} \tag{9.39}$$

This is Parseval's identity (energy conservation) for the CWT.

- Non-orthogonality

  All previously considered orthogonal transforms, such as the Fourier transform, convert a 1-D time signal $x(t)$ into another 1-D function in transform domain, such as the spectrum $X(f)$ in frequency domain. All basis functions that span the vector space containing the signal $x(t)$ are orthogonal

  $$< \phi_f(t), \phi_{f'}(t) >= 0, \qquad (f \neq f') \tag{9.40}$$

  indicating that they are completely uncorrelated and there exists no redundancy among these basis functions. In other words, every single point f in the transform domain makes its unique contribution to the reconstruction of the time signal by the inverse transform.

  However, this is no longer the case for the continuous wavelet transform, which converts a 1-D time signal $x(t)$ to a 2-D function $X(s,\tau)$ defined over a half plane $-\infty < \tau < \infty$ and $s > 0$. Consequently, there exists a large amount of

redundancy in the 2-D transform domain in terms of the information needed for 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 corresponding kernel functions (basis functions) $\psi_{s,\tau}(t)$ and $\psi_{s',\tau'}(t)$:

$$K(s, \tau, s', \tau') =< \psi_{s,\tau}(t), \psi_{s',\tau'}(t) >= \int_{-\infty}^{\infty} \psi_{s,\tau}(t)\overline{\psi}_{s',\tau'}(t)dt \neq 0 \qquad (9.41)$$

The fact that this inner product is not zero indicates a major difference between the CWT and all orthogonal transforms considered before, i.e., the CWT is not an orthogonal transform anymore. This reproducing kernel can be considered as the correlation between the two kernel functions $\psi_{s,\tau}(t)$ and $\psi_{s',\tau'}(t)$, representing the redundancy between them.
On the other hand, if $X(s, \tau)$ is a CWT coefficient of a signal $x(t)$, $X(s', \tau')$ at any other point in the (s, τ) plane may not also be a CWT coefficient of $x(t)$ unless it is a linear combination of all those true coefficients, each weighted by the corresponding reproducing kernel $K(s, \tau, s', \tau')$:

$$X(s', \tau') = \frac{1}{C_\psi} \int_0^{\infty} \int_{-\infty}^{\infty} K(s, \tau, s', \tau')X(s, \tau)d\tau \frac{ds}{s^2} \qquad (9.42)$$

This identity can be easily proved. For $X(s', \tau')$ to be a CWT coefficient of $x(t)$, it has to satisfy the CWT definition:

$$X(s', \tau') =< x(t), \psi_{s',\tau'}(t) >= \int_{-\infty}^{\infty} x(t)\overline{\psi}_{s',\tau'}(t)dt \qquad (9.43)$$

Substituting the reconstructed $x(t)$ in Eq.9.20 into this equation, we get:

$$\begin{aligned} X(s', \tau') &= \int_{-\infty}^{\infty} \left[\frac{1}{C_\psi} \int_0^{\infty} \int_{-\infty}^{\infty} X(s, \tau)d\tau \, \frac{ds}{s^2} \right] \psi_{s,\tau}(t)\overline{\psi}_{s',\tau'}(t)dt \\ &= \frac{1}{C_\psi} \int_0^{\infty} \int_{-\infty}^{\infty} X(s, \tau) \left[\int_{-\infty}^{\infty} \psi_{s,\tau}(t)\overline{\psi}_{s',\tau'}(t)dt \right] d\tau \, \frac{ds}{s^2} \\ &= \frac{1}{C_\psi} \int_0^{\infty} \int_{-\infty}^{\infty} K(s, \tau, s', \tau')X(s, \tau)d\tau \frac{ds}{s^2} \end{aligned} \qquad (9.44)$$

9.4 Typical Mother Wavelet Functions

In the discussion of the wavelet transform, the waveform of the mother wavelet function is not specifically defined. Here we consider some commonly used mother wavelets.

- **Shannon wavelet**

This wavelet can be more conveniently defined in frequency domain as an ideal band-pass filter:

$$\Psi(f) = \begin{cases} 1 & f_1 < |f| < f_2 \\ 0 & \text{otherwise} \end{cases} \tag{9.45}$$

The Shannon wavelet in time domain can be obtained by inverse Fourier transform as:

$$\begin{aligned} \psi(t) = \mathcal{F}^{-1}[\Psi(f)] &= \int_{-\infty}^{\infty} \Psi(f) e^{j2\pi ft} df = \int_{-f_2}^{-f_1} e^{j2\pi ft} df + \int_{f_1}^{f_2} e^{j2\pi ft} df \\ &= \frac{1}{\pi t}[\sin(2\pi f_2 t) - sin(2\pi f_1 t)] \end{aligned} \tag{9.46}$$

Note that while the Shannon wavelet has very good locality in frequency domain, it has relatively poor locality in time domain. However, this wavelet has some significant importance for the discussion of an algorithm for the reconstruction of the time signal from its wavelet coefficients, to be considered in the next section.

- **Morlet wavelet**
  As shown in Fig.9.3, a Morlet wavelet is a complex exponential $e^{j\omega_0 t} = \cos(j\omega_0 t) + j\sin(j\omega_0 t)$ modulated by a normalized Gaussian function $e^{-t^2/2}/\sqrt{2\pi}$:

$$\psi(t) = \frac{1}{\sqrt{2\pi}} e^{j\omega_0 t} e^{-t^2/2} = \frac{1}{\sqrt{2\pi}}[\cos(\omega_0 t) e^{-t^2/2} + j\sin(\omega_0 t) e^{-t^2/2}] \tag{9.47}$$

  According to the frequency shift property of the Fourier transform (Eq.3.104), the spectrum of the Morlet wave is another Gaussian function shifted by $-\omega_0$, as shown in the bottom panel of Fig.9.3:

$$\begin{aligned} \Psi(\omega) = \mathcal{F}[\psi(t)] &= \int_{-\infty}^{\infty} \psi(t) e^{-j\omega t} dt = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-t^2/2} e^{j(\omega-\omega_0)t} dt \\ &= e^{-(\omega-\omega_0)^2/2} = e^{-(2\pi(f-f_0))^2/2} \end{aligned} \tag{9.48}$$

  Note that when $\omega_0 = 0$, $\Psi(0) = e^{-\omega_0^2/2} > 0$, violating the requirement needed for the admissibility condition. However, when ω_0 is large enough, e.g., when $f_0 = 1$ Hz, $\omega_0 = 2\pi$, $\Psi(0) = e^{-6.28^2/2} = 2.7 \times 10^{-9}$ 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 1.

- **Derivative of Gaussian**
  This wavelet is the first order derivative of a normalized Gaussian function $g(t) = e^{-\pi(t/a)^2}/a$:

$$\psi(t) = \frac{d}{dt} g(t) = \frac{d}{dt}[\frac{1}{a} e^{-\pi(t/a)^2}] = -\frac{2\pi t}{a^3} e^{-\pi(t/a)^2} \tag{9.49}$$

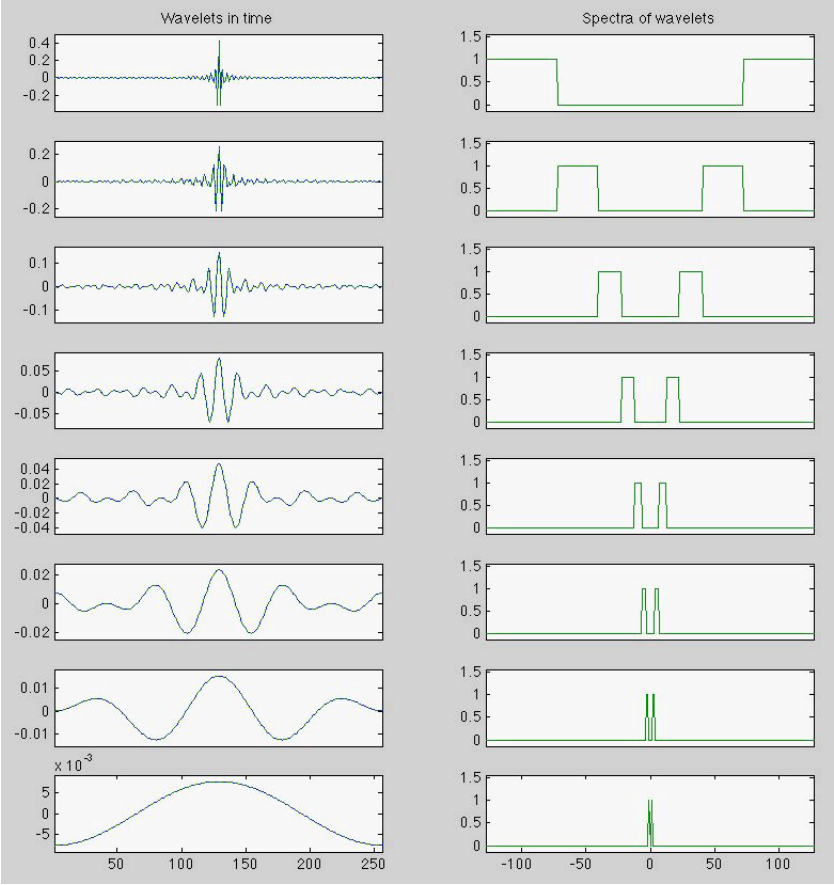

Figure 9.2 Shannon wavelets of different scale levels and their spectra

Note that the Gaussian function is normalized

$$\int_{-\infty}^{\infty} g(t)dt = 1 \tag{9.50}$$

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. 3.118) to be

$$\Psi(f) = \mathcal{F}[\psi(t)] = j2\pi f t e^{-\pi(af)^2} \tag{9.51}$$

- **Marr wavelet (Mexican hat)**
  This wavelet is the negative version of the second derivative of the Gaussian function $g(t) = e^{-\pi(t/a)^2}/a$. As we have

$$\frac{d^2}{dt^2}g(t) = \frac{d}{dt}[-\frac{2\pi t}{a^3}e^{-\pi(t/a)^2}] = -\frac{2\pi}{a^3}(1 - \frac{2\pi}{a^2})e^{-\pi(t/a)^2} \tag{9.52}$$

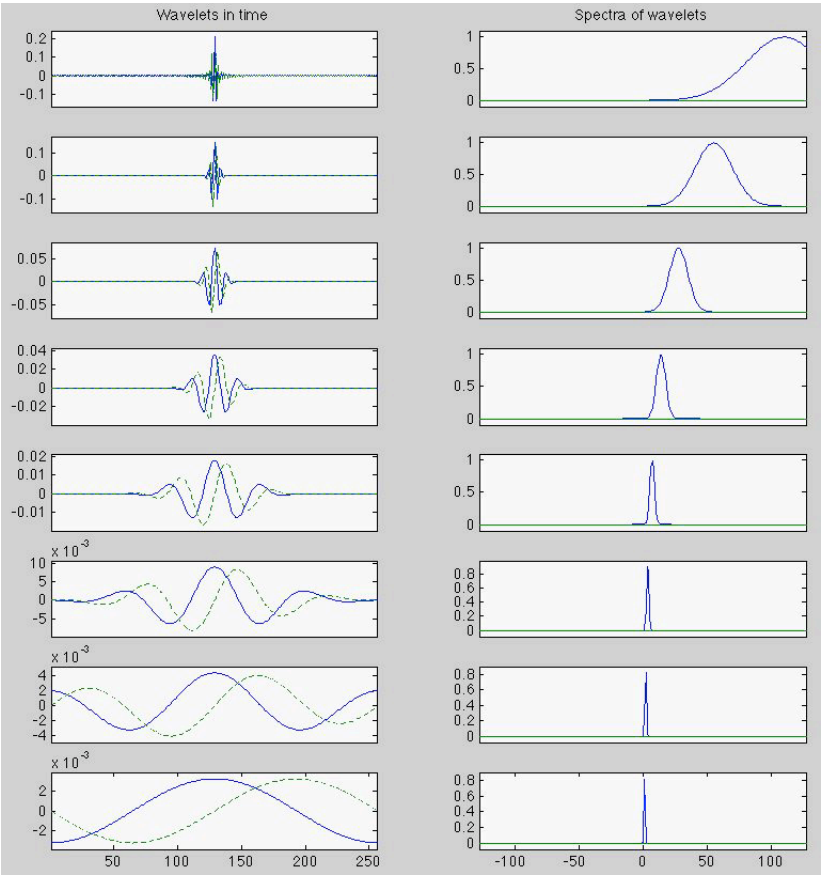

Figure 9.3 Morlet wavelets of different scale levels and their Spectra

the Marr wavelet is:

$$\psi(t) = \frac{2\pi}{a^3}(1 - \frac{2\pi}{a^2})e^{-\pi(t/a)^2} \tag{9.53}$$

If we let $a = \sqrt{2\pi\sigma^2}$, the Gaussian function $g(t)$ is normalized $\int g(t) = 1$, and the Marr wavelet becomes:

$$\psi(t) = \frac{1}{\sqrt{2\pi}\sigma^3}(1 - \frac{t^2}{\sigma^2})e^{-t^2/2\sigma^2} \tag{9.54}$$

The Marr wavelet function is also referred to as the Mexican hat function due to its shape. The Fourier transform of the Gaussian function is also Gaussian (Eq.3.148):

$$\mathcal{F}[\frac{1}{a}e^{-\pi(t/a)^2}] = e^{-\pi(af)^2} \tag{9.55}$$

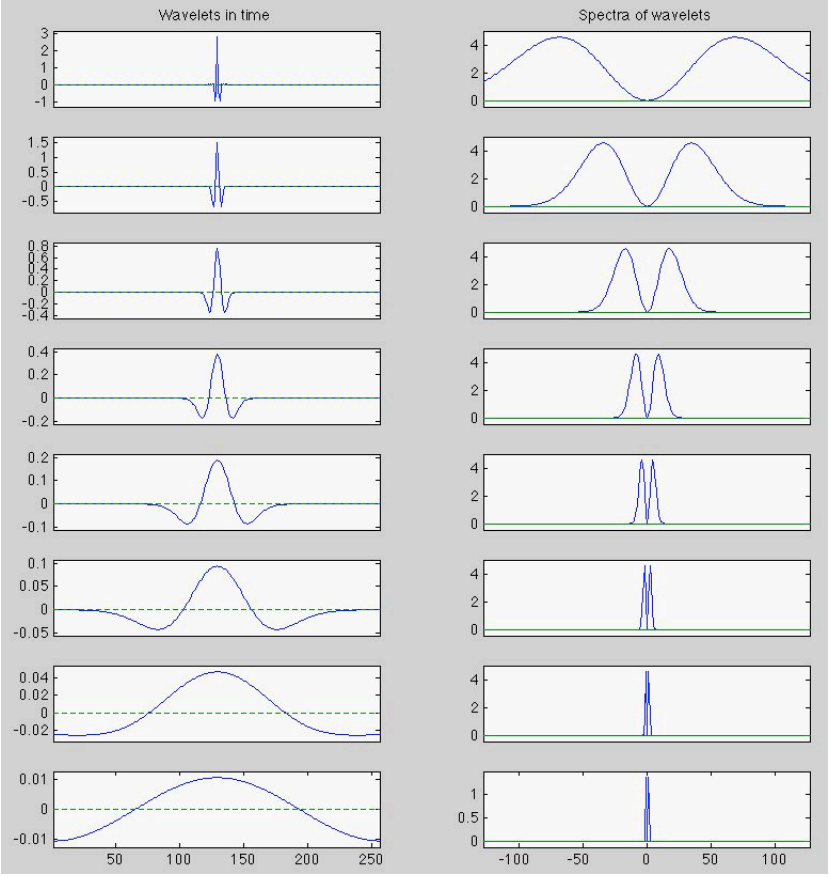

Figure 9.4 Marr wavelets of different scale levels and their spectra

and according to the time derivative property of the Fourier transform (Eq.3.118), we get the Fourier transform of the Marr wavelet

$$\Psi(f) = \mathcal{F}[\psi(t)] = -(j2\pi ft)^2 e^{-\pi(af)^2} = 4\pi^2 f^2 e^{-\pi(af)^2} \tag{9.56}$$

The Marr wavelet and its Fourier transform are shown in Fig.9.4. Note that as this wavelet function is real, its Fourier spectrum is symmetric with respect to zero frequency.

- **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):

$$\psi(t) = g_1(t) - g_2(t) = \frac{1}{a_1} e^{-\pi(t/a_1)^2} - \frac{1}{a_2} e^{-\pi(t/a_2)^2} \tag{9.57}$$

  The spectrum of this function is the difference between the spectra of the two Gaussian functions, which are also Gaussian:

$$\Psi(f) = G_1(f) - G_2(f) = e^{-\pi(a_1 f)^2} - e^{-\pi(a_2 f)^2} \tag{9.58}$$

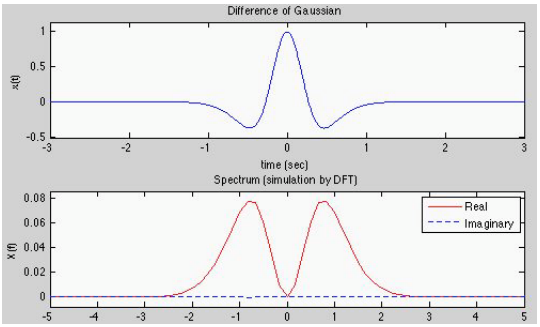

Figure 9.5 Difference of Gaussians and its spectrum

Note that $\Psi(0) = 1 - 1 = 0$. As can be seen in Fig.9.5, the difference of Gaussians looks very much like the second derivative of Gaussian (Marr) wavelet, and both functions could be brieviated as DoG. But note that they are two different types of functions.

9.5 Discrete-time wavelet transform (DTWT)

9.5.1 Discretization of wavelet functions

In order to actually obtain the wavelet transform $X(s, \tau)$ of a given signal $x(t)$ by a computer, we need to discretize not only both the signal $x(t)$ and the wavelet functions $\psi_{s,\tau}(t)$, but also the scaling factor s. Then the discrete version of the wavelet transform, referred to as the discrete-time wavelet transform (DTWT), can be carried out numerically.

Specifically, by sampling both the signal $x(t)$ and the mother wavelet $\psi(t)$, we get N samples for each functions: $x[m]$ and $\psi[m]$ $(m = 0, \cdots, N-1)$, together with their DFT $\mathcal{F}[x[m]] = X[n]$ and $\mathcal{F}[\psi[m]] = \Psi[n]$, where n is for all frequency components. Also, based on the mother wavelet, we generate a set of K wavelet functions $\psi_{s_k,0}[m] = \psi[m/s_k]$, each scaled by a different factor s_k:

$$s_k = s_0 2^{k/r} = s_0 (2^{1/r})^k, \qquad (k = 0, \cdots, K) \tag{9.59}$$

Here s_0 is the base scale factor and r is a parameter that determines the total number of scale levels $K = r \log_2(N/s_0)$. When the mother wavelet $\psi[m]$ is scaled by s_k, it becomes $\psi_{s_k,0}[m] = \psi[m/s_k]$, and, correspondingly, its DFT becomes $\Psi_{s_k,0} = \Psi[s_k n]$. When $k = 0$, the mother wavelet is scaled minimally by a factor s_0. When $k > 0$, the scale factor becomes $s_k = s_0 2^{k/r} > s_0$, and the wavelet $\psi_{s_k,0}[m] = \psi[m/s_k]$ is expanded in time domain and its DFT $\Psi_{s_k,0}[n] = \Psi[s_k n]$ is compressed in frequency domain. In the extreme case when $k = K$, the mother wavelet is maximally stretched by a factor of $s_K = N$, and its N-point Fourier spectrum $\Psi_{s_K,0}[n] = \Psi[s_K n]$ is maximally compressed to become a single point. Moreover, if $r > 1$, the base of the exponent is reduced from 2 to $2^{1/r} < 2$,

consequently we get a finer scale resolution with smaller step size between two consecutive scale levels. For example, when $r = 2$, the base of the exponent in Eq.9.59 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.

9.5.2 The forward and inverse transform

Now the DTWT of a discrete signal $x[m]$ can be obtained according to Eq.9.16 as a correlation of the signal and the wavelet function $\psi_{s_k,0}[m]$:

$$X[k,m] = x[m] \star \psi_{s_k,0}[m] \tag{9.60}$$

Alternatively, similar to the continuous case in Eq.9.19, this transform can also be carried out as a multiplication in frequency domain:

$$\hat{X}[k,n] = \mathcal{F}[X[k,m]] = X[n]\overline{\Psi}_{s_k,0}[n] \tag{9.61}$$

where $\hat{X}[k,n]$ is the DFT spectrum of the DTWT coefficients $X[k,m]$ of the signal $x[m]$. Now the DTWT can be readily obtained by the inverse DFT:

$$X[k,m] = \mathcal{F}^{-1}\left[\hat{X}[k,n]\right] = \mathcal{F}^{-1}\left[X[n]\overline{\Psi}_{s_k,0}[n]\right] \tag{9.62}$$

The inverse DTWT can also be more conveniently obtained in frequency domain, similar to the derivation of the inverse transform in the continuous case in Eq. 9.26. We first multiply both sides of Eq.9.61 by $\Psi_{s_k,0}[n]$ and then sum both sides over all K scale levels to get:

$$\sum_{k=0}^{K} \hat{X}[k,n]\Psi_{s_k,0}[n] = \sum_{k=0}^{K} [X[n]\overline{\Psi}_{s_k,0}[n]]\ \Psi_{s_k,0}[n] = X[n]\ \sum_{k=0}^{K} |\Psi_{s_k,0}[n]|^2 \tag{9.63}$$

But according to Eq.9.24, the summation of the daughter wavelets squared over all scales is a constant, i.e., in discrete case we have:

$$\sum_{k=0}^{K} |\Psi_{s_k,0}[n]|^2 = C \tag{9.64}$$

Now the above equation becomes:

$$X[n] = \frac{1}{C}\ \sum_{k=0}^{K} \hat{X}[k,n]\Psi_{s_k,0}[n] \tag{9.65}$$

Taking inverse DFT on both sides we get the inverse DTWT by which the original time signal $x[m]$ is reconstructed:

$$x[m] = \mathcal{F}^{-1}[X[n]] = \mathcal{F}^{-1}\left[\frac{1}{C}\sum_{k=0}^{K} \hat{X}[k,n]\Psi_{s_k,0}[n]\right] \tag{9.66}$$

9.5.3 A fast inverse transform

Next we consider a fast algorithm for the inverse DTWT, under the condition that the sum of the DFTs of the wavelet functions over all scales is constant:

$$\sum_{k=0}^{K} \Psi_{s_k,0}[n] = \sum_{k=0}^{K} \Psi[s_k n] = \sum_{k=0}^{K} \Psi[s_0 2^{k/r}\, n] = C, \qquad (\text{for all } n \neq 0) \tag{9.67}$$

where the constant C is not the same as that in Eq.9.64. This equation holds for all n representing different frequency components, independent of the specific function $\Psi[n]$.

To prove this result, we first consider the corresponding situation in the continuous case, the integral of an arbitrarily given function $f(x)$, scaled exponentially by a factor $s = b^u$:

$$\begin{aligned} \int_{-\infty}^{\infty} f(b^u x)du &= \int_{-\infty}^{\infty} f(sx)d(\log_b s) = \frac{1}{ln\, b}\int_0^{\infty} \frac{f(sx)}{s}ds \\ = \frac{1}{ln\, b}\int_0^{\infty} \frac{f(sx)}{sx}d(sx) &= \frac{1}{ln\, b}\int_0^{\infty} \frac{f(s')}{s'}ds' = C \end{aligned} \tag{9.68}$$

Here we have assumed $s' = sx$, and that the integral converges to a constant C. This result indicates that the summation of all exponentially scaled versions of any function $f(x)$ is a constant over the entire domain x of the function, independent of the specific form of the function. Now we see that the summation in Eq.9.67 above is simply the discrete approximation of the integral in the continuous case. If the parameter r takes a large enough value, then the resolution for the different scales should be fine enough, and the summation will be a valid approximation of the integral in Eq.9.68, therefore it should also converge to a constant. For example, as shown Fig.9.6, the differently scaled Morlet and Marr wavelets in frequency domain do indeed add up to a constant over the entire horizontal axis for frequency.

We are now ready to consider the fast algorithm for the inverse DTWT. Based on the result in Eq.9.67, we see that the same should also hold for $\overline{\Psi}_{s_k,0}[n]$ (in fact the DFTs of all typical wavelets discussed above are real $\overline{\Psi}_{s_k,0}[n] = \Psi_{s_k,0}[n]$), i.e., they also add up to a constant $\sum_{k=0}^{K} \overline{\Psi}_{s_k,0}[n] = C$. We can show that the inverse DTWT can be carried out simply by summing all the DTWT coefficients obtained by Eq.9.62:

$$\begin{aligned} \sum_{k=0}^{K} X[k,m] &= \sum_{k=0}^{K} \mathcal{F}^{-1}\left[X[n]\overline{\Psi}_{s_k,0}[n]\right] = \sum_{k=0}^{K}\left[\sum_{n=0}^{N-1} X[n]\overline{\Psi}_{s_k,0}[n]e^{j2\pi mn/N}\right] \\ = \sum_{n=0}^{N-1} X[n]\left[\sum_{k=0}^{K} \overline{\Psi}_{s_k,0}[n]\right] e^{j2\pi mn/N} &= C\sum_{n=0}^{N-1} X[n]e^{j2\pi mn/N} = C\,x[m] \end{aligned} \tag{9.69}$$

Therefore the original time signal can be reconstructed by the inverse DTWT:

$$x[m] = \frac{1}{C}\sum_{k=0}^{K} X[k,m], \qquad \text{where} \qquad C = \sum_{k=0}^{K} \overline{\Psi}_{s_k,0}[n] \tag{9.70}$$

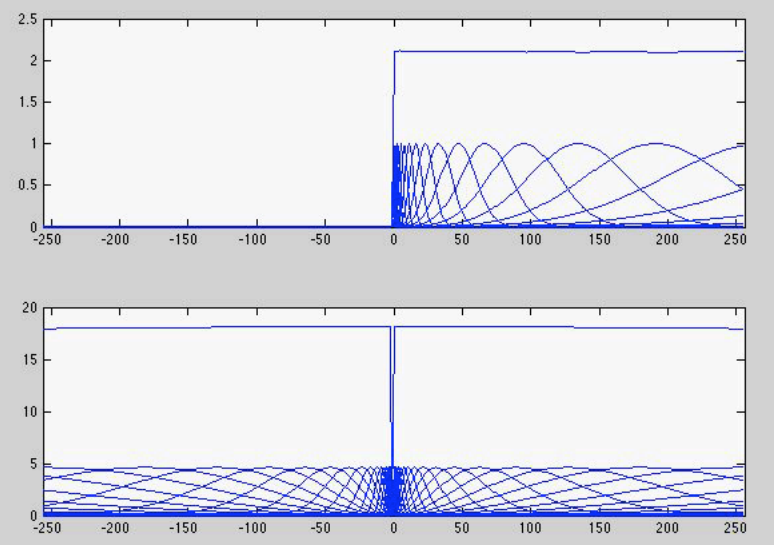

Figure 9.6 Summations of the spectra of Morlet wavelets (top) and Marr (Mexican hat) wavelets (bottom)

This fast algorithm for inverse DTWT can also be illustrated as an all-pass filter bank. We first consider the case of the DTWT based on the Shannon wavelet, whose spectrum $\Psi(f)$ is an ideal band-pass filter in frequency domain (Eq.9.45). As the magnitude of this filter is 1 within a finite passing band $\Delta f = f2 - f1$, but 0 elsewhere, it preserves all signal information inside the passing band while suppressing to zero all frequency components of the signal outside the passing band. Moreover, as we choose the scale levels in such a way that these band-pass filters $\Psi_{s_k}(f)$ corresponding to all K different scales form a filter bank that completely covers the entire frequency range without any overlap or gap, Eq.9.67 is satisfied. These ideal band-pass filters in the filter bank form an all-pass filter with a collective frequency response equal to 1 at all frequencies $n \neq 0$, except at zero frequency where all $\Psi_{s_k,0}[0] = 0$ (Eq.9.10), as required by the admissibility condition. In frequency domain, the outputs of these filters are simply the DFTs of the DTWT coefficients $\hat{X}[k,n] = X[n]\ \Psi_{s_k,0}[n]$, and when combined together, they carry the complete information contained in the signal $x(t)$, except its DC component. Therefore it becomes clear that the original signal $x(t)$ can be perfectly reconstructed as the sum of the outputs from all filters of the all-pass filter bank, as indicated in Eq.9.70. As a specific example, eight Shannon wavelets and their spectra as ideal band-pass filters are shown in Fig.9.2, and a filter bank composed of these band-pass filters is illustrated in Fig.9.7. This filter bank can therefore be considered as the inverse DTWT.

Due to the result in Eq.9.70, the discussion above for the Shannon wavelets applies to all other wavelet functions, such as Morlet and Marr wavelets, as they can all form a all-pass filter bank due to the result of Eq.9.67, also shown

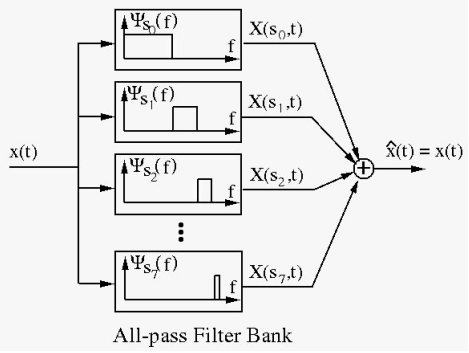

Figure 9.7 All-pass filter bank composed of band-pass wavelets

in Fig.9.6. Although their spectra corresponding to different scale levels are no longer ideal band-pass filters and their passing bands have significant overlaps, these filters still add up to a constant over all frequencies, and their outputs also carry collectively the complete information of the signal, which can therefore be reconstructed as the sum of the outputs of all these band-pass filters.

For example, consider the wavelet transform of a sawtooth time signal of $N = 128$ samples, shown in Fig.9.8. Here we choose to use Morlet wavelets of $K = 8$ different scale levels, corresponding to the same number of band-pass filters. These wavelets $\psi_s(t)$ in time domain and their spectra $\Psi_s(f)$ in frequency domain have already been shown in Fig. 9.3. The DTWT coefficients $X[k, m]$ corresponding to different scale levels s_k are shown on the left of Fig.9.8, and their partial sums are shown on the right, where the kth panel is the partial sum of the first k scale levels. We see that the partial sums as the approximation of the original sawtooth signal $x[m]$ improves progressively as more scale levels are included, until eventually a perfect reconstruction of the signal is obtained when all K scale levels are included.

9.6 Wavelet Transform Computation

Here we give a few segments of C code needed to implement the DTWT algorithm discussed above.

- Scale levels

```
r=2;                          // scale resolution
s0=1;                         // smallest scale
K=r*log2((float)N/s0);        // total number of scale levels
scale=alloc1df(K);            // allocate memory for K scales
for (k=0; k<K; k++) {
  scale[k]=s0*pow(2.0,k/r);   // kth scale s_k
}
```

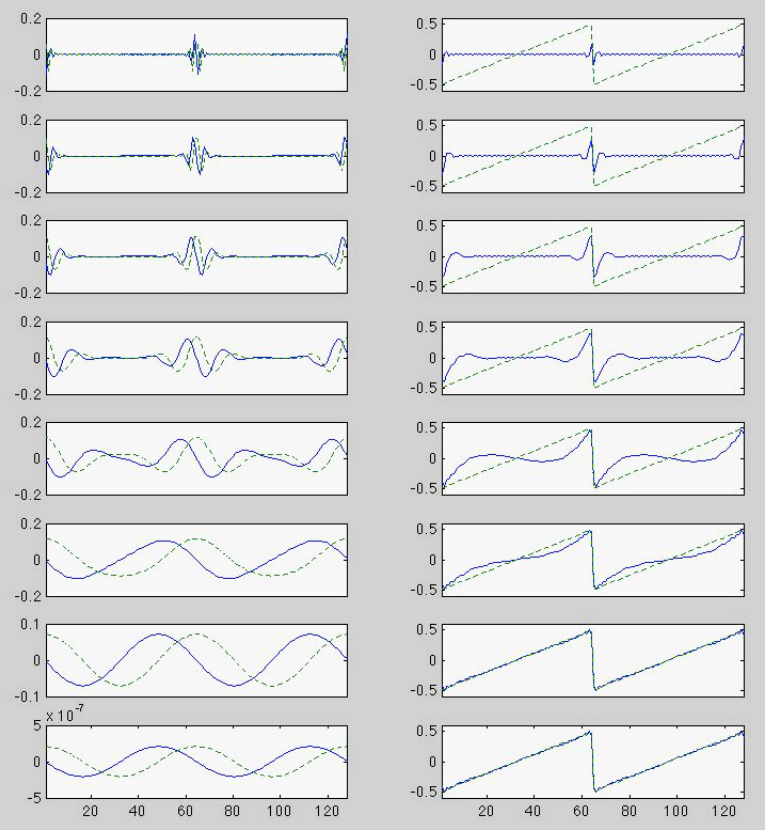

Figure 9.8 The reconstruction of a sawtooth signal (right) as the sum of its DTWT coefficients over $K = 8$ scale levels (left)

In the panels on the left, the solid curves are for the real part of the DTWT and the dashed curves for the imaginary part. In the panels on the right, the solid curves are for the partial sums of the DTWT coefficients, in comparison with the original signal shown by the dashed curves.

The scales corresponding to three different sets of parameters are plotted in Fig.9.9 to show how the resolution r and base scale s0 affect the scales s_k.

- Wavelet functions:
  As both forward and inverse DTWT can be 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 K scales:

```
f0=0.6;                        // wavelet parameter
for (k=0; k<K; k++) {          // for all K scale levels
  for (n=0; n<N; n++) {        // for all N frequencies
```

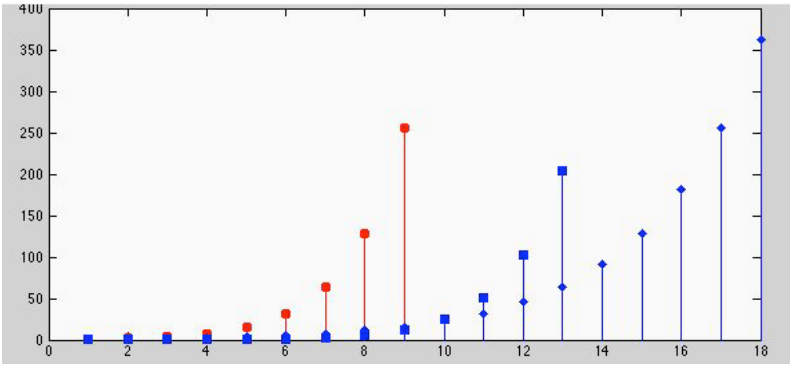

Figure 9.9 Scales s_k versus $k = 0, \cdots, K$ corresponding to different parameters r and s_0 for DTWT of a signal with $N = 512$ samples
The circles represent $K = 9$ scales corresponding to $r = 1$ and $s_0 = 1$; the squares represent $K = 13$ scales corresponding to $r = 1$ and $s_0 = 0.05$; and the diamonds represent $K = 18$ scales corresponding to $r = 2$ and $s_0 = 1$.

```
wavei[k][n]=0;           // imaginary part is zero
v=2*Pi*(scale[k]*((float)(n-N/2)/N)-f0);   // DC in middle
waver[k][n]=exp(-v*v/2); // real part of spectrum
      }
    }
```

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 K scales. Also shown below is the code for generating Mexican hat wavelets of K scales:

```
    a=2;                          // wavelet parameter
    for (k=0; k<K; k++) {         // for all K scale levels
      for (n=0; n<N; n++) {       // for all N frequencies
wavei[k][n]=0;           // imaginary part is zero
v=a*scale[k]*(n-N/2)/N;  // DC in middle
waver[k][n]=4*Pi*Pi*v*v*exp(-Pi*v*v);  // real part of spectrum
      }
    }
```

- 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 $K \times N$ arrays Xr and Xi for wavelet coefficients of K scales and N time translations:

```
    dft(xr,xi,N,0);            // DFT of time signal to get spectrum
    for (k=0; k<K; k++) {      // for all K scale levels
```

```
      for (n=0; n<N; n++) {    // for all N frequencies
Xr[k][n]=xr[n]*waver[k][n]+xi[n]*wavei[k][n];
Xi[k][n]=xi[n]*waver[k][n]-xr[n]*wavei[k][n];
      }
      dft(Xr[k],Xi[k],N,1);    // inverse DFT to go back to time domain
    }
```

- Inverse DTWT:
  Listed below is the code for the inverse DTWT algorithm based on Eq.9.66. Again, the real and imaginary parts of the DTWT coefficients are stored in the two $K \times N$ 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 (k=0; k<K; k++) {    // for all K scale leves
dft(Xr[k],Xi[k],N,0);  // DFT of DTWT coefficients to frequency domain
for (n=0; n<N; n++) {
  yr[n]=yr[n]+Xr[k][n]*waver[k][n]-Xi[k][n]*wavei[k][n];
  yi[n]=yi[n]+Xr[k][n]*wavei[k][n]+Xi[k][n]*waver[k][n];
}
dft(yr,yi,N,1);        // inverse DFT to go back to time domain
      }
```

  The code based on Eq.9.70 is trivial and not listed.

As some examples of the DTWT, a set of typical signals as well as their DTWT transforms are shown in Fig.9.10 qualitatively in image forms. All of these DTWTs are based on the Morlet wavelets. 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.

9.7 Filtering Based on Wavelet Transform

Here we consider some examples to illustrate the filtering effects of the wavelet transform in comparison with that based on the Fourier transform.

Example 9.1: The monthly Dow Jones Industrial Average (DJIA) from 1999 to 2008 and its Fourier spectrum are plotted in top two panels of Fig.9.11. The LP-filtered spectrum is plotted in panel 3. Similar LP-filtering can also be carried out based on the wavelet transform, as shown in Fig.9.12. The LP-filtered data

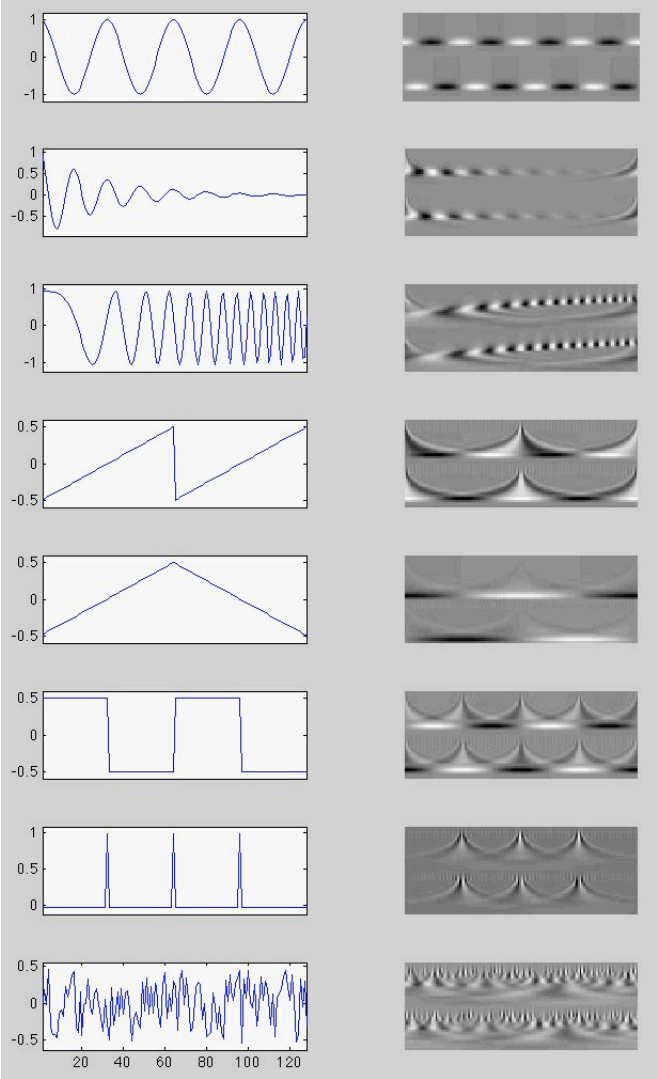

Figure 9.10 Typical signals and their DTWT transforms
Eight typical signals $x(t)$ (left) and their DTWT (Morlet) coefficients $X[k, m]$ as a 2-D image (right), with the real part is in the upper part and the imaginary in the lower part.

obtained from both the Fourier and wavelet transforms are re-plotted as the solid and dashed curves 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.

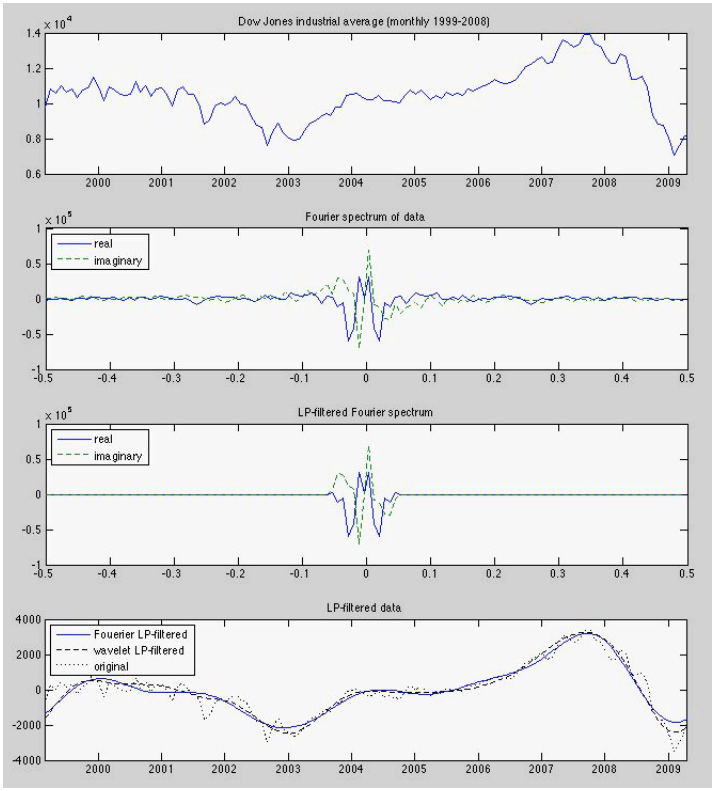

Figure 9.11 Monthly Dow Jones Industrial Average (DJIA) from 1999 to 2008
The four panels are, respectively, the DJIA index, its Fourier transform, the LP-filtered spectrum, and the LP-filtered data by both the Fourier and wavelet transforms.

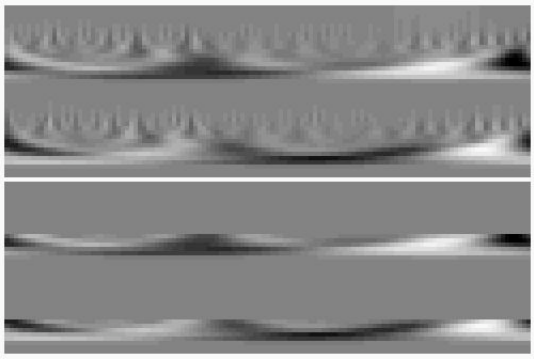

Figure 9.12 Wavelet transform or DJIA and its filtering
The top panel shows the wavelet transform coefficients and the bottom panel shows the LP-filtered version of the same spectrum. All coefficients for higher scales are suppressed to zero (gray in the image).

Example 9.2: In this example we compare the filtering effects of an exponential chirp signal based on the Fourier transform and the wavelet transform. In general, a chirp is a signal whose frequency increases or decreases with time. If the frequency increases linearly in time, the signal is a linear chirp, if the frequency increases exponentially in time, the signal is an exponential chirp. As the frequency changes continuously over time, it may seem that filtering out certain frequency should only affect the signal locally during the time interval corresponding to the frequency removed. However, this is not actually the case if the filtering is carried out in Fourier domain.

As shown in Fig.9.13, the original signal (top panel) is first Fourier transformed to get its spectrum (2nd panel), then certain frequency components in the signal spectrum are completely suppressed to zero by an ideal band-pass filter (3rd panel), and finally the signal is reconstructed by the inverse Fourier transform of the filtered version of the spectrum (bottom). Note that although only a relatively narrow frequency band is suppressed, the entire time signal is affected, including the slow changing portion on the very left, as well as the time interval (roughly from 150 to 250) corresponding to the frequency components that are 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 the suppressed frequency components also contribute to the slow changing portion of the signal as well.

On the other hand, the filtering based on the wavelet transform demonstrate some different effect, as shown in Fig.9.14, where the same chirp signal and its DTWT coefficients are shown in the top two panels, and the filtering in transform domain and the filtered time signal are shown respectively in the 3rd and 4th 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, in the reconstructed signal, only a local portion (also roughly from 150 to 250) of the signal corresponding to the scale levels being suppressed is significantly affected, while the rest of the signal has been affected little. This very different filtering effect is due to the fact that the wavelet transform maintains the local information in time as well as in different scales levels.

Example 9.3: One of the weaknesses of the Fourier transform is that it is insensitive to non-stationary characteristics in the signal because all 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, and shown on the top and bottom panels on the left in Fig.9.15, and the corresponding Fourier spectrum shown on right. As we can see, the spiky noise has a very wide-spread spectrum in frequency domain, consequently all frequency components of the signal are affected by the noise. In particular, some of the weaker

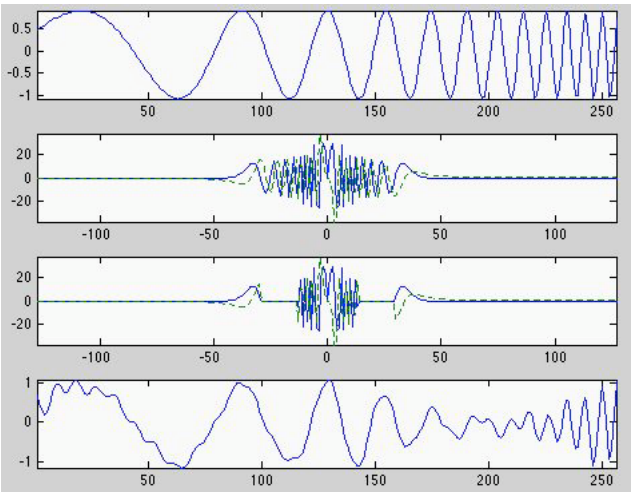

Figure 9.13 FT filtering of chirp signal

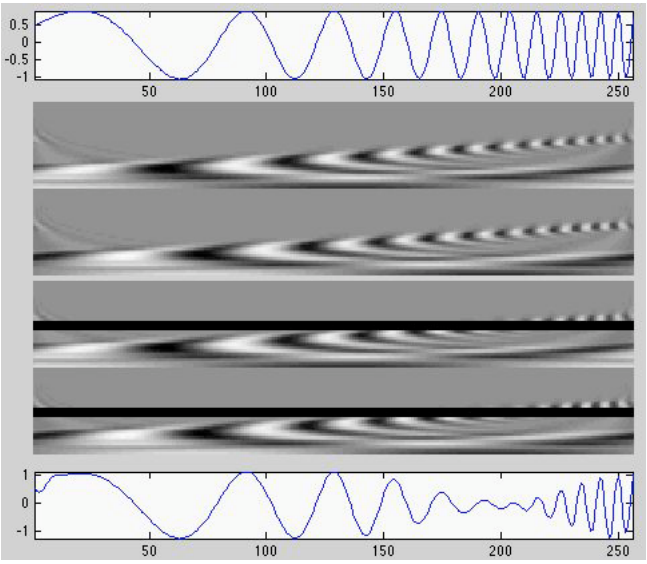

Figure 9.14 CWT filtering of chirp signal

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.

The same problem can also be addressed by the wavelet filtering, as shown in Fig.9.16. The top, middle and bottom panels on the left show the original signal and its reconstructions after high-pass and low-pass filtering, respectively, and their corresponding wavelet coefficients are shown in the panels on the right. We see that it is possible to separate the noise from the signal by wavelet filtering

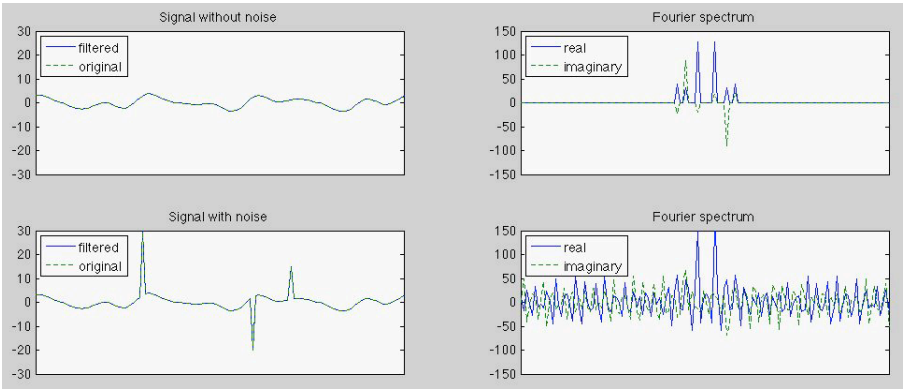

Figure 9.15 A noise-contaminated signal and its Fourier spectrum

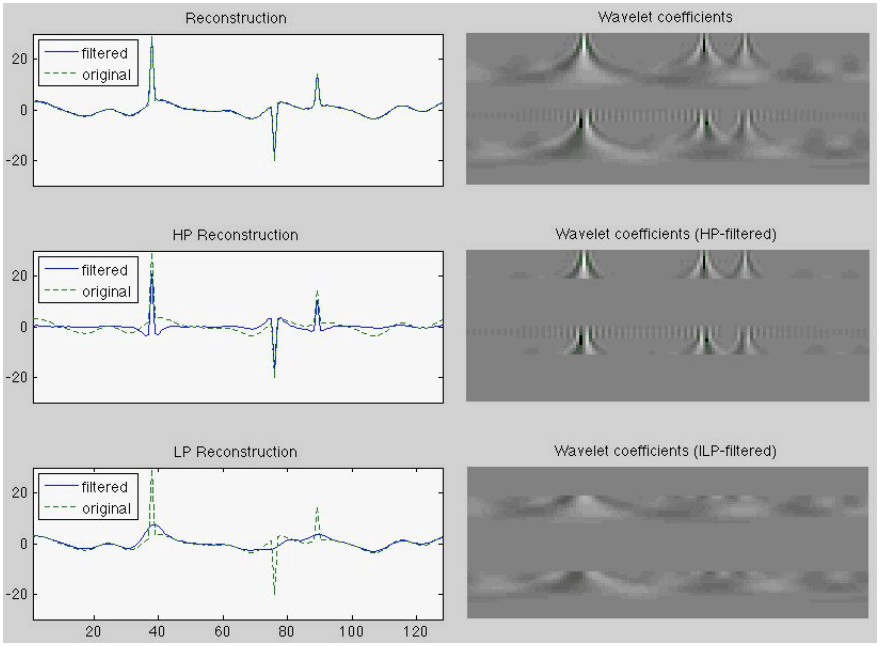

Figure 9.16 Separation of the signal and noise by wavelet filtering
Time signals are shown on the left, and their wavelet coefficients are shown on the right. The original signal and its reconstructions after HP and LP filtering are shown in the top, middle and bottom rows, respectively.

(top-left panel), due obviously to the temporal locality of the wavelet transform. The signal is reasonably recovered after low-pass filtering (bottom left), while the spiky noise is separated out by high-pass filtering (middle left).

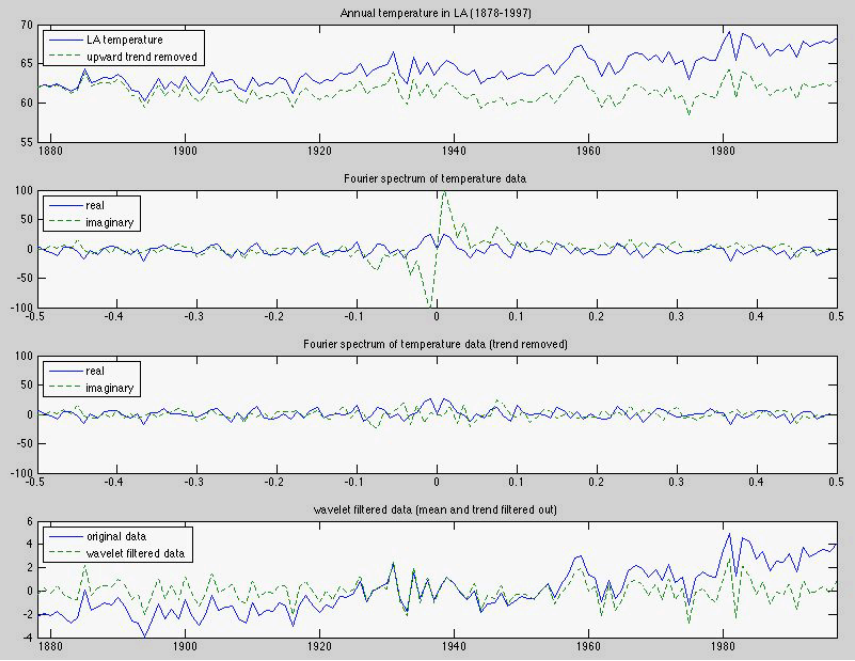

Figure 9.17 Annual temperature in LA area from 1878 to 1997

Example 9.4: Here we consider the annual average temperature in Los Angeles area from 1878 to 1997 (from NOAA National Weather Service Center in the US), as shown in the top panel of Fig.9.17 (solid curve). The data clearly shows a upward trend of the annual temperature. In fact there is a 5.57° F annual temperature rise during these 120 years, with an average annual increase of 0.0464° F.

If needed, the upward drift in the data can be removed in either time domain or some transform domain. In time domain, we can 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.9.17. We next consider if and how this could 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.9.17. We see that their real parts are the same, but their imaginary parts differ significantly at the low frequency region. The positive and negative peaks in the imaginary part of the spectrum for the original data (2nd panel) represent the slow-changing upward trend (an odd function), which no longer exist in the spectrum of the data when this trend is removed (3rd panel). As the frequency components for the slow-changing trend are mixed with those for the more rapid variations, it is hard to separate out the trend from the rest of the signal by filtering in frequency domain.

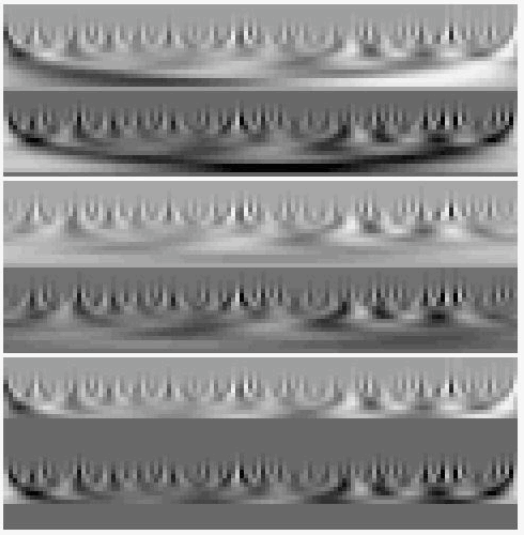

Figure 9.18 Wavelet transform of LA temperature data
The top panel shows the wavelet transform of the original data. In comparison, the middle panel shows the transform of the same data but with the upward trend removed. The bottom panel shows the filtering in wavelet transform domain by suppressing to zero (grey) the coefficients representing the low-scale (frequency) components in the signal.

On the other hand, the wavelet transform generates some different result in its transform domain, as shown in Fig.9.18. The wavelet transforms of the original data with and without the upward trend are shown in the top two panels, while in the bottom panel the coefficients for the trend are filtered out. After the inverse wavelet transform, the temperature signal is reconstructed, as shown in the bottom panel of Fig.9.17. Indeed the upward trend is removed by wavelet filtering.

Provide a platform, a toolbox for a wide variety of data processing and analysis methods based on orthogonal transforms.

10 Multiresolution Analysis and Discrete Wavelet Transform

Similar to the Fourier and other orthogonal transforms, the continuous wavelet transform (CWT) discussed in the previous chapter also converts a signal $x(t)$ into the transform domain by an integral transform based on a kernel function, the wavelet function $\psi_{s,\tau}(t)$ in this case. However, different from all those transforms, the CWT of a 1-D signal $x(t)$ is a 2-D function $X(s,\tau)$ with high redundancy. Moreover, unlike all transforms discussed before, the CWT is not an orthogonal transform, as its transform kernel functions do not form an orthogonal basis to span the function space containing the signal functions $x(t)$ (Eq.9.41). As the result, CWT, or its discrete version DTWT, with some favorable features in filtering, is not suitable for data compression.

In this chapter, we will consider the concept of multiresolution analysis (MRA), also called multi-scale approximation (MSA), based on which a set of orthogonal wavelet functions can be constructed to 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.

Specifically, the two parameters s and τ of a wavelet function considered previously

$$\psi_{s,\tau}(t) = \frac{1}{\sqrt{s}}\psi(\frac{t-\tau}{s}) \tag{10.1}$$

are discretized in a binary fashion to become:

$$\psi_{j,k}(t) = \frac{1}{\sqrt{2^{-j}}}\psi(\frac{t-2^{-j}n}{2^{-j}}) = 2^{j/2}\psi(2^j t-k), \quad (j,k\in\mathbb{Z}=\{\cdots,-1,0,1,\cdots\}) \tag{10.2}$$

where $\mathbb{Z}=\{\cdots,-1,0,1,\cdots\}$ is the set of all integers. This function is either an expanded (dilated) version of the mother wavelet $\psi(t)$ if $j<0$, or a compressed version of $\psi(t)$ 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 will further impose the orthogonality requirement so that all daughter wavelets $\psi_{j,k}(t)$ are orthogonal with respect to not only integer translations (in terms of k), but also binary scaling (in terms of j). In other words, for a given j, these functions form an orthogonal basis that

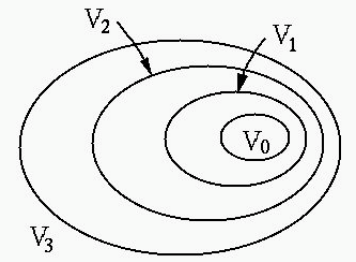

Figure 10.1 The nested V_j spaces for MRA

spans a space of a certain scale level j, and all bases across different scale levels are also orthogonal to each other. In the following, we will develop the necessary theory for the construction of such a set of orthogonal wavelet functions.

10.1 Multiresolution Analysis

10.1.1 Scale spaces

Definition 10.1. *The multiresolution analysis is based on a sequence of nested* scale spaces $V_j \subset L^2(\mathbb{R})$*:*

$$\{0\} \subset \cdots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \subset \cdots \subset L^2(\mathbb{R}) \tag{10.3}$$

that satisfies the following requirements:

- *Completeness: the union of the nested spaces is the entire function space and their intersection is a set containing* 0 *as its only member:*

$$\cup_{j\in\mathbb{Z}} V_j = L^2(\mathbb{R}), \qquad \cap_{j\in\mathbb{Z}} V_j = \{0\} \tag{10.4}$$

- *Self-similarity in scale:*

$$x(t) \in V_0 \quad \textit{iff} \quad x(2^j t) \in V_j, \quad j \in \mathbb{Z} \tag{10.5}$$

- *Self-similarity in translation and scale:*

$$x(t) \in V_0 \quad \textit{iff} \quad x(t-k) \in V_0, \quad k \in \mathbb{Z} \tag{10.6}$$

- *Existence of a basis, called Riesz basis,* $\theta_k(t)$ $(k \in \mathbb{Z})$ *that spans* V_0*:*

$$V_0 = span(\theta_k, \ \ k \in \mathbb{Z}) \tag{10.7}$$

Note that although Eq.10.6 is only for self-similarity in V_0, i.e., any $x(t) \in V_0$ translated by integer k is still in V_0, we can generalize this result to V_j. To do so, we replace t by $2^j t$ in 10.6 and combine it with Eq.10.5 to get:

$$x(t) \in V_0 \quad \text{iff} \quad x(2^j t) \in V_j \quad \text{iff} \quad x(2^j t - k) \in V_j \tag{10.8}$$

If we define a new function $y(t) = x(2^{-j}t)$, the relationship above can also be expressed as

$$y(t) \in V_j \quad \text{iff} \quad y(t - 2^{-j}k) \in V_j \tag{10.9}$$

which indicates that 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 at a different level of details in each subspace $V_j \subset L^2(\mathbb{R})$. We consider the following two cases:

- First, a space V_j ($j > 0$) is spanned by the basis functions that are time-compressed versions of $\theta_k(t)$ in V_0. As the width of these basis function becomes 2^j times narrower, they are capable of representing variations of smaller scales or more detailed information in a given signal $x(t)$ than the basis functions $\theta_k(t)$ of V_0, i.e., $V_0 \subset V_j$. In particular, when $j \to \infty$, the basis functions of V_j is maximally compressed to become an impulse function with a zero width and infinite height (for finite energy), and the space they span becomes the entire $L^2(\mathbb{R})$ space capable of representing all details in a signal, as indicated by Eq.1.5 at the beginning of the book:

$$\int_{-\infty}^{\infty} x(\tau)\delta(t - \tau)d\tau = x(t) \in L^2(\mathbb{R}) \tag{10.10}$$

- Second, a space V_{-j} ($j > 0$) is spanned by the basis functions which are time-expanded versions of $\theta_k(t)$ in V_0. As the width of these basis function becomes 2^j times wider, they can only represent variations of larger scales or less detailed information in a given signal $x(t)$ than the basis functions $\theta_k(t)$ of V_0, i.e., $V_{-j} \subset V_0$. In particular, when $j \to \infty$, the basis function is expanded to have an infinite width but zero height, a constant 0 for all t, and the corresponding space becomes $\{0\}$, a set containing 0 as its only member.

Based on the Riesz basis $\theta(t)$, a set of orthogonal scaling functions $\phi(t)$, also called *father wavelet*, can be constructed in frequency domain:

$$\Phi(f) = \mathcal{F}[\phi(t)] = \frac{\Theta(f)}{[\sum_k |\Theta(f - k)|^2]^{1/2}} \tag{10.11}$$

where $\Theta(f) = \mathcal{F}[\theta(t)]$ is the spectrum of $\theta(t)$. We now show that this scaling function $\phi(t)$ is indeed orthogonal to itself translated by any integer amount:

$$< \phi(t - k), \phi(t) >= \int_{-\infty}^{\infty} \phi(t - k)\overline{\phi}(t)dt = \delta[k] \tag{10.12}$$

We first represent this orthogonality in frequency domain. As the inner product in the equation is actually the self-correlation of $\phi(t)$ evaluated at $t = k$ for all $k \in \mathbb{Z}$, it can be expressed as the product of the self-correlation $r_\phi(t)$ and an impulse train with unity interval, and the equation above becomes:

$$r_\phi(\tau)\big|_{\tau=k\in\mathbb{Z}} = \int_{-\infty}^{\infty} \phi(t - \tau)\overline{\phi}(t)dt\big|_{\tau=k\in\mathbb{Z}} = r_\phi(\tau)\sum_{k\in\mathbb{Z}} \delta(\tau - k) = \delta[k] \tag{10.13}$$

This product in time domain corresponds to a convolution in frequency domain of the spectrum of the correlation $r_\phi(t)$ (Eq.3.111) and that of the impulse train (Eq.3.2.5):

$$|\Phi(f)|^2 * \sum_{k\in\mathbb{Z}} \delta(f-k) = \sum_{k\in\mathbb{Z}} |\Phi(f-k)|^2 \tag{10.14}$$

Also, as the Fourier transform of $\delta[k]$ on the right-hand side of the equation above is 1, we have:

$$\sum_{k\in\mathbb{Z}} |\Phi(f-k)|^2 = 1 \tag{10.15}$$

Obviously this condition for orthogonality is satisfied by the father wavelet $\Phi(f)$ constructed in Eq.10.11, i.e., the scaling functions translated by different integer k $\phi_k(t) = \phi(t-k)$ form an orthonormal basis that spans V_0. We denote these functions by $\phi_{0,k}(t)$ and write

$$V_0 = span(\phi_{0,k}(t), \quad k \in \mathbb{Z}) \tag{10.16}$$

This result can be generalized to space V_j. Replacing t in Eq.10.12 with $2^j t$ we get:

$$\begin{aligned} &\int_{-\infty}^{\infty} \phi(2^j t-k)\overline{\phi}(2^j t)d(2^j t) = \int_{-\infty}^{\infty} \sqrt{2^j}\phi(2^j t-k)\sqrt{2^j}\overline{\phi}(2^j t)dt \\ &= <\phi_{j,k}(t), \phi_{j,0}(t)> = \delta[k] \end{aligned} \tag{10.17}$$

where we have defined

$$\phi_{j,k}(t) = \sqrt{2^j}\phi(2^j t-k) = 2^{j/2}\phi(2^j t-k) \in V_j, \quad k \in \mathbb{Z} \tag{10.18}$$

which, according to Eq.10.8, is in V_j, i.e., they form an orthogonal basis in V_j:

$$V_j = span(\phi_{j,k}(t), \quad k \in \mathbb{Z}) \tag{10.19}$$

For $j > 0$, functions $\phi_{j,k}(t)$ are compressed in time (shorter duration) but scaled up in value (larger amplitude), and therefore they span a space $V_j \supset V_0$ that can better approximate a given function.

The scaling functions in spaces V_j of different levels are related. Specifically, the scaling functions $\phi(t) \in V_0 \subset V_1$ can be expressed in terms of the orthonormal basis $\phi_{1,k}(t) = \sqrt{2}\phi(2t-k)$ of V_1:

$$\phi(t) = \sum_{k\in\mathbb{Z}} h_0[k]\phi_{1,k}(t) = \sqrt{2}\sum_{k\in\mathbb{Z}} h_0[k]\phi(2t-k) \tag{10.20}$$

where the coefficients $h_0[k]$ can be found as the projection of $\phi(t)$ onto the kth basis function $\phi_{1,k}(t) = \sqrt{2}\phi(2t-k)$:

$$h_0[k] = <\phi(t), \sqrt{2}\phi(2t-k)> = \sqrt{2}\int_{-\infty}^{\infty} \phi(t)\overline{\phi}(2t-k)dt \tag{10.21}$$

Next, this relationship between V_0 and V_1 can be generalized to V_j and V_{j+1}. Replacing t in Eq.10.20 with $2^j t - l$, we get:

$$\phi(2^j t - l) = \sqrt{2}\sum_{k\in\mathbb{Z}} h_0[k]\phi(2(2^j t - l) - k) = \sqrt{2}\sum_{k\in\mathbb{Z}} h_0[k]\phi(2^{j+1}t - (2l + k)) \tag{10.22}$$

But due to Eq.10.18, we have

$$\phi_{j,l}(t) = \sum_{k\in\mathbb{Z}} h_0[k]\phi_{j+1,2l+k}(t) = \sum_{k'\in\mathbb{Z}} h_0[k' - 2l]\phi_{j+1,k'}(t) \tag{10.23}$$

where we have assumed $k' = 2l + k$, i.e., $k = k' - 2l$. This relationship can also be described in frequency domain. Taking the Fourier transform of Eq.10.20, we get

$$\begin{aligned}
\Phi(f) &= \int_{-\infty}^{\infty} \phi(t)e^{-j2\pi ft}dt = \sqrt{2}\sum_{k\in\mathbb{Z}} h_0[k]\int_{-\infty}^{\infty} \phi(2t - k)e^{-j2\pi ft}dt \\
&= \sqrt{2}\sum_{k\in\mathbb{Z}} h_0[k]\int_{-\infty}^{\infty} \phi(t')e^{-j2\pi f(t'+k)/2}d(\frac{t'}{2}) \\
&= \frac{1}{\sqrt{2}}\sum_{k\in\mathbb{Z}} h_0[k]e^{-jk\pi f}\int_{-\infty}^{\infty} \phi(t')e^{-j2\pi ft'/2}dt' \\
&= \frac{1}{\sqrt{2}}H_0(\frac{f}{2})\Phi(\frac{f}{2})
\end{aligned} \tag{10.24}$$

where we have assumed $t' = 2t - k$, and $H_0(f)$ is the discrete-time Fourier transform of the coefficients $h_0[k]$ for the *scaling filter*:

$$H_0(f) = \sum_{k\in\mathbb{Z}} h_0[k]e^{-j2k\pi f} \tag{10.25}$$

Note that for As $h_0[k]$ is discrete with sampling frequency $F = 1$ by assumption, $H_0(f \pm 1) = H_0(f)$ is periodic with a period of $F = 1$. Eq.10.24 can be further recursively expanded to become:

$$\Phi(f) = \frac{1}{\sqrt{2}}H_0(\frac{f}{2})[\frac{1}{\sqrt{2}}H_0(\frac{f}{4})\Phi(\frac{f}{4})] = \cdots = \prod_{j=1}^{\infty}\frac{1}{\sqrt{2}}H_0(\frac{f}{2^j})\phi(0) = \prod_{j=1}^{\infty}\frac{1}{\sqrt{2}}H_0(\frac{f}{2^j}) \tag{10.26}$$

The last equal sign is based on the assumption that $\phi(t)$ is normalized, i.e., its DC component is 1:

$$\phi(0) = \int_{-\infty}^{\infty} \phi(t)e^{-j2\pi 0t}dt = 1 \tag{10.27}$$

The summation index in the discussion above always takes values in the set of integers, e.g., $k \in \mathbb{Z} = \{-\infty, \cdots, -1, 0, 1, \cdots, \infty\}$. In the following, for simplicity, we will only indicate the summation index without explicitly specifying the range of values it takes.

Example 10.1: Consider a square impulse function defined as:

$$\phi(t) = \begin{cases} 1 \; 0 < t < 1 \\ 0 \text{ otherwise} \end{cases} \tag{10.28}$$

This function is orthogonal to itself translated by any integer amount:

$$< \phi(t), \phi(t-k) >= \int_{-\infty}^{\infty} \phi(t)\phi(t-k)dt = \delta[k], \qquad (k, l \in \mathbb{Z}) \tag{10.29}$$

Based on this 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 this space V_0 as:

$$x(t) \approx \sum_k c_k \phi_{0,k}(t) = \sum_k c_k \phi(t-k) \tag{10.30}$$

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:

$$\phi_{j,k}(t) = 2^{j/2}\phi(2^j t - k), \qquad k \in \mathbb{Z} \tag{10.31}$$

As $\phi(t) = 1$ when its argument satisfies $0 < t < 1$, we see that $\phi_{j,k}(t) = \phi(2^j t - k) = 1$ when its argument satisfies:

$$0 < 2^j t - k < 1, \quad \text{i.e.,} \quad \frac{k}{2^j} < t < \frac{k}{2^j} + \frac{1}{2^j} \tag{10.32}$$

i.e., $\phi_{j,k}(t) = \phi(2^j t - k)$ is a square impulse of height $\sqrt{2^j}$ 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:

$$< \phi_{j,k}(t), \phi_{j,l}(t) >= \delta[k-l] \tag{10.33}$$

The basic ideas above are illustrated in Fig.10.2. The first two panels show two scaling functions $\phi(t) = \phi_{0,0}(t)$ and $\phi_{0,1}(t) = \phi(t-1)$ both in space 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 space V_1. Panel 5 shows a function $x(t) \in V_1$ represented as a linear combination of the scaling functions $\phi_{1,k}(t)$:

$$x(t) = 0.5\phi_{1,0}(t) + \phi_{1,1}(t) - 0.25\phi_{1,4}(t) \tag{10.34}$$

Finally panel 6 shows that a scaling function $\phi_{0,0}(t)$ in V_0 can also be represented as a linear combination of the basis functions $\phi_{1,k}(t)$ in V_1 (Eq.10.23):

$$\phi_{0,l}(t) = h_0[0]\phi_{1,2l}(t) + h_0[1]\phi_{1,2l+1}(t) = \frac{1}{\sqrt{2}}\phi_{1,2k}(t) + \frac{1}{\sqrt{2}}\phi_{1,2k+1}(t) \tag{10.35}$$

where the coefficients $h_0[0] = h_0[1] = 1/\sqrt{2}$ are obtained according to Eq.10.21.

The ideas illustrated in this example are still valid if the square impulses are replaced by any family of functions with finite support (with non-zero function values over a finite time duration).

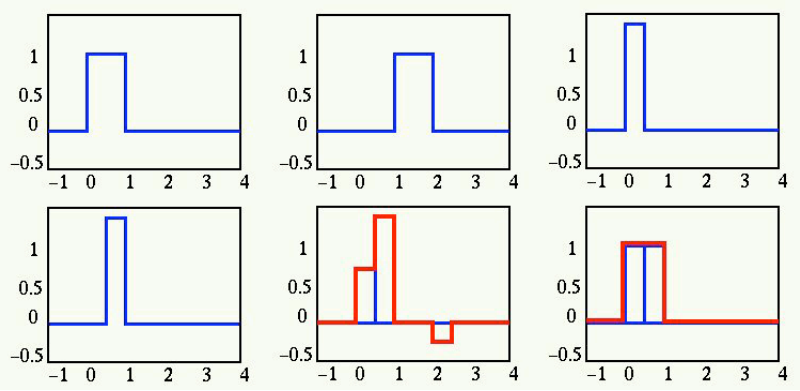

Figure 10.2 Square impulses used as scaling functions

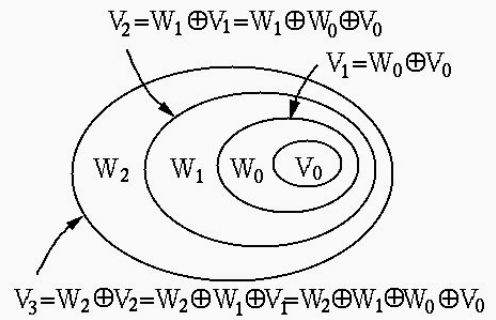

Figure 10.3 The nested V_j and W_j spaces for MRA

10.1.2 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 levels of details, 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} are not representable in V_j. All such functions are contained in the difference space between V_{j+1} and V_j, define the *wavelet space* W_j. 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:

$$V_{j+1} = W_j \oplus V_j = W_j \oplus W_{j-1} \oplus V_{j-1} = \cdots \tag{10.36}$$

This relationship can be further expanded recursively for all $j \in \mathbb{Z}$, and according to Eq.10.3, have:

$$L^2(\mathbb{R}) = \oplus_{j\in\mathbb{Z}} W_j \tag{10.37}$$

This result indicates that the L^2 space is the direct sum of all the wavelet spaces W_j. In other words, any function $x(t) \in L^2(\mathbb{R})$ can be considered as a combination of infinitely many approximations of different levels of details.

Same as a scale space V_j which is spanned by a set of orthogonal scaling functions $\phi_{j,k}(t)$, a wavelet space W_j can also be spanned by a set of orthogonal *wavelet functions* $\psi_{j,k}(t)$, defined similarly to the scale functions $\phi_{j,k}(t)$. These wavelet functions are further required to be orthogonal to the scaling functions, i.e., the following should hold for all integer shifts $k, l \in \mathbb{Z}$ at all levels $j \in \mathbb{Z}$:

$$< \psi_{j,k}(t), \psi_{j,0}(t) >= \delta[k] \tag{10.38}$$

$$< \phi_{j,k}(t), \psi_{j,l}(t) >= 0 \tag{10.39}$$

Specifically at scale level $j = 0$, the orthogonalities above can be written as:

$$\begin{aligned} < \phi(t-k), \psi(t) >&= \int_{-\infty}^{\infty} \phi(t-k)\overline{\psi}(t)dt = \delta[k] \\ < \phi(t-k), \psi(t) >&= \int_{-\infty}^{\infty} \phi(t-k)\overline{\psi}(t)dt = 0 \end{aligned} \tag{10.40}$$

Following the same process for the derivation of Eq.10.15 from Eq.10.12 for the orthogonality of the scaling functions, we can also represent these required orthogonalities for the wavelet functions in frequency domain:

$$\begin{aligned} \sum_k |\Psi(f-k)|^2 &= 1 \\ \sum_k \Phi(f-k)\overline{\Psi}(f-k) &= 0 \end{aligned} \tag{10.41}$$

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 $W_j \perp W_{j-1}$ and $W_j \perp V_{j-1}$ for all $j \in \mathbb{Z}$. The fact that $W_j \perp W_{j-1}$ also indicates that 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. Note that in contrast, the scaling functions $\phi_{j,k}(t)$ are not orthogonal across different scale levels. Further more, since all wavelet spaces W_j are spanned by $\psi_{j,k}(t)$, the entire function space $L^2(\mathbb{R}) = \oplus_j W_j$, as the direct sum of these wavelet spaces, is also spanned by these orthogonal wavelet functions:

$$L^2(\mathbb{R}) = span(\psi_{j,k}(t),\ (j,k \in \mathbb{Z}) \tag{10.42}$$

In the following we will construct such wavelet functions that satisfy Eq.10.39.

We first consider the case when $j = 0$ and how corresponding wavelet function $\psi(t) = \psi_{0,0}(t)$, called *mother wavelet*, is related to the scaling functions. Similar to the representation of the father wavelet $\phi(t) \in V_1$ in Eq.10.20, the mother wavelet $\psi(t) \in V_1$ can also be expressed as a linear combination of the basis $\phi_{1,k}(t) = \sqrt{2}\phi(2t-k)$ of V_1:

$$\psi(t) = \sqrt{2}\sum_k h_1[k]\phi_{1,k}(t) = \sqrt{2}\sum_k h_1[k]\phi(2t-k) \tag{10.43}$$

The coefficients $h_1[k]$ are obviously different from but certainly related to the coefficients $h_0[k]$ for $\phi(t)$, for the wavelet functions $\psi(t)$ to be orthogonal to the

scaling functions $\phi(t)$, as to be discussed later. We next replace t by $2^j t - l$ in the equation above to get

$$\psi(2^j t - l) = \sqrt{2}\sum_k h_1[k]\phi(2(2^j t - l) - k) = \sqrt{2}\sum_k h_1[k]\phi(2^{j+1}t - (2l + k))$$
$$= \sqrt{2}\sum_{k'} h_1[k' - 2l]\phi(2^{j+1}t - k') \tag{10.44}$$

But due to Eq.10.18,

$$\phi(2^{j+1}t - k) = 2^{-(j+1)/2}\phi_{j+1,k}(t) \tag{10.45}$$

the above equation becomes:

$$\psi_{j,l}(t) = 2^{j/2}\psi(2^j t - l) = \sum_k h_1[k - 2l]\phi_{j+1,k}(t) \tag{10.46}$$

which defines the wavelet functions $\psi_{j,l}(t)$ that span W_j.

Eq.10.43 above for the wavelet function can be equivalently represented in frequency domain, similar to Eq.10.24 for the scaling functions:

$$\Psi(f) = \mathcal{F}[\psi(t)] = \sqrt{2}\sum_k h_1[k]\mathcal{F}[\phi(2t - k)]$$
$$= \frac{1}{\sqrt{2}}\sum_k h_1[k]e^{-jk\pi}\Phi(\frac{f}{2}) = \frac{1}{\sqrt{2}}H_1(\frac{f}{2})\Phi(\frac{f}{2}) \tag{10.47}$$

where $H_1(f)$ is the discrete-time Fourier transform for the *wavelet filter*:

$$H_1(f) = \sum_k h_1[k]e^{-j2k\pi f} \tag{10.48}$$

Note that $H_1(f \pm 1) = H_1(f)$ is periodic with period 1. Again, same as in Eq.10.26, the wavelet filter can also be recursively expanded to become:

$$\Psi(f) = \frac{1}{\sqrt{2}}H_1(\frac{f}{2})\prod_{j=2}^{\infty}\frac{1}{\sqrt{2}}H_0(\frac{f}{2^j}) \tag{10.49}$$

Also recall that in order to satisfy the admissibility condition (Eq.9.21), the integral of the wavelet $\psi(t)$ needs to be zero (Eq.9.10), i.e., its DC component should be zero:

$$\Psi(0) = \int_{-\infty}^{\infty}\psi(t)e^{-j2\pi 0t}dt = \int_{-\infty}^{\infty}\psi(t)dt = 0 \tag{10.50}$$

For the wavelet functions to be orthonormal and also orthogonal to the scaling functions as required, they obviously need to satisfy certain conditions in terms of the coefficients $h_1[k]$ or equivalently the wavelet filter $H_1(f)$. Now we consider the how to construct the wavelet functions that satisfy the required orthogonalities. To do so, we first prove the theorem below, and then construct a wavelet function accordingly.

Theorem 10.1. *The wavelet functions $\psi(t)$ defined in Eq.10.43 are orthogonal to the scaling functions $\phi(t-k)$, i.e., Eqs.10.40 and 10.41 hold, if and only if the wavelet filter $H_1(f)$ is related to the scaling filter $H_0(f)$ by the following:*

$$H_0(f)\overline{H}_1(f) + H_0(f-\frac{1}{2})\overline{H}_1(f-\frac{1}{2}) = 0 \tag{10.51}$$

Note that as $H_i(f \pm 1) = H_i(f)$ is periodic, $H_i(f-\frac{1}{2}) = H_i(f+\frac{1}{2})$ for $i = 0, 1$.

Proof:

Substituting Eqs.10.24 and 10.47 into Eq.10.41, we get:

$$\begin{aligned}&\sum_k H_0(\frac{f-k}{2})\phi(\frac{f-k}{2})\overline{H}_1(\frac{f-k}{2})\overline{\phi}(\frac{f-k}{2}) \\ &= \sum_k H_0(\frac{f-k}{2})\overline{H}_1(\frac{f-k}{2})\left|\phi(\frac{f-k}{2})\right|^2 = 0\end{aligned} \tag{10.52}$$

Separating the even and odd terms in the summation we rewrite the above as:

$$\begin{aligned}&\sum_k H_0(\frac{f-2k}{2})\overline{H}_1(\frac{f-2k}{2})\left|\phi(\frac{f-2k}{2})\right|^2 \\ &+ \sum_k H_0(\frac{f-(2k+1)}{2})\overline{H}_1(\frac{f-(2k+1)}{2})\left|\phi(\frac{f-(2k+1)}{2})\right|^2 = 0\end{aligned} \tag{10.53}$$

We replace $f/2$ by f' and recall that $H_0(f)$ and $H_1(f)$ have period 1 to get

$$\begin{aligned}&H_0(f')\overline{H}_1(f')\sum_k |\Phi(f'-k)|^2 \\ &+ H_0(f'-\frac{1}{2})\overline{H}_1(f'-\frac{1}{2})\sum_k \left|\Phi(f'-k-\frac{1}{2})\right|^2 = 0\end{aligned} \tag{10.54}$$

The proof is complete by realizing that both summations are equal to 1 (Eq.10.15).

Next we show that the condition in Eq.10.51 is satisfied by the wavelet filter $H_1(f)$ constructed below:

$$H_1(f) = -e^{-j2\pi f}\,\overline{H}_0(f-\frac{1}{2}) \tag{10.55}$$

Substituting this $H_1(f)$ into the left-hand side of Eq.10.51 we can easily verify that this equation indeed holds:

$$\begin{aligned}&-H_0(f)e^{j2\pi f}H_0(f-\frac{1}{2}) - H_0(f-\frac{1}{2})e^{j2\pi(f+1/2)}H_0(f-1) \\ &= -H_0(f)e^{j2\pi f}H_0(f-\frac{1}{2}) + H_0(f-\frac{1}{2})e^{j2\pi f}H_0(f) = 0\end{aligned} \tag{10.56}$$

The time domain filter coefficients $h_1[n]$ can be obtained as the inverse Fourier transform of $H_1(f)$:

$$h_1[k] = \mathcal{F}^{-1}[-e^{-j2\pi f}\ \overline{H}_0(f - \frac{1}{2})] = (-1)^k\ \overline{h}_0[1-k] \tag{10.57}$$

Substituting these coefficients into Eq.10.43, we can construct the wavelet functions that are indeed orthogonal to the scaling functions $\phi(t-k)$:

$$\psi(t) = \sqrt{2}\sum_k h_1[k]\phi(2t-k) = \sqrt{2}\sum_k (-1)^k\ \overline{h}_0[1-k]\phi(2t-k) \tag{10.58}$$

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.

Theorem 10.2. *The wavelet function $\psi(t) \in V_0$ defined in Eq.10.58 are orthogonal to its integer translations $\psi(t-l)$ for all $l \in \mathbb{Z}$:*

$$< \psi(t-l), \psi(t) >= \int_{-\infty}^{\infty} \phi(t-l)\overline{\phi}(t)dt = \delta[l] \tag{10.59}$$

Proof:
Substituting 10.58 into Eq.10.59, we have

$$\begin{aligned}
&< \psi(t-l), \psi(t) >= \int_{-\infty}^{\infty} \psi(t-l)\overline{\psi}(t)dt \\
&= 2\sum_{k'}\sum_{k}(-1)^{k+k'}\overline{h}_0[1-k]h_0[1-k']\int_{-\infty}^{\infty}\phi(2(t-l)-k')\overline{\phi}(2t-k)dt \\
&= 2\sum_{k}\sum_{m}(-1)^{m+k}\overline{h}_0[1-k]h_0[1-m+2l]\int_{-\infty}^{\infty}\phi(2t-m)\overline{\phi}(2t-k)dt \\
&= \sum_{k}\sum_{m}(-1)^{m+k}\overline{h}_0[1-k]h_0[1-m+2l]\delta[m-k] \\
&= \sum_{k}\overline{h}_0[1-k]h_0[1-k+2l] = \delta[l]
\end{aligned} \tag{10.60}$$

Here we have assumed $m = 2l + k'$ and used the fact that $\phi_{1,k}(t)$ are orthonormal (Eq.10.17). The last equal sign is due to a property of the coefficients $h_0[k]$, to be proven below (Eq.10.66)

Example 10.2: 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 the coefficients for the wavelet functions $\psi_{1,k}(t)$ can be obtained as

$$\begin{aligned}
h_1[0] &= (-1)^0 h_0[1-0] = h_0[0] = 1/\sqrt{2} \\
h_1[1] &= (-1)^1 h_0[1-1] = -h_0[0] = -1/\sqrt{2}
\end{aligned} \tag{10.61}$$

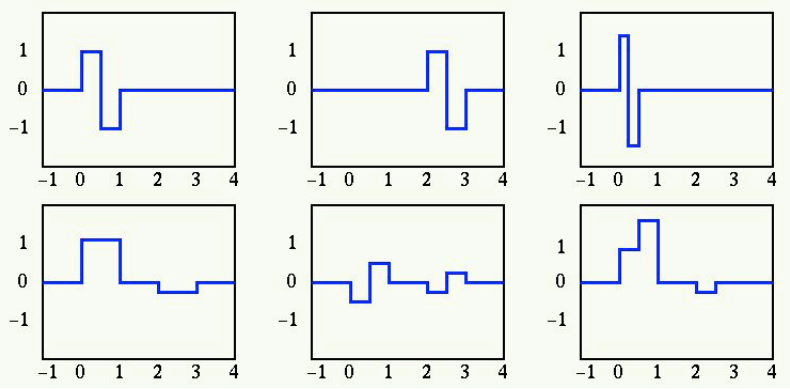

Figure 10.4 Square impulses used as wavelet functions

and the wavelet function is:

$$\psi(t) = \sum_l h_1[l]\sqrt{2}\phi[2t-l] = \frac{1}{\sqrt{2}}\sqrt{2}\phi(2t) - \frac{1}{\sqrt{2}}\sqrt{2}\phi(2t-1) = \begin{cases} 1 & 0 \le t < 1/2 \\ -1 & 1/2 \le t < 1 \\ 0 & \text{otherwise} \end{cases} \tag{10.62}$$

The first two panels of Fig.10.4 show two of the wavelet functions $\psi(t) = \psi_{0,0}(t)$ and $\psi_{0,2}(t) = \psi(t-2)$ in space W_0. Note that $\phi_{1,k}(t) = \sqrt{2}\phi_{0,k}(2t)$ can be generated by the linear combination of $\phi_{0,k}(t)$ and $\psi_{0,k}(t)$:

$$\phi_{1,k}(t) = \frac{\sqrt{2}}{2}[\phi_{0,k}(t) + \psi_{0,k}(t)] \tag{10.63}$$

The 3rd panel shows a wavelet function $\psi_{1,0}(t) = \sqrt{2}\psi(2t)$ in space W_1. The 4th panel shows a function in space V_0, while the 5th panel shows a function in space W_0. Finally the 6th panel shows a function in space $V_1 = V_0 \oplus W_0$, which can be written as a linear combination of $\phi_{1,k}(t)$, or, equivalently, of $\phi_{0,k}(t)$ and $\psi_{0,k}(t)$.

This example together with the one in previous section illustrate that the Haar transform as discussed in Chapter 6 is actually a wavelet transform. For example, when $N = 2$, as shown in Fig.7.8 and Eq.7.74, the first row contains the scaling coefficients $h_0[k]$, the second row contains the wavelet coefficients $h_1[k]$.

10.1.3 Properties of the scaling and wavelet filters

We now consider a set of important properties for both the scaling filter $h_0[k]$ or $H_0(f)$ and the wavelet filter $h_1[k]$ or $H_1(f)$, which will be of great importance for the wavelet filter design to be discussed in future.

1. Normalization:

$$\sum_k h_0[2k] = \sum_{k\in\mathbb{Z}} h_0[2k+1] = 1 \tag{10.64}$$

We integrate both sides of Eq.10.20 with respect to t to get

$$\int_{-\infty}^{\infty} \phi(t)dt = \sqrt{2}\sum_k h_0[k] \int_{-\infty}^{\infty} \phi(2t-k)dt = \sum_k h_0[k]\frac{1}{\sqrt{2}} \int_{-\infty}^{\infty} \phi(t')dt' \tag{10.65}$$

where we have assumed $t' = 2t - k$, i.e.,$t = (t' - k)/2$. Dividing both sides by $\int_{-\infty}^{\infty} \phi(t)dt = 0$, we get the second equation.

2. Shift-Orthonormality:
   The scaling and wavelet filters are orthogonal to themselves translated by any even number of positions:

$$\sum_k h_0[k]\overline{h}_0[k-2n] = \delta[n]$$
$$\sum_k h_1[k]\overline{h}_1[k-2n] = \delta[n] \tag{10.66}$$

   In particular, when $n = 0$, we have

$$\sum_k |h_0[k]|^2 = 1, \quad \sum_k |h_1[k]|^2 = 1 \tag{10.67}$$

   Proof: Substituting Eq.10.23 into Eq.10.17 (and replacing k by l), we get

$$\begin{aligned}\delta[l] = <\phi_{j,l}(t), \phi_{j,0}(t)> &= \int_{-\infty}^{\infty} \phi_{j,l}(t)\overline{\phi}_{j,0}(t)dt \\ &= \sum_k \sum_{k'} h_0[k-2l]\overline{h}_0[k'] \int_{-\infty}^{\infty} \phi_{j+1,k}(t)\overline{\phi}_{j+1,k'}(t)dt \\ &= \sum_k \sum_{k'} h_0[k-2l]\overline{h}_0[k]\delta[k-k'] = \sum_k h_0[k-2l]\overline{h}_0[k]\end{aligned} \tag{10.68}$$

   The proof for $\sum_k |h_1[k]|^2 = 1$ is identical.

3. Normalization in frequency domain:

$$H_1(0) = 0, \quad H_0(0) = \sqrt{2} \tag{10.69}$$

   These can easily obtained by letting $f = 0$ in Eqs.10.24 and 10.47, and noting $\phi(0) = 1$ (Eq.10.27) and $\Psi(0) = 0$ (Eq.10.50). Equivalently, we have

$$\sum_k h_1[k] = 0, \quad \sum_k h_0[k] = \sqrt{2} \tag{10.70}$$

   which can be also be easily shown by letting $f = 0$ in Eqs.10.25 and 10.48 and applying the results $H_0(0) = \sqrt{2}$ and $H_1(0) = 0$ above.

4. Shift-Orthogonalities in frequency domain:

$$|H_0(f)|^2 + |H_0(f+\frac{1}{2})|^2 = 2$$
$$|H_1(f)|^2 + |H_1(f+\frac{1}{2})|^2 = 2$$
$$H_0(f)\overline{H}_1(f) + H_0(f+\frac{1}{2})\overline{H}_1(f+\frac{1}{2}) = 0 \tag{10.71}$$

Proof:
Substituting Eq.10.24 into Eq.10.15, we get

$$\sum_k \left|H_0(\frac{f-k}{2})\right|^2 \left|\phi(\frac{f-k}{2})\right|^2 = 2 \tag{10.72}$$

We separate the even and odd terms in the summation on the left-hand side to get:

$$\sum_k \left|H_0(\frac{f-2k}{2})\right|^2 \left|\phi(\frac{f-2k}{2})\right|^2 + \sum_k \left|H_0(\frac{f-(2k+1)}{2})\right|^2 \left|\phi(\frac{f-(2k+1)}{2})\right|^2 = 2 \tag{10.73}$$

But as $H_0(f \pm 1) = H_0(f)$ is periodic, the above can be written as

$$\left|H_0(\frac{f}{2})\right|^2 \sum_k \left|\phi(\frac{f}{2}-k)\right|^2 + \left|H_0(\frac{f+1}{2})\right|^2 \sum_k \left|\phi(\frac{f+1}{2}-k)\right|^2$$
$$= \left|H_0(\frac{f}{2})\right|^2 + \left|H_0(\frac{f}{2}+\frac{1}{2})\right|^2 = 1 \tag{10.74}$$

The last equal sign is due to Eq.10.15. Replacing $f/2$ by f, we complete the proof. The relation for $H_1(f)$ can be proven in the same way. The third equation relating $H_0(f)$ and $H_1(f)$ is Eq.10.51 already proven above.

Example 10.3: Here we verify that all properties above are satisfied by the scaling and wavelet filters of Haar transform as illustrated in the previous two examples. Recall that the scaling and wavelet coefficients are $h_0[0] = h_0[1] = 1/\sqrt{2}$ and $h_1[0] = 1/sqrt2$, $h_1[1] = -1/\sqrt{2}$, with all other $h_0[k] = h_1[k] = 0$ for $k \neq 0, 1$. We can see immediately that $\sum_k h_0[k] = h_0[0] + h_0[1] = 2/\sqrt{2} = \sqrt{2}$ and $\sum_k h_1[k] = h_1[0] + h_1[1] = 0$. Also, he orthogonality in time domain is obvious. Next we find the DTFT spectra of $h_0[k]$ and $h_1[k]$:

$$H_0(f) = \sum_k h_0[k]e^{-j2k\pi f} = \frac{1}{\sqrt{2}}(1 + e^{-j2\pi f})$$
$$H_1(f) = \sum_k h_1[k]e^{-j2k\pi f} = \frac{1}{\sqrt{2}}(1 - e^{-j2\pi f})$$

We obviously have

$$H_0(0) = 1/\sqrt{2}, \qquad H_1(0) = 0$$

and

$$|H_0(f)|^2 + |H_0(f+\frac{1}{2})|^2 = [1+\cos(2\pi f)] + [1-\cos(2\pi f)] = 2$$

$$|H_1(f)|^2 + |H_1(f+\frac{1}{2})|^2 = [1-\cos(2\pi f)] + [1+\cos(2\pi f)] = 2$$

$$H_0(f)\overline{H}_1(f) + H_0(f+\frac{1}{2})\overline{H}_1(f+\frac{1}{2}) = 0$$

Moreover, we can find $\Phi(f)$ and $\Psi(f)$ of $\phi(t)$ and $\psi(t)$ respectively (Eqs.10.28 and 10.62):

$$\begin{aligned}\Phi(f) &= \int_{-\infty}^{\infty}\phi(t)e^{-j2\pi ft}dt = \int_0^1 e^{-j2\pi ft}dt = \frac{1}{-j2\pi f}(1-e^{-j2\pi f})\\ \Psi(f) &= \int_{-\infty}^{\infty}\psi(t)e^{-j2\pi ft}dt = \int_0^{1/2} e^{-j2\pi ft}dt + \int_{1/2}^1 e^{-j2\pi ft}dt\\ &= \frac{1}{-j2\pi f}(1-2e^{-j\pi f}+e^{-j2\pi f}) = \frac{1}{-j2\pi f}(1-e^{-j\pi f})^2\end{aligned}$$

and further verify that Eqs.10.24 and 10.47 hold:

$$\frac{1}{\sqrt{2}}H_0(\frac{f}{2})\Phi(\frac{f}{2}) = \frac{1}{\sqrt{2}}\frac{1}{\sqrt{2}}(1+e^{-j\pi f})\frac{1}{-j\pi f}(1-e^{-j\pi f}) = \frac{1}{-j2\pi f}(1-e^{-j2\pi f}) = \Phi(f)$$

and

$$\frac{1}{\sqrt{2}}H_1(\frac{f}{2})\Phi(\frac{f}{2}) = \frac{1}{\sqrt{2}}\frac{1}{\sqrt{2}}(1-e^{-j\pi f})\frac{1}{-j\pi f}(1-e^{-j\pi f}) = \frac{1}{-j2\pi f}(1-e^{-j\pi f})^2 = \Psi(f)$$

10.1.4 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 done in one of three different ways:

- Specify directly $\phi(t)$ and $\psi(t)$;
- Specify their spectra $\Phi(f)$ and $\Psi(f)$ in frequency domain;
- Specify their corresponding filter coefficients $h_0[k]$ and $h_1[k]$.

Ideally our goal is to find the scaling and wavelet functions with good locality in both time and frequency domains. In the following we will consider these three different methods for the construction the of scaling and wavelet functions, each illustrated by one example.

- **Haar wavelets**
  Based on the discussions above, we can now specifically construct the scaling and wavelet functions following the steps below:

1. Choose the scaling function $\phi(t)$ satisfying Eq.10.12:

$$< \phi(t-k), \phi(t) >= \delta[k] \tag{10.75}$$

or $\Phi(f)$ satisfying Eq.10.15:

$$\sum_k |\Phi(f-k)|^2 = 1 \tag{10.76}$$

For Haar transform, the scaling function is:

$$\phi(t) = \begin{cases} 1 \; 0 \leq t < 1 \\ 0 \text{ otherwise} \end{cases} \tag{10.77}$$

2. Find scaling coefficients $h_0[k]$ based on Eq.10.21:

$$h_0[k] =< \phi(t), \sqrt{2}\phi(2t-k) > \tag{10.78}$$

or $H_0(f)$ according to Eq.10.24:

$$H_0(f) = \sqrt{2}\,\frac{\Phi(2f)}{\Phi(f)} \tag{10.79}$$

For Haar transform, we have

$$\begin{aligned} h_0[k] &= \sqrt{2}\int_{-\infty}^{\infty} \phi(t)\phi(2t-k)dt = \sqrt{2}\int_0^1 \phi(2t-k)dt \\ &= \frac{1}{\sqrt{2}}\int_0^2 \phi(t'-k)dt' = \frac{1}{\sqrt{2}} \begin{cases} 1 \; k = 0, 1 \\ 0 \text{ otherwise} \end{cases} \end{aligned} \tag{10.80}$$

3. Find wavelet coefficients $h_1[k]$ according to Eq.10.57

$$h_1[k] = (-1)^k\, \overline{h}_0[1-k] \tag{10.81}$$

or $H_1(f)$ according to Eq.10.55

$$H_1(f) = -e^{-j2\pi f}\, \overline{H}_0(f - \frac{1}{2}) \tag{10.82}$$

For Haar transform, we have:

$$h_1[k] = (-1)^k h_0[1-k] = \frac{1}{\sqrt{2}} \begin{cases} 1 \; k = 0 \\ -1 \; k = 1 \\ 0 \text{ otherwise} \end{cases} \tag{10.83}$$

4. Find wavelet function $\psi(t)$ according to Eq.10.58

$$\psi(t) = \sqrt{2}\sum_k (-1)^k\, \overline{h}_0[1-k]\phi(2t-k) \tag{10.84}$$

or $\Psi(f)$ according to Eq.10.47

$$\Psi(f) = H_1(\frac{f}{2})\Psi(\frac{f}{2}) \tag{10.85}$$

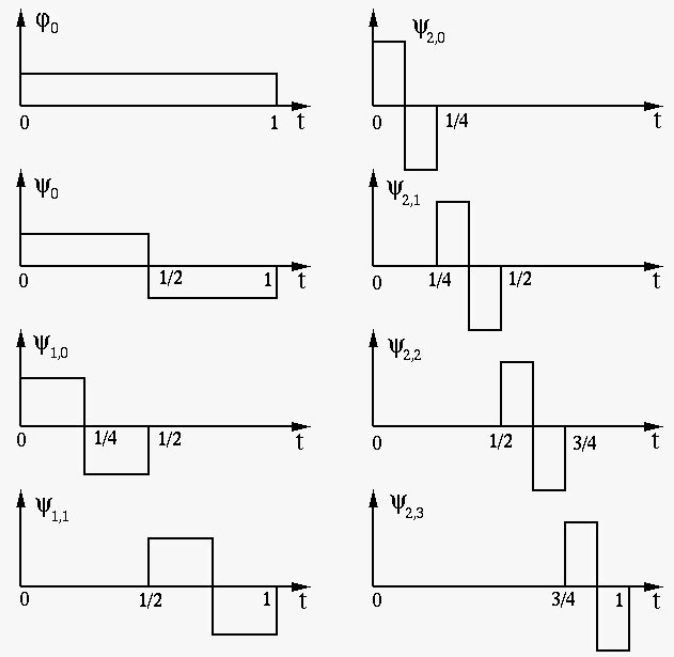

Figure 10.5 Haar scaling and wavelet functions

For Haar transform, we have:

$$\psi(t) = h_1[0]\phi_{1,0}(t) + h_1[1]\phi_{1,1}(t) = \begin{cases} 1 \; 0 \le t < 1/2 \\ -1 \; /2 \le t < 1 \\ 0 \text{ otherwise} \end{cases} \tag{10.86}$$

Based on $\phi(0) = \phi_{0,0}(t)$ and $\psi(0) = \psi_{0,0}(t)$, all other $\psi_{j,k}(t)$ can be obtained. 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 will try to construct a wavelet that has good locality in both time and frequency domains. To do so, we need to avoid sharp discontinuities in both domains. This time we start in frequency domain by considering the spectrum $\Phi(f)$ of the scaling function $\phi(t)$. We will first define a function for a smooth transition from 0 to 1 and then use it to define a smooth frequency window. While there exist many different functions with a smooth transition between 0 and 1, we here consider a 3rd order polynomial function with a

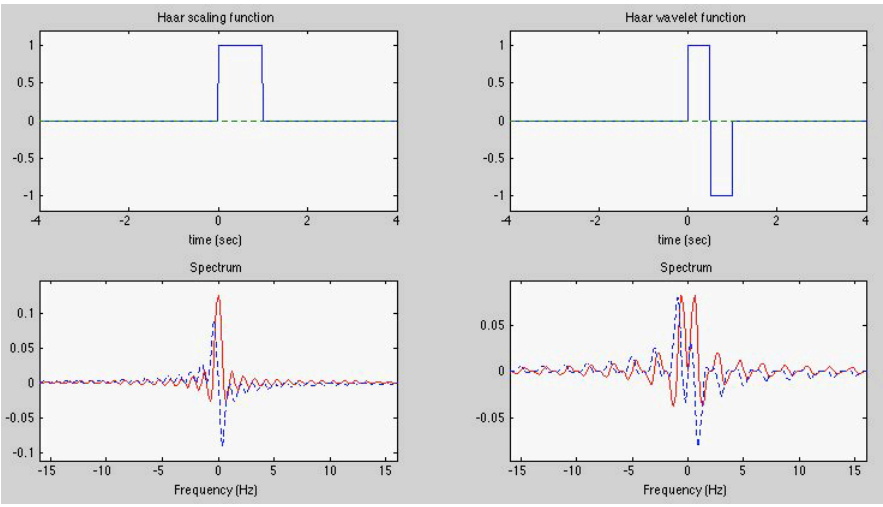

Figure 10.6 Haar scaling and wavelet functions and their spectra

smooth transition between 0 and 1 (Fig.10.7(a)):

$$\nu(x) = \begin{cases} 0 & x < 0 \\ 3x^2 - 2x^3 & 0 \leq x \leq 1 \\ 1 & x > 1 \end{cases} \tag{10.87}$$

The coefficients are chosen so that $\nu(1/2) = 1/2$ and $\nu(x) + \mu(1-x) = 1$, a property needed for the orthogonality requirement Eq.10.15 to be satisfied by the corresponding spectrum $\Phi(f)$ defined below:

$$\Phi(f) = \begin{cases} \sqrt{\nu(2+3f)} & f \leq 0 \\ \sqrt{\nu(2-3f)} & f \geq 0 \end{cases} \tag{10.88}$$

As shown in Fig.10.7(b), $\Phi^2(f) = 1$ when $|f| \leq 1/3$, $\Phi^2(f) = 0$ when $2/3 \leq |f| < 1$, and $\phi^2(f) + \phi^2(f \pm 1) = 1$ when $1/3 < |f| < 2/3$ during the transition interval where the two neighboring copies of $\Phi(f)$ overlap, i.e., Eq.10.15 is indeed satisfied. Note that $\Phi(f)$ is non-zero only for $|f| < 2/3$

Having obtained $\Phi(f)$, we will next find the scaling filter $H_0(f)$ based on $H_0(f) = \sqrt{2}\Phi(2f)/\Phi(f)$ (Eq.10.24). As shown in Fig.10.7(c), $\Phi(2f)$, a compressed version of $\Phi(f)$, is zero for all f except for $|f| < 1/3$, therefore $\sqrt{2}\Phi(2f)/\Phi(f)$ is also zero for all f except when $|f| < 1/3$, during which interval $\Phi(2f)/\Phi(f) = \Phi(2f)$ as $\Phi(f) = 1$. Also, as the scaling filter $H_0(f)$ is periodic $H_0(f \pm 1) = H_0(f)$, we can write the scaling filter as:

$$H_0(f) = \sum_k \Phi(2(f-k)) = \sum_k \Phi(2f - 2k) \tag{10.89}$$

Given $H_0(f)$, we can find the wavelet filter $H_1(f)$ based on Eq.10.55:

$$H_1(f) = -e^{-j2\pi f}\overline{H}_0(f-\frac{1}{2}) = -e^{-j2\pi f}\sum_k \Phi(2f-2k-1) \tag{10.90}$$

and then the spectrum $\Psi(f)$ of the wavelet function $\psi(t)$ based on Eq.10.47:

$$\begin{aligned}\Psi(f) &= \frac{1}{\sqrt{2}}H_1(\frac{f}{2})\Phi(\frac{f}{2}) = \frac{1}{\sqrt{2}}e^{-j\pi f}H_0(\frac{f-1}{2})\Phi(\frac{f}{2}) \\ &= \frac{1}{\sqrt{2}}e^{-j\pi f}\sum_k \Phi(f-2k-1)\Phi(\frac{f}{2})\end{aligned} \tag{10.91}$$

As is shown in Fig.10.7, $\Psi(f)$ can be written as:

$$\Psi(f) = \begin{cases} 0 & |f| < 1/3 \\ -\frac{1}{\sqrt{2}}e^{-j2\pi f}\Phi(f-1) & 1/3 < |f| < 2/3 \\ -\frac{1}{\sqrt{2}}e^{-j2\pi f}\Phi(f/2) & 2/3 < |f| < 4/3 \\ 0 & |f| > 4/3 \end{cases} \tag{10.92}$$

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.10.8, and the scaling filter coefficients can be found as:

$$h_0[k] = \int_0^1 H_0(f)e^{-j2\pi kf}df, \qquad k \in \mathbb{Z} \tag{10.93}$$

We see that in the case, there may exist infinite number of coefficients $h_0[k]$.

- **Daubechies' wavelets**
  Another way to achieve better smoothness in time domain and locality is frequency domain is based on the following observation which is also illustrated in Fig.10.9. If we convolve the highly discontinuous rectangular function $x(t)$ with itself, a smoother triangular function $y(t) = x(t) * x(t)$ is obtained. In frequency domain, the spectrum of the rectangular function $X(f) = \mathbb{F}[x(t)]$, a sinc function, is raised to the 2nd power by the convolution to become $Y(f) = X^2(f)$ with higher frequency components attenuated. If we further convolve this triangular function with itself, a smoother still bell-shaped function $z(t) = y(t) * y(t) = x(t) * x(t) * x(t) * x(t)$ is obtained, corresponding to $X(f)$ in frequency domain raised to the 4th power $Z(f) = X^4(f)$, with higher frequency components further suppressed. We see that in general, if the spectrum of a scaling function is raised to a higher power, it becomes better localized in frequency domain and smoother in time domain, due to the attenuation of most higher frequency components. Now consider in particular the identity $\cos^2(\pi f) + \sin^2(\pi f) = 1$ raised to the 3rd power:

$$\begin{aligned}1 &= [\cos^2(\pi f) + \sin^2(\pi f)]^3 \\ &= \cos^6(\pi f) + 3\cos^4(\pi f)\sin^2(\pi f) + 3\cos^2(\pi f)\sin^4(\pi f) + \sin^6(\pi f) \\ &= \cos^6(\pi f) + 3\cos^4(\pi f)\sin^2(\pi f) + 3\sin^2(\pi f + \pi/2)\cos^4(\pi f + \pi/2) + \cos^6(\pi f + \pi/2)\end{aligned}$$

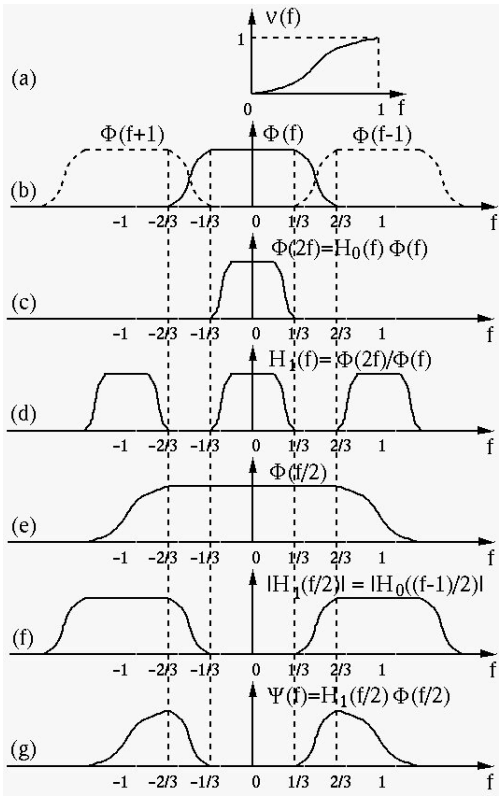

Figure 10.7 Construction of Meyer scaling and wavelet functions

The last equal sign is due to these identities:

$$\cos(\theta) = \sin(\theta + \pi/2), \qquad \sin(\theta) = -\cos(\theta + \pi/2)$$

We further define

$$|H_0(f)|^2 = 2[\cos^6(\pi f) + 3\cos^4(\pi f)\sin^2(\pi f)] \tag{10.94}$$

then we have $H_0(0) = \sqrt{2}$ and the above equation becomes:

$$|H_0(f)|^2 + |H_0(f + \frac{1}{2})|^2 = 2 \tag{10.95}$$

As this function $H_0(f)$ satisfies both the normalization and shift-orthogonality properties of a scaling filter given in Eq.10.69 and 10.71, it can indeed be used as a scaling filter, as the notation suggested. Now all we need is to find $H(f)$ by taking square root of $H^2(f)$. To do so, we rewrite its expression as:

$$\begin{aligned} |H_0(f)|^2 &= 2\cos^4(\pi f)[(\cos(\pi f))^2 + (\sqrt{3}\sin(\pi f))^2] \\ &= 2\cos^4(\pi f)\,|\cos^2(\pi f) + j\sqrt{3}\sin^2(\pi f)|^2 \end{aligned} \tag{10.96}$$

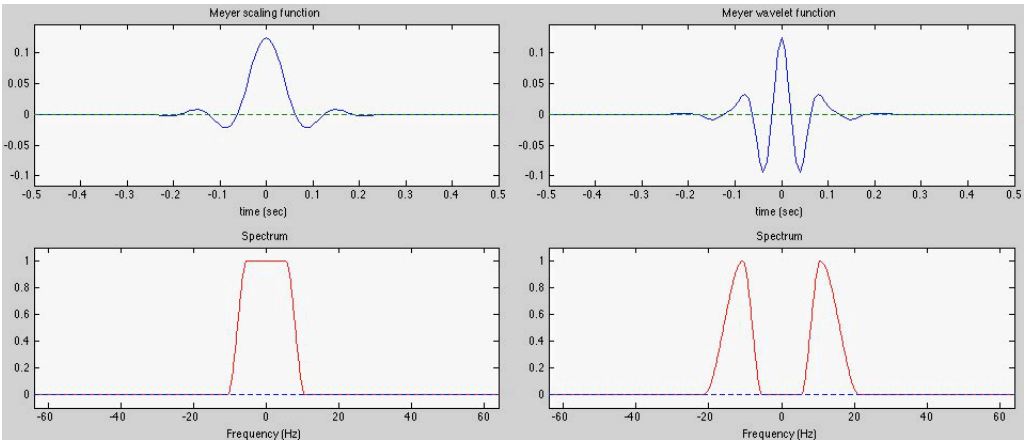

Figure 10.8 Meyer scaling and wavelet functions and their spectra

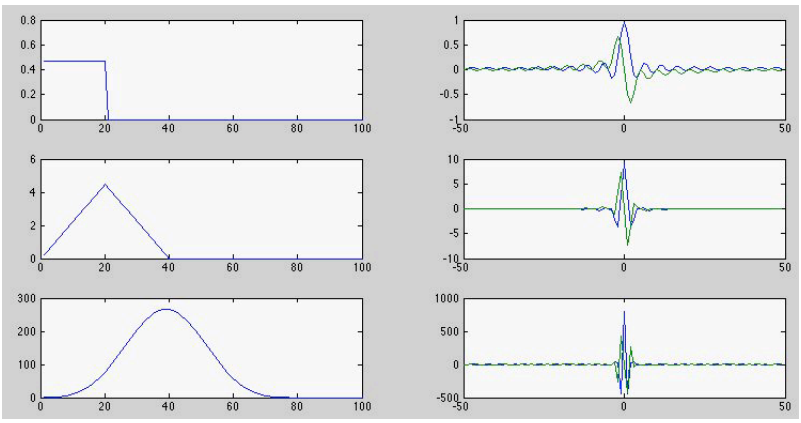

Figure 10.9 Getting smoother time function by attenuating higher frequency components

and get:

$$
\begin{aligned}
H_0(f) &= \sqrt{2}\cos^2(\pi f)[\cos^2(\pi f) + j3\sin^2(\pi f)] \\
&= \frac{1}{4\sqrt{2}}(e^{j2\pi f} + 2 + e^{-j\pi f})(e^{j\pi f} + e^{j-\pi f} + \sqrt{3}e^{j\pi f} - \sqrt{3}e^{-j\pi f}) \\
&= \frac{1}{\sqrt{2}}(\frac{1+\sqrt{3}}{4}e^{j3\pi f} + \frac{3+\sqrt{3}}{4}e^{j-2\pi f} + \frac{3-\sqrt{3}}{4}e^{-4j\pi f} + \frac{1-\sqrt{3}}{4}e^{-j6\pi f})e^{j3\pi f} \\
&= \left[\sum_{k=0}^{3} h_0[k]e^{-j2k\pi f}\right] e^{-j3\pi f} = \sum_{k=0}^{3} h_0[k]e^{j2k\pi f} \qquad (10.97)
\end{aligned}
$$

Note that we have dropped the exponential factor $e^{j3\pi f}$ as the value of $|H_0(f)|^2$ is not changed. Here we have obtained a set of four scaling coef-

ficients for Daubechies' 4-point wavelet transform:

$$h_0[0] = \frac{1+\sqrt{3}}{4\sqrt{2}} = \frac{0.683}{\sqrt{2}}, \quad h_0[1] = \frac{3+\sqrt{3}}{4\sqrt{2}} = \frac{1.183}{\sqrt{2}},$$
$$h_0[2] = \frac{3-\sqrt{3}}{4\sqrt{2}} = \frac{0.317}{\sqrt{2}}, \quad h_0[3] = \frac{1-\sqrt{3}}{4\sqrt{2}} = \frac{-0.183}{\sqrt{2}} \tag{10.98}$$

The corresponding wavelet coefficients can be obtained according to Eq.10.57 $h_1[k] = (-1)^k h_0[1-k]$ as:

$$h_1[1] = -h_0[0] = -\frac{0.683}{\sqrt{2}}, \quad h_1[0] = h_0[1] = \frac{1.183}{\sqrt{2}},$$
$$h_1[-1] = -h_0[2] = -\frac{0.317}{\sqrt{2}}, \quad h_1[-2] = h_0[3] = \frac{-0.183}{\sqrt{2}} \tag{10.99}$$

Next, both the scaling function $\phi(t)$ and wavelet function $\psi(f)$ can be obtained based on Eqs.10.26 and 10.49. Alternatively, $\phi(t)$ and $\psi(t)$ could also be obtained iteratively by Eqs.10.20 and 10.43 based on an initial Haar scaling function:

$$\phi(t) = \begin{cases} 1 \ 0 < t < 1 \\ 0 \text{ else where} \end{cases} \tag{10.100}$$

The approximated scaling and wavelet functions from the first six iterations are shown in Fig.10.10.

```
function daubechies
T=3;                    % time period in second
s=64;                   % sampling rate: s saples/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;
```

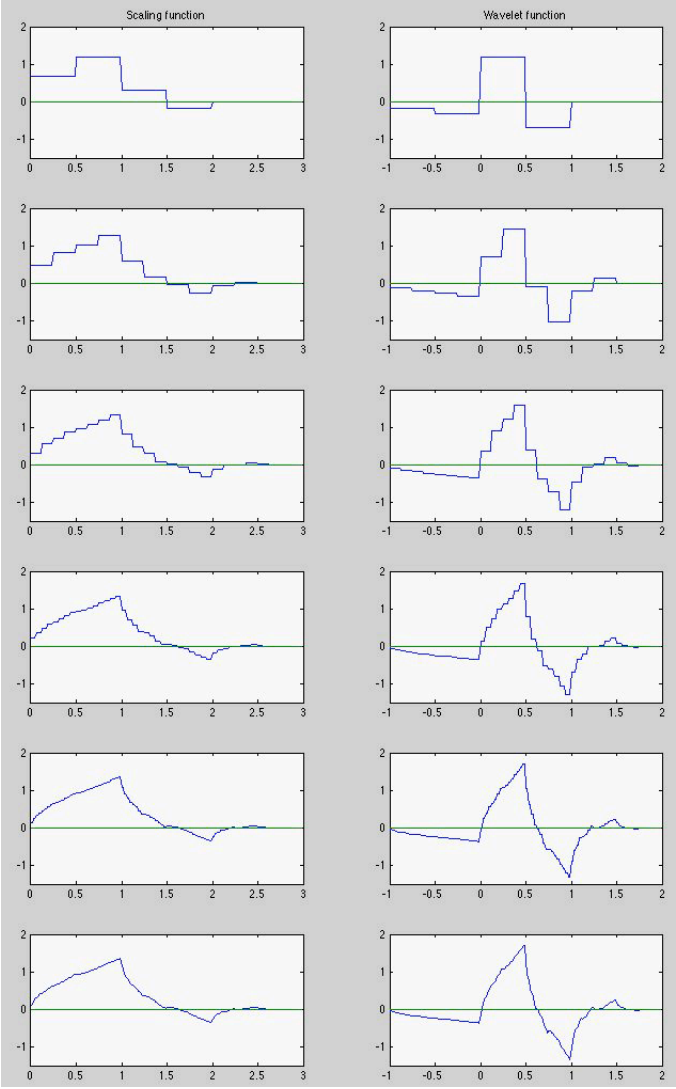

Figure 10.10 Iterative approximations of Daubechies' scaling and wavelet functions

```
            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')
```

10.2 Wavelet Series Expansion

As discussed at the beginning of the book in Eq.1.5, a given signal $x(t) \in L^2(\mathbb{R})$ can be approximated as a sequence of square impulse functions of unit height weighted by its discrete samples x_k. For simplicity we assume the sampling interval between two signal samples is $\triangle = 1$, and call the square impulse as the scaling function $\phi(t) \in V_0$, as defined in Eq.10.28, then the signal can be approximated as:

$$x(t) \approx \sum_k c[k]\phi_{0,k}\phi(t-k) \tag{10.101}$$

where $c[k]$ is the kth sample value of the amplitude of the signal. This expression is a linear combination of a set of standard basis functions $\phi_{0,k}$ $(k \in \mathbb{Z})$ that spans space V_0, i.e., it can be considered as an identity transform. Obviously this approximation can be improved if the more detailed information contained in W_0 is added, i.e., if the signal is approximated in space $V_1 = V_0 \oplus W_0$. Of course the approximation can be further improved in space $V_2 = V_0 \oplus W_0 \oplus W_1$ with still more detailed information in W_1 added. In general the approximation can be progressively refined if this process is repeated to include more and more wavelet spaces W_j, until $j \to \infty$, when $x(t)$ is precisely represented in $L^2(\mathbb{R})$. In this case, the signal can be written as a linear combination of the orthogonal basis functions $\phi_{0,k}(t)$ and $\psi_{j,k}(t)$ $(j,k \in \mathbb{Z})$ of $L^2(\mathbb{R})$:

$$x(t) = \sum_k c_{0,k}\phi_{0,k}(t) + \sum_{j=0}^{\infty}\sum_k d_{j,k}\psi_{j,k}(t) \tag{10.102}$$

where $c_{0,k}$ is the *approximation coefficient*:

$$c_{0,k} =< x(t), \phi_{0,k}(t) >= \int x(t)\overline{\phi}_{0,k}(t)dt, \quad \text{(for all } k\text{)} \tag{10.103}$$

and $d_{j,k}$ is the *detail coefficient*:

$$d_{j,k} =< x(t), \psi_{j,k}(t) >= \int x(t)\overline{\psi}_{j,k}(t)dt, \quad \text{(for all } k \text{ and } j > 0\text{)} \tag{10.104}$$

The first term contained in the wavelet expansion of the function $x(t)$ represents the approximation of the function at scale level 0 by the linear combination of the scaling functions $\phi_{0,k}(t)$, and the summation with index j in the second term in the expansion is for the details of different levels contained in the function $x(t)$ approximated by the linear combination of the wavelet functions of progressively higher scales.

Example 10.4:

Here we use the Haar wavelets to approximate the following continuous function $x(t)$ defined over the period $0 \leq t < 1$:

$$x(t) = \begin{cases} t^2 & 0 \leq t < 1 \\ 0 & \text{otherwise} \end{cases} \tag{10.105}$$

We start at scale level $j = 0$. Each individual space $(V_0, W_0, W_1, \cdots)$ is spanned by different number of basis functions. For example, there is only one basis function in spaces V_0 and W_0, while space W_1 is spanned by 2 basis functions, and space W_2 is spanned by 4.

$$\begin{aligned} c_0(0) &= \int_0^1 t^2 \varphi_{0,0}(t)dt = \int_0^1 t^2(t)dt = \frac{1}{3} \\ d_0(0) &= \int_0^1 t^2 \psi_{0,0}(t)dt = \int_0^{0.5} t^2(t)dt - \int_{0.5}^1 t^2(t)dt = -\frac{1}{4} \\ d_1(0) &= \int_0^1 t^2 \psi_{1,0}(t)dt = \int_0^{0.25} \sqrt{2}t^2(t)dt - \int_{0.25}^{0.5} t^2\sqrt{2}(t)dt = -\frac{\sqrt{2}}{32} \\ d_1(1) &= \int_0^1 t^2 \psi_{1,1}(t)dt = \int_{0.5}^{0.75} \sqrt{2}t^2(t)dt - \int_{0.75}^1 t^2\sqrt{2}(t)dt = -\frac{3\sqrt{2}}{32} \end{aligned} \tag{10.106}$$

Therefore the wavelet series expansion of the function $x(t)$ is

$$x(t) = \frac{1}{3}\phi_{0,0}(t) + [-\frac{1}{4}\psi_{0,0}(t)] + [-\frac{\sqrt{2}}{32}\psi_{1,0}(t) - \frac{3\sqrt{2}}{32}\psi_{1,1}(t)] + \cdots \tag{10.107}$$

Here the first term is V_0, the second term is W_0, the third term is W_1, and $V_1 = V_0 \oplus W_0$, $V_2 = V_1 \oplus W_1 = V_0 \oplus W_0 \oplus W_1$

This process can be carried out further by including progressively more detailed information in wavelet spaces W_2, W_3, until $j \to \infty$. When higher temporal resolution (doubled) is obtained in the next wavelet space W_j, the corresponding frequency resolution is always reduced (halved), as shown in the Heisenberg box.

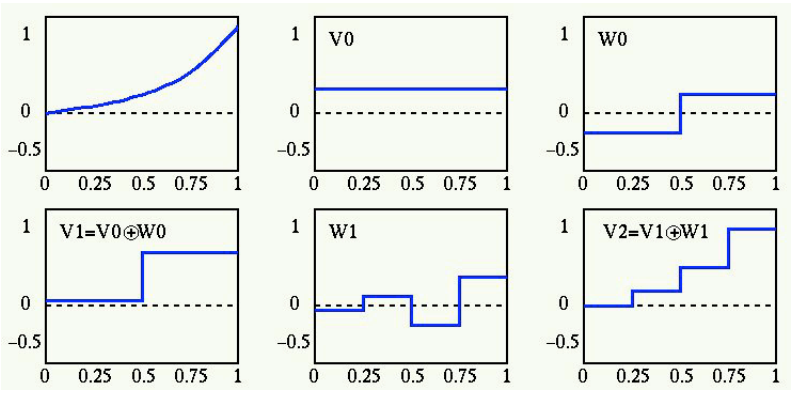

Figure 10.11 Wavelet approximation of a function

10.3 Discrete Wavelet Transform (DWT)

10.3.1 Iteration algorithm

To carry out the discrete wavelet transform, both the signal $x(t)$ and the basis functions $\phi_{0,k}(t)$ and $\psi_{j,k}(t)$ will need to be discretized. The signal becomes a vector $\mathbf{x} = [x[0], \cdots, x[N-1]]^T$ containing a set of N samples taken from a continuous signal

$$x[m] = x(m\triangle), \quad (m = 0, 1, \cdots, N-1) \tag{10.108}$$

for some sampling period $\triangle$. Similarly the basis functions $\phi_{0,k}(t)$ and $\psi_{j,k}(t)$ are also discretized to become basis vectors $\boldsymbol{\phi}_{0,k} = [\cdots, \phi_{0,k}[m], \cdots]^T$ and $\boldsymbol{\psi}_{j,k} = [\cdots, \psi_{j,k}[m], \cdots]^T$ for all k and all scale levels $j = 0, 1, \cdots, J-1$. Now the wavelet expansion becomes discrete wavelet transform (DWT) by which the discretized signal $x[m]$ is represented as a weighted sum in the space spanned by the discretized bases $\boldsymbol{\phi}_{0,k}$ and $\boldsymbol{\psi}_{j,k}$:

$$x[m] = \sum_k X_\phi[0,k]\phi_{0,k}[m] + \sum_{j=0}^{J-1}\sum_k X_\psi[j,k]\psi_{j,k}[m], \quad (m = 0, \cdots, N-1) \tag{10.109}$$

This is the inverse wavelet transform where the coefficients or weights are the projections of the signal vector on the orthogonal basis vectors:

$$X_\phi[0,k] = <\mathbf{x}, \boldsymbol{\phi}_{0,k}> = \sum_{m=0}^{N-1} x[m]\overline{\phi}_{0,k}[m], \quad (\text{for all } k) \tag{10.110}$$

$$X_\psi[j,k] = <\mathbf{x}, \boldsymbol{\psi}_{j,k}> = \sum_{m=0}^{N-1} x[m]\overline{\psi}_{j,k}[m], \quad (\text{for all } k \text{ and all } j > 0) \tag{10.111}$$

where $X_\phi[0,k]$ and $X_\psi[j,k]$ are the *approximation coefficient* and *detail coefficient*, respectively. These are the forward wavelet transform. Same as all other orthogonal transforms discussed before, the general application of the discrete wavelet transform is to represent the signal in terms of the DWT coefficients for different scales and translations (similar to the Fourier transform coefficients for different frequencies) in the transform domain, in which various filtering, feature extraction and compression can be carried out. The inverse DWT transform can then be carried out to reconstruct the signal back in time domain.

Example 10.5:

Assume $N = 4$-point discrete signal $\mathbf{x} = [x[0], \cdots, x[N-1]]^T = [1, 4, -3, 0]^T$ and the discrete Haar scaling and wavelet functions are:

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix} \begin{matrix} \phi_{0,0}[m] \\ \psi_{0,0}[m] \\ \psi_{1,0}[m] \\ \psi_{1,1}[m] \end{matrix} \tag{10.112}$$

The coefficient for V_0:

$$X_\phi[0,0] = \frac{1}{2}\sum_{m=0}^{3} x[m]\phi_{0,0}[m] = \frac{1}{2}[1\cdot 1 + 4\cdot 1 - 3\cdot 1 + 0\cdot 1] = 1 \tag{10.113}$$

The coefficient for W_0:

$$X_\psi[0,0] = \frac{1}{2}\sum_{m=0}^{3} x[m]\psi_{0,0}[m] = \frac{1}{2}[1\cdot 1 + 4\cdot 1 - 3\cdot(-1) + 0\cdot(-1)] = 4 \tag{10.114}$$

The two coefficients for W_1:

$$X_\psi[1,0] = \frac{1}{2}\sum_{m=0}^{3} x[m]\psi_{1,0}[m] = \frac{1}{2}[1\cdot\sqrt{2} + 4\cdot(-\sqrt{2}) - 3\cdot 0 + 0\cdot 0] = -1.5\sqrt{2} \tag{10.115}$$

$$X_\psi[1,1] = \frac{1}{2}\sum_{m=0}^{3} x[m]\psi_{1,0}[m] = \frac{1}{2}[1\cdot 0 + 4\cdot 0 - 3\cdot\sqrt{2} + 0\cdot(-\sqrt{2})] = -1.5\sqrt{2} \tag{10.116}$$

In matrix form, we have

$$\begin{bmatrix} 1 \\ 4 \\ -1.5\sqrt{2} \\ -1.5\sqrt{2} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}\begin{bmatrix} 1 \\ 4 \\ -3 \\ 0 \end{bmatrix} \tag{10.117}$$

Now the function $x[m]$ $(m = 0, \cdots, 3)$ can be expressed as a linear combination of these basis functions:

$$x[m] = \frac{1}{2}[X_\phi[0,0]\phi_{0,0}[m] + C_\psi[0,0]\psi_{0,0}[m] + X_\phi[1,0]\psi_{1,0}[m] + X_\phi[1,1]\psi_{1,1}[m]\ ] \tag{10.118}$$

or in matrix form:

$$\begin{bmatrix} 1 \\ 4 \\ -3 \\ 0 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 1 & 1 & \sqrt{2} & 0 \\ 1 & 1 & -\sqrt{2} & 0 \\ 1 & -1 & 0 & \sqrt{2} \\ 1 & -1 & 0 & -\sqrt{2} \end{bmatrix}\begin{bmatrix} 1 \\ 4 \\ -1.5\sqrt{2} \\ -1.5\sqrt{2} \end{bmatrix} \tag{10.119}$$

10.3.2 Fast Discrete Wavelet Transform (FDWT)

Here we consider Mallat's fast algorithm for the discrete wavelet transform. As shown before, the discrete wavelet transform of a discrete signal $\mathbf{x} = [x[0], \cdots, x[N-1]]^T$ is the process of getting the coefficients:

$$X_\phi[0,k] = \sum_{m=0}^{N-1} x[m]\phi_{0,k}[m] \quad \text{(for all } k\text{)} \tag{10.120}$$

$$X_\psi[j,k] = \sum_{m=0}^{N-1} x[m]\psi_{j,k}[m] \quad \text{(for all } k \text{ and all } j > 0\text{)} \tag{10.121}$$

However, as both $\phi_{j,l}[m]$ and $\psi_{j,l}[m]$ can be expressed as a linear combination of $\phi_{j+1,k}[m]$ (Eqs.10.23 and 10.46), the two equations above can be written as:

$$X_\phi[j,k] = \sum_{m=0}^{N-1} x[m]\phi_{j,k}[m] = \sum_{m=0}^{N-1} x[m] \sum_l h_0[l-2k]\phi_{j+1,l}(t)$$

$$= \sum_l h_0[l-2k] \sum_{m=0}^{N-1} x[m]\phi_{j+1,l}(t) = \sum_l h_0[l-2k]X_\phi[j+1,l] \tag{10.122}$$

and

$$X_\psi[j,k] = \sum_{m=0}^{N-1} x[m]\psi_{j,k}[m] = \sum_{m=0}^{N-1} x[m] \sum_l h_1[l-2k]\phi_{j+1,l}(t)$$

$$= \sum_l h_1[l-2k] \sum_{m=0}^{N-1} x[m]\phi_{j+1,l}(t) = \sum_l h_1[l-2k]X_\phi[j+1,l] \tag{10.123}$$

Comparing these equations with a discrete convolution:

$$y[k] = h[k] * x[k] = \sum_n h[k-n]x[n] \tag{10.124}$$

we see that the wavelet transform coefficients $X_\phi[j,k]$ and $X_\psi[j,k]$ at the jth scale can be obtained from the coefficients $X_\phi[j+1,k]$ at the (j+1)th scale by:

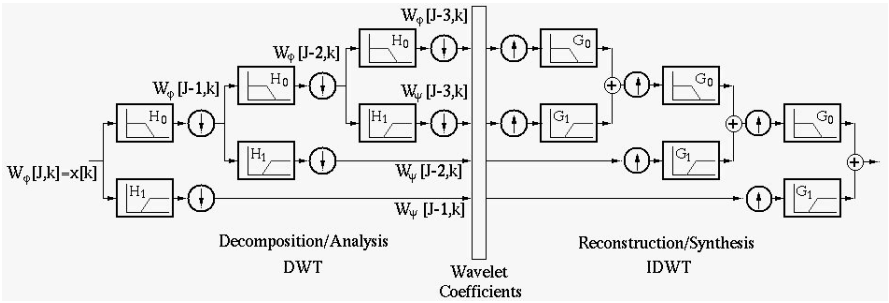

Figure 10.12 Filter banks for both forward and inverse DWT

- Convolution with time-reversed h_0 or h_1;
- Sub-sampling to get every other samples in the convolution.

We can therefore write

$$\begin{aligned} X_\psi[j,k] &= h_1[-n] * X_\phi[j+1,n]\big|_{n=2k} \\ X_\phi[j,k] &= h_0[-n] * X_\phi[j+1,n]\big|_{n=2k} \end{aligned} \tag{10.125}$$

Based on these two equations, all wavelet and scaling coefficients $X_\psi[j,k]$ and $X_\phi[j,k]$ for all scale levels of a given signal $\mathbf{x}$ can be obtained recursively from the coefficients $X_\phi[J,k]$ at the highest resolution level (with maximum details), which are the data samples $x[m]$ directly from the signal $x(t)$. As a member of the vector space V_J at the highest scale level, the discrete signal can be written as a linear combination of the scaling basis functions $\phi_{J,k}[m]$:

$$x[m] = \sum_k X_\phi[J,k]\phi_{J,k}[m], \qquad (m = 0, \cdots, N-1) \tag{10.126}$$

If we let the kth basis function be a unit impulse at the kth sampling time, i.e., $\phi_{J,k}[m] = \delta[k-m]$ (same as the ith component of a unit vector $\mathbf{e}_j$ in N-dimensional vector space is $e_{ij} = \delta[i-j]$), then the kth coefficient $X_\phi[J,k]$ is the same as the kth sample of the function $x(t)$. In other words, given $X_\phi[J,k] = x(k)$, the scaling and wavelet coefficients of the lower scales $j < J$ can be obtained by the subsequent filter bank, as shown on the left-hand side of Fig.10.12. The right-hand side is for the signal reconstruction, to be discussed in the following section.

The computation cost of the fast wavelet transform (FWT) is the convolutions carried out in each of the filters. The number of data samples in the convolution is halved after each sub-sampling, therefore the total complexity is:

$$O(N + \frac{N}{2} + \frac{N}{4} + \frac{N}{8} + \cdots + 1) = O(N) \tag{10.127}$$

i.e., the FWT has linear computational complexity.

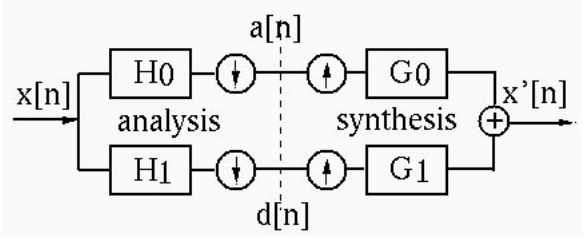

Figure 10.13 Two-channel filter bank

10.4 Filter Bank Implementation of DWT

As shown before, the forward wavelet transform that converts a given signal vector $\boldsymbol{x}$ into a set of transform coefficients $X_\phi[j,k]$ and $X_\psi[j,k]$ in the transform domain can be implemented by the analysis filter bank. Here we will further show that the inverse wavelet transform for the reconstruction of the signal from the DWT coefficients can be similarly implemented by a synthesis filter bank, as illustrated on the right-hand side of Fig.10.12. In the following we will derive the theory for the design of the the filters G_0 and G_1 in the synthesis filter bank.

10.4.1 Two-Channel Filter Bank

The DWT filter bank shown in Fig.10.12 can be considered as a recursive structure based on a two-channel filter bank, shown in Fig.10.13. This two-channel filter bank is composed of a low-pass filter $h_0[n]$ with output $a[n]$ (for approximation) and a high-pass filter $h_1[n]$ with outputs $d[n]$ (for detail) for the analysis filter bank, and two additional filters $g_0[n]$ and $g_1[n]$ for the synthesis filter bank. Our goal is to design the two filters $g_0[n]$ and $g_1[n]$ so that their output $\boldsymbol{x}'$ is the same as the input $\boldsymbol{x}$. Once this perfect reconstruction is achieved by the two-channel filter bank at the lowest level, it can also be achieved recursively at all higher levels in the entire filter bank in Fig.10.12.

According to Eqs.10.122 and 10.123, we have:

$$\begin{aligned} a[k] &= \sum_n h_0[n-2k]x[n] =< \boldsymbol{x}, \boldsymbol{h}_0(k) > \\ d[k] &= \sum_n h_1[n-2k]x[n] =< \boldsymbol{x}, \boldsymbol{h}_1(k) > \end{aligned} \tag{10.128}$$

and the output $x'[n]$ of the two-channel filter bank is:

$$x'[n] = \sum_k a[k]g_0[n-2k] + \sum_k d[k]g_1[n-2k], \quad \text{(for all } n\text{)} \tag{10.129}$$

or in vector form:

$$\boldsymbol{x}' = \sum_k a[k]\boldsymbol{g}_0(k) + \sum_k d[k]\boldsymbol{g}_1(k) \tag{10.130}$$

Our goal here is to design the two filters $g_0[n]$ and $g_1[n]$ on the right-hand side for the inverse DWT so that the output $x'[n] = x[n]$, i.e., the original signal can be perfectly reconstructed after DWT and inverse DWT. For convenience, we will carry out the derivation in the following in frequency domain based on the discrete-time Fourier transforms (DTFT) of signals and the impulse responses of the filters. Note that the DTFT spectra are all periodic with period 1, e.g., $H_0(f+1) = H_0(f)$ (or equivalently $H_0(\omega + 2\pi) = H_0(\omega)$).

Based on the down-sampling property of the discrete-time Fourier transform (Eq.4.39), the outputs of filters $H_0(f)$ and $H_1(f)$ can be expressed in frequency domain as:

$$A(f) = \frac{1}{2}[H_0(\frac{f}{2})X(\frac{f}{2}) + H_0(\frac{f+1}{2})X(\frac{f+1}{2})] \tag{10.131}$$

$$D(f) = \frac{1}{2}[H_1(\frac{f}{2})X(\frac{f}{2}) + H_1(\frac{f+1}{2})X(\frac{f+1}{2})] \tag{10.132}$$

Then, based on the up-sampling property of the DTFT (Eq.4.36), the overall output $x'[n]$ can be expressed as:

$$\begin{aligned} X'(f) &= G_0(f)A(2f) + G_1(f)D(2f) \\ &= \frac{1}{2}[G_0(f)H_0(f) + G_1(f)H_1(f)]\ X(f) \\ &+ \frac{1}{2}[G_0(f)H_0(f+\frac{1}{2}) + G_1(f)H_1(f+\frac{1}{2})]\ X(f+\frac{1}{2}) \end{aligned} \tag{10.133}$$

For perfect reconstruction, the output must be identical to the original signal, i.e., $X(z) = X'(z)$, we need to have

$$\begin{cases} G_0(f)H_0(f+\frac{1}{2}) + G_1(f)H_1(f+\frac{1}{2}) = 0 \\ G_0(f)H_0(f) + G_1(f)H_1(f) = 2 \end{cases} \tag{10.134}$$

These two equations can be written in matrix form as:

$$\begin{bmatrix} H_0(f+\frac{1}{2}) & H_1(f+\frac{1}{2}) \\ H_0(f) & H_1(f) \end{bmatrix} \begin{bmatrix} G_0(f) \\ G_1(f) \end{bmatrix} = \mathbf{H}(f) \begin{bmatrix} G_0(f) \\ G_1(f) \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \end{bmatrix} \tag{10.135}$$

where $\boldsymbol{H}$ is defined as:

$$\mathbf{H}(f) = \begin{bmatrix} H_0(f+\frac{1}{2}) & H_1(f+\frac{1}{2}) \\ H_0(f) & H_1(f) \end{bmatrix} \quad \text{and} \quad \mathbf{H}^{-1}(f) = \frac{1}{\Delta(f)} \begin{bmatrix} H_1(f) & -H_1(f+\frac{1}{2}) \\ -H_0(f) & H_0(f+\frac{1}{2}) \end{bmatrix} \tag{10.136}$$

where $\Delta(f)$ is the determinant of $\mathbf{H}(f)$:

$$\Delta(f) = H_0(f+\frac{1}{2})H_1(f) - H_0(f)H_1(f-\frac{1}{2}) \tag{10.137}$$

Solving the equation above for $G_0(f)$ and $G_1(f)$, we get:

$$\begin{aligned} \begin{bmatrix} G_0(f) \\ G_1(f) \end{bmatrix} &= \mathbf{H}^{-1}(f) \begin{bmatrix} 0 \\ 2 \end{bmatrix} = \frac{1}{\Delta(f)} \begin{bmatrix} H_1(f) & -H_1(f+\frac{1}{2}) \\ -H_0(f) & H_0(f+\frac{1}{2}) \end{bmatrix} \begin{bmatrix} 0 \\ 2 \end{bmatrix} \\ &= \frac{2}{\Delta(f)} \begin{bmatrix} -H_1(f+\frac{1}{2}) \\ H_0(f+\frac{1}{2}) \end{bmatrix} \end{aligned} \tag{10.138}$$

Now $G_0(f)$ and $G_1(f)$ can be expressed as:

$$G_0(f) = \frac{-2}{\Delta(f)} H_1(f + \frac{1}{2}), \quad G_1(f) = \frac{2}{\Delta(f)} H_0(f + \frac{1}{2}) \tag{10.139}$$

Also, if we replace f by $f + \frac{1}{2}$ in $\mathbf{H}$, and notice that $H_0(f+1) = H_0(f)$ and $H_1(f+1) = H_1(f)$ have period 1, we get:

$$\mathbf{H}(f + \frac{1}{2}) = \begin{bmatrix} H_0(f) & H_1(f) \\ H_0(f + \frac{1}{2}) & H_1(f + \frac{1}{2}) \end{bmatrix} \tag{10.140}$$

and $\Delta(f + \frac{1}{2}) = -\Delta(f)$. Now we can replace f by $f + \frac{1}{2}$ in the above expression for $G_1(f)$ to get:

$$G_1(f + \frac{1}{2}) = \frac{-2}{\Delta(f)} H_0(f) \tag{10.141}$$

Multiplying the two sides of this equation by $\frac{-2}{\Delta} H_1(f + \frac{1}{2}) = G_0(f)$, which is just the first equation in Eq.10.139, we get:

$$G_1(f + \frac{1}{2}) H_1(f + \frac{1}{2}) = G_0(f) H_0(f), \quad \text{i.e.} \quad G_1(f) H_1(f) = G_0(f + \frac{1}{2}) H_0(f + \frac{1}{2}) \tag{10.142}$$

This equation can be substituted back into the two equations in Eq.10.134 in different ways to get the following four conditions for perfect reconstruction:

$$\begin{aligned} G_0(f) H_0(f) + G_0(f + \frac{1}{2}) H_0(f + \frac{1}{2}) &= 2 \\ G_1(f) H_1(f) + G_1(f + \frac{1}{2}) H_1(f + \frac{1}{2}) &= 2 \\ G_1(f) H_0(f) + G_1(f + \frac{1}{2}) H_0(f + \frac{1}{2}) &= 0 \\ G_0(f) H_1(f) + G_0(f + \frac{1}{2}) H_1(f + \frac{1}{2}) &= 0 \end{aligned} \tag{10.143}$$

Comparing these equations with the three equations in Eq.10.71 for the properties of $H_0(f)$ and $H_1(f)$, we see that these conditions will be satisfied if the following hold:

$$G_0(f) = \overline{H}_0(f), \quad \text{and} \quad G_1(f) = \overline{H}_1(f) \tag{10.144}$$

These two relations can be considered as the new conditions for perfect reconstruction, and they will be satisfied if the following is true in time domain for $i = 0, 1$:

- $g_i[n] = h_i[-n]$ is the time reversal of $h_i[n]$, so that $G_i(f) = H_i(-f)$ according to the time reversal property of DTFT (Eq.4.23);
- All filter coefficients $\overline{h}_i[n] = h_i[n]$ are real, so that $H_i(-f) = \overline{H}_i(f)$ according to the time reversal property of DTFT (Eq.4.24.

Now we see that given $h_0[n]$ and $h_1[n]$ in the analysis filter bank, $g_0[n]$ and $g_1[n]$ in the synthesis filter bank can be easily obtained as shown above for a perfect signal reconstruction.

Moreover, based on the DTFT properties of down and up-sampling (Eqs.4.39 and 4.36), the four biorthogonal relations in Eq.10.143 are the down and up-sampled versions of $G_0(f)H_0(f)$, $G_1(f)H_1(f)$, $G_1(f)H_0(f)$, $G_0(f)H_1(f)$, corresponding to the following four down-sampled convolutions in time domain:

$$\begin{aligned} g_0[2n] * h_0[2n] &= \sum_k h_0[k]g_0[2n-k] = \delta[n] \\ g_1[2n] * h_1[2n] &= \sum_k h_1[k]g_1[2n-k] = \delta[n] \\ g_1[2n] * h_0[2n] &= \sum_k h_1[k]g_0[2n-k] = 0 \\ g_0[2n] * h_1[2n] &= \sum_k h_0[k]g_1[2n-k] = 0 \end{aligned} \tag{10.145}$$

Comparing the first two convolutions above with the orthonormality property of the scaling filter h_0 and wavelet filter h_1 in Eq.10.66, we also see that here $g_0[n] = h_0[-n]$ and $g_1[n] = h_1[-n]$ are the time reversal of $h_0[n]$ and $h_1[n]$, respectively. If express the four filters h_0, h_1, g_0 and g_1, all shifted by $2n$ positions, as four vectors $\boldsymbol{h}_i(n) = [\cdots, h_i[k-2n], \cdots]^T$ and $\boldsymbol{g}_i(n) = [\cdots, h_1[k-2n], \cdots]^T$ for $i = 0, 1$, the four convolutions in Eq.10.145 can now be written as vector inner products:

$$\begin{aligned} &< \boldsymbol{g}_0(0), \boldsymbol{h}_0(n) >= \delta[n] \\ &< \boldsymbol{g}_1(0), \boldsymbol{h}_1(n) >= \delta[n] \\ &< \boldsymbol{g}_1(0), \boldsymbol{h}_0(n) >= 0 \\ &< \boldsymbol{g}_0(0), \boldsymbol{h}_1(n) >= 0 \end{aligned} \tag{10.146}$$

These four equations can be further summarized as:

$$< \boldsymbol{g}_i(n), \boldsymbol{h}_j(0) >= \delta[i-j]\delta[n], \qquad (i, j = 0, 1) \tag{10.147}$$

This is the biorthogonal relationship between the analysis and synthesis filters, as discussed in Chapter 2 (Theorem 2.11).

We can now verify that Eq.10.130 is indeed a perfect reconstruction of the original signal $\boldsymbol{x}$, if Eq.10.146 is satisfied. We first assume the given signal $\boldsymbol{x}$ can indeed be expanded in the following form:

$$\boldsymbol{x} = \sum_k a_k \boldsymbol{g}_0(k) + \sum_k d_k \boldsymbol{g}_1(k) \tag{10.148}$$

where a_k and d_k are two sets of coefficients which can be found by taking the inner product with $\boldsymbol{h}_0(l)$ and $\boldsymbol{h}_1(l)$, respectively, on both sides of this equation:

$$\begin{aligned} < \boldsymbol{x}, \boldsymbol{h}_0(l) > &= \sum_k a_k < \boldsymbol{g}_0(k), \boldsymbol{h}_0(l) > + \sum_k d_k < \boldsymbol{g}_1(k), \boldsymbol{h}_0(l) > \\ &= \sum_k a[k]\delta[k-l] = a_l \end{aligned} \tag{10.149}$$

and

$$< \boldsymbol{x}, \boldsymbol{h}_1(l) > = \sum_k a_k < \boldsymbol{g}_0(k), \boldsymbol{h}_1(l) > + \sum_k d_k < \boldsymbol{g}_1(k), \boldsymbol{h}_1(l) >$$
$$= \sum_k d[k]\delta[k-l] = d_l \tag{10.150}$$

We see that the two coefficients a_l and d_l needed for the expansion of $\boldsymbol{x}$ are exactly the same as $a[k]$ and $d[k]$ in Eq.10.128, used in Eq.10.130 to generate the output $\boldsymbol{x}'$, i.e., it is indeed the perfect reconstruction of the input signal $\boldsymbol{x}$.

In summary, we can view the two-channel filter bank in Fig.10.13 as a process of signal transform based on two pairs of biorthogonal bases $\{\boldsymbol{h}_0, \boldsymbol{g}_0\}$ and $\{\boldsymbol{h}_1, \boldsymbol{g}_1\}$, where $\boldsymbol{g}_0$ and $\boldsymbol{g}_1$ are dual to $\boldsymbol{h}_0$ and $\boldsymbol{h}_1$, respectively. This transform is essentially the same as what we discussed in Theorem 2.11, where a signal $\boldsymbol{x}$ is reconstructed according to Eq.2.253 as

$$\boldsymbol{x}' = \sum_k < \boldsymbol{x}, \boldsymbol{h}_k > \tilde{\boldsymbol{\phi}}_k \tag{10.151}$$

We see that this reconstruction is different from the two-channel filter bank discussed above in that only a pair of dual biorthogonal bases is used, while in the case of the 2-channel filter bank, two pairs are used.

Recall that we discussed a two-point filter bank implementation of Haar transform (section 7.10 of Chapter 7), which is actually a simple example of the general discrete wavelet transform.

We list below the Matlab code for the implementation of the two-channel filter bank. Based on this code, the forward discrete wavelet transform for signal decomposition and the inverse DWT for signal reconstruction can be recursively constructed.

```
function y=reconstruction(x,h)
N=length(x);                        % length of signal vector
K=length(h);                        % length of filter (K<N)
h=h/norm(h);                        % normalize h
h0=zeros(1,N); h0(1:K)=h;           % analysis filter H0
H0=fft(h0);
for k=0:N-1
    m=mod(k-N/2,N)+1;
    H1(k+1)=-exp(-j*2*pi*k/N)*conj(H0(m));  % analysis filter H1
end
G0=conj(H0); G1=conj(H1);           % synthesis filters G0 and G1:
% Decomposition by analysis filters:
A=fft(x);                           % input signal as initial approximation
d=ifft(A.*H1);                      % filtering of detail d
a=ifft(A.*H0);                      % filtering of approximation a
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 for A
d=upsample(d,2);                 % upsampling for D
a=ifft(fft(a).*G0);              % filtering of a
d=ifft(fft(d).*G1);              % filtering of d
y=a+d;                           % perfect reconstruction of x
```

As can be seen, here the filtering is carried out in frequency domain by multiplication. Alternatively, the filtering can also be carried out in time domain as a circular convolution, as discussed in 4.2.5.

The code for both forward and inverse DWT transforms is listed below. The input of the forward DWT function includes a vector x for the signal to be transformed and another vector h for the father wavelet coefficients $h_0[k]$, and the output is a vector w for the DWT coefficients.

```
function w=dwt(x,h)
K=length(h);
N=length(x); n=log2(N);
if n~=int16(n)
     error('Length of data x should be power of 2');
end
if K>N
   error('K should be less than N'); % assume N > K
end
h=h/norm(h);                          % normalize h
h0=zeros(1,N);
h0(1:K)=h;
H0=fft(h0);
for k=0:N-1
   m=mod(k-N/2,N)+1;
   H1(k+1)=-exp(-j*2*pi*k/N)*conj(H0(m));
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 wavelet coefficients
    n=n/2;
end
w=[a w];                        % append signal residual
```

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=idwt(w,h)
N=length(w); n=log2(N); K=length(h);
if n~=int16(n)
     error('Length of data w should be power of 2');
end
h0=zeros(1,N); h0(1:K)=h; H0=fft(h0);
for k=0:N-1
    m=mod(k-N/2,N)+1;
    H1(k+1)=-exp(-j*2*pi*k/N)*conj(H0(m));
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'; end % upsampling 1x1 is column vector
    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;
```

10.4.2 Perfect Reconstruction Filters

As there are four function variables H_0, H_1, G_0 and G_1 in the two equations in Eq.10.134, there exist multiple designs for the filter banks. Here are three particular ones:

- **Quadrature mirror filters (QMFs)** We let

$$H_1(z) = H_0(-z), \quad G_0(z) = H_0(z), \quad G_1(z) = -H_0(-z) \tag{10.152}$$

both of the two equations above can be written in terms of $H_0(z)$. The first equation above becomes:

$$G_0(z)H_0(-z) + G_1(z)H_1(-z) = H_0(z)H_0(-z) - H_0(-z)H_0(z) = 0 \tag{10.153}$$

and the second equation becomes:

$$G_0(z)H_0(z) + G_1(z)H_1(z) = H_0(z)H_0(z) - H_0(-z)H_0(-z) = 2 \tag{10.154}$$

where $H_0(z)$ is so chosen that $H_0^2(z) - H_0^2(-z) = 2$ to satisfy the requirement for perfect reconstruction.

- **Conjugate quadrature filters (CQFs)**
We let

$$G_0(z) = H_0(z^{-1}), \quad G_1(z) = zH_0(-z), \quad H_1(z) = z^{-1}H_0(-z^{-1}) \tag{10.155}$$

and both of the two equations above can be written in terms of H_0. The first equation above becomes:

$$G_0(z)H_0(-z) + G_1(z)H_1(-z) = H_0(z^{-1})H_0(-z) - zH_0(-z)z^{-1}H_0(z^{-1}) = 0 \tag{10.156}$$

and the second equation becomes:

$$\begin{aligned} &G_0(z)H_0(z) + G_1(z)H_1(z) = H_0(z^{-1})H_0(z) + zH_0(-z)z^{-1}H_0(-z^{-1}) \\ &= H_0(z^{-1})H_0(z) + H_0(-z)H_0(-z^{-1}) = 2 \end{aligned} \tag{10.157}$$

where $H_0(z)$ is so chosen that the second expression is 2 to satisfy the requirement for perfect reconstruction.

- **Orthonormal (fast wavelet transform) filter**
We let

$$H_0(z) = G_0(z^{-1}), \quad H_1(z) = G_1(z^{-1}), \quad G_1(z) = -z^{-2k+1}G_0(-z^{-1}) \tag{10.158}$$

and both of the two equations above can be written in terms of $G_0(z)$. The first equation above becomes:

$$\begin{aligned} &G_0(z)H_0(-z) + G_1(z)H_1(-z) = G_0(z)G_0(-z^{-1}) + G_1(z)G_1(-z^{-1}) \\ &= G_0(z)G_0(-z^{-1}) - z^{-2k+1}G_0(-z^{-1})z^{2k-1}G_0(z) \\ &= G_0(z)G_0(-z^{-1}) - G_0(-z^{-1})G_0(z) = 0 \end{aligned} \tag{10.159}$$

and the second equation becomes:

$$\begin{aligned} &G_0(z)H_0(z) + G_1(z)H_1(z) = G_0(z)G_0(z^{-1}) + G_1(z)G_1(z^{-1}) \\ &= G_0(z)G_0(z^{-1}) + [-z^{-2k+1}G_0(-z^{-1})][-z^{2k-1}G_0(-z)] \\ &= G_0(z)G_0(z^{-1}) + G_0(-z^{-1})G_0(-z) = 2 \end{aligned} \tag{10.160}$$

where $G_0(z)$ is so chosen that the second expression is 2 to satisfy the requirement for perfect reconstruction. Note that $P(z) = G_0(z)G_0(z^{-1})$ is the Z-transform of the autocorrelation $p[n] = \sum_m g_0[m]g_0[m+n]$, and the second equation becomes

$$P(z) + P(-z) = 2 \quad \text{i.e.} \quad \frac{1}{2}[P(z) + P(-z)] = 1 \tag{10.161}$$

Replacing z by $z^{1/2}$, we get

$$\frac{1}{2}[P(z^{1/2}) + P(-z^{1/2})] = 1 \tag{10.162}$$

Consider the down sampled version of the function $g_0'[n] = g_0[2n]$, and its autocorrelation $p'[n] = p[2n]$. As in Z domain we have:

$$P'(z) = \frac{1}{2}[P(z^{1/2}) + P(-z^{1/2})] = 1 \tag{10.163}$$

in time domain we have

$$p'[n] = \sum_m g'[m]g'[n+m] = \sum_m g[m]g[2n+m] = \delta[n] \tag{10.164}$$

i.e., the down-sampled version of $g_0[n]$ is orthonormal.

Example: There are different ways to design the FIR filter orthonormal impulse response $g_0[n]$ for the two-channel filter bank.

The conditions for perfect construction filters listed above can be inverse Z-transformed to get:

$$H_i(z) = G_i(z^{-1}) \leftrightarrow h_i[n] = g_i[-n], \quad (i = 0, 1) \tag{10.165}$$

$$G_1(z) = -z^{-2k+1}G_0(-z^{-1}) \leftrightarrow g_1[n] = (-1)^n g_0[2k-1-n] \tag{10.166}$$

i.e., h_i is the time-reversed version of g_i $(i = 0, 1)$, and g_1 is both time reversed and modulated version of g_0. Once g_0 is determined, the rest can all be determined.

10.5 Two-Dimensional DWT

Similar to all orthogonal transforms previously discussed, the discrete wavelet transform can also be applied to two-dimensional signals such as an image. Similar to the 1-D DWT two-channel filter bank shown in Fig.10.13, a 2-D DWT

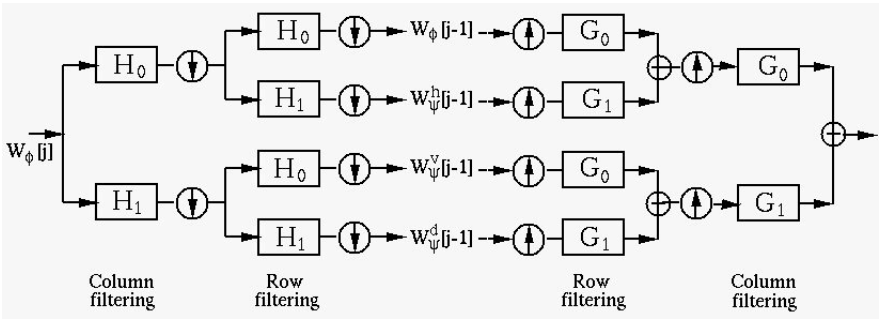

Figure 10.14 2-D two-channel filter bank

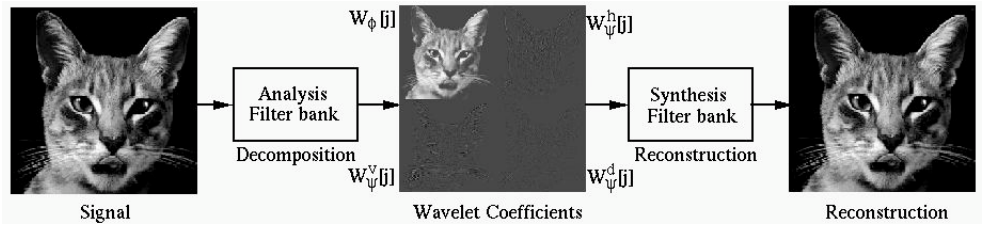

Figure 10.15 Signal decomposition and reconstruction by 2-D two-channel filter bank

two-channel filter bank for both analysis and synthesis is shown in Fig.10.14, where the left half is the analysis filter bank for signal decomposition and the right half is the synthesis filter bank for signal reconstruction. The input of the analysis filter bank is a 2-D signal array treated as the coefficients $X_\phi[j]$ at scale level j, and its columns are filtered (horizontal filtering) by the low-pass filter $H_0(f)$ and high-pass filter $H_1(f)$, and then the columns of the two resulting arrays are further filtered (vertical filtering) by $H_0(f)$ and $H_1(f)$ to generate four sets of coefficients at the next scale level $j-1$, including $X_\phi^h[j-1]$ low-pass filtered by $H_0(f)$ in both directions, $X_\psi^h[j-1]$ high-pass filtered by $H_1(f)$ in horizontal direction, $X_\psi^v[j-1]$ high-pass filtered in vertical direction, and $X_\psi^d[j-1]$ high-pass filtered in both directions (diagonal). The synthesis filter bank reverse the process to generate a perfectly reconstructed signal as the output.

This two-channel filtering can be carried out to the low-pass filtered signal $X_\phi[j-1]$ to generate four sets of coefficients at the next scale level $j-2$, and this process can be further carried out recursively to obtain the complete 2-D DWT coefficients, as illustrated in Fig.10.16. Four sets of these coefficients obtained at four consecutive stages of the recursion are shown in Fig.10.17. Note that the 2-D DWT coefficients look very much like other 2-D transforms such as discrete cosine transform and Haar transform, in the sense that the coefficients around the top left corner represent low-frequency components of the signal while those around the bottom right corner represent high-frequency components.

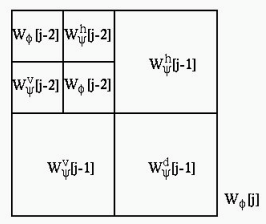

Figure 10.16 Recursion of 2-D discrete wavelet transform

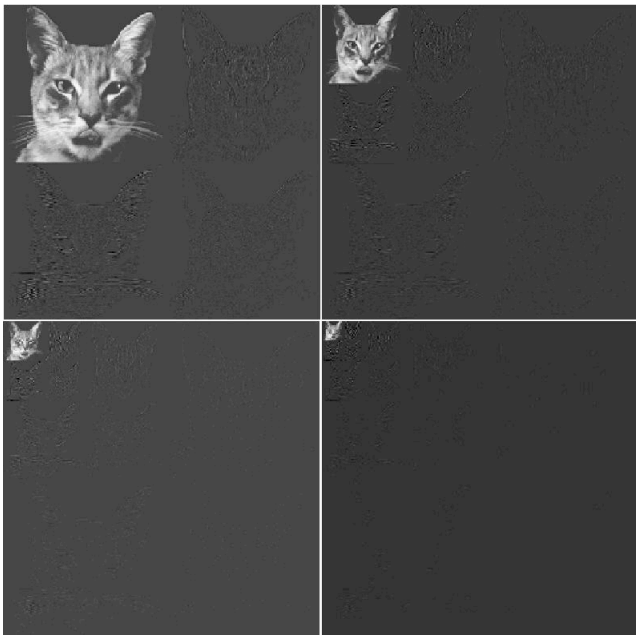

Figure 10.17 2-D DWT coefficients obtained at four consecutive stages

The Matlab code for both 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 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
   m=mod(k-N/2,N)+1;
   H1(k+1)=-exp(-j*2*pi*k/N)*conj(H0(m));
end
a=x;
imshow(a,[]);
w=zeros(N);
n=length(a);
while n>=K
    pause;
    t=zeros(n,n);
    for k=1:n                  % for all n columns
        A=fft(double(a(:,k)));
        D=real(ifft(A.*H1'));  % convolution d=a*h1
        A=real(ifft(A.*H0'));  % convolution a=a*h0
        t(:,k)=[A(2:2:n); D(2:2:n)];
    end
    for k=1:n                  % for all n rows
        A=fft(t(k,:));
        D=real(ifft(A.*H1));   % convolution d=a*h1
        A=real(ifft(A.*H0));   % convolution a=a*h0
        t(k,:)=[A(2:2:n) D(2:2:n)];
    end
    w(1:n,1:n)=t;              % concatenate wavelet coefficients
    H0=H0(1:2:length(H0));     % subsampling H0
    H1=H1(1:2:length(H1));     % subsampling H1
    n=n/2;
    a=t(1:n,1:n);
    imshow(w,[]);
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)
N=length(w); n=log2(N); K=length(h);
h=h/norm(h);              % normalize h
h0=zeros(1,N); h0(1:K)=h; H0=fft(h0);
for k=0:N-1
    m=mod(k-N/2,N)+1;
    H1(k+1)=-exp(-j*2*pi*k/N)*conj(H0(m));
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);                             % signal size of initial scale
y=w;
t=y(1:n,1:n);
while n<N
    fprintf('\ndata length: %d\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
        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;
    imshow(y,[]);  pause;
end
```

10.6 Applications in Data Compression

In many fields of social and natural sciences as well as engineering, a large quantity of raw data is regularly collected and accumulated, often automatically. However, it may become more challenging to transmit and store the data, and, more importantly, to extract the information meaningful to the specific field, due also to the large quantity of the data. To address such issues, transform based methods for data compression and information extraction are widely used, based on various orthogonal transforms such as DFT, DCT and DWT.

Due to the essential nature of all orthogonal transforms, a given signal in the transform domain is always decorrelated and its energy (information) redistributed, and more concentrated in a small number of components, compared to the original signal in either temporal or spatial domain. Consequently it is in general much more convenient and effective to carry out information extraction and data compression in transform domain.

Specifically, by taking the orthogonal transform on a signal originally given as a 1-D function of time or 2-D function of space (e.g., an image), it is converted to a set of transform coefficients, and the total energy contained in the signal is likely to be concentrated in a small number of the coefficients so that the rest of the coefficients containing little energy can be dropped, i.e., suppressed to be zero, without losing much information contained in the signal. Such methods are called lossy compression, as some information, although very little, does get lost. Due to the tremendous reduction of the data achieved by the orthogonal transform, combined with other coding methods (such as Huffman coding), the subsequent storage and transmission can be carried out much more efficiently.

For example, the DCT transform is used in the image compression standard JPEG, named after the Joint Photographic Experts Group, who developed this standard, and the DWT transform is used in the later version of the standard JPEG2000. Both transform methods can drastically compact most of the signal energy in a small number of transform coefficients. However, as the wavelet transform is capable of representing information of the signal $x(t)$ with temporal or spatial locality, as well as frequency locality in terms of different scale levels for different resolutions, its performance is superior to the DCT, both in terms of the percentage of energy conserved and the percentage of data kept, but also in terms of subjective evaluation of the compressed images.

Example 10.6: The image Lenna is compressed using both DCT and DWT, as shown in Fig. 10.18. The original image with pixels $x[i, j]$ and its DCT spectrum composed of frequency coefficients $X[k, l]$ are shown respectively in the top and bottom panels in left column. Then a threshold value is use to suppress to zero all DCT coefficients in the spectrum containing energy less than the threshold energy level. Here the energy contained in a frequency component is simply its

value squared, and the total energy contained in the signal is conserved before and after any orthogonal transform according to Parseval's theorem:

$$\mathcal{E} = \sum_k \sum_l X[k,l]^2 = \sum_i \sum_j x[i,j]^2 \tag{10.167}$$

The threshold value is so chosen that 99% of the total amount of signal energy are kept by 2.44% of the DCT coefficients with values above the threshold, while all remaining coefficients containing only 1% are suppressed to zero. The filtered spectrum and the reconstructed image are shown in the 2nd column of the figure.

The result of the same compression method based on the DWT is shown in the right four panels in the figure. The original image and its DWT coefficients are shown in the 3rd column. Now a threshold value is chosen so that 2.01% of the DWT coefficients above the threshold also contain 99% of the total signal energy, as shown in the panel on the bottom right, together with the corresponding reconstruction of the image in the top right panel.

We can make an obvious observation: in order to preserve 99% of the total signal energy, 2.01% of the data is needed after DWT, but 2.44% of the data is needed after DCT. Moreover, the quality of the reconstructed image can also be evaluated subjectively. We see that the reconstructed image based on the DWT is more visually acceptable than that based on the DCT.

Another aspect of this image compression example is the different energy distributions before and after the transform, either DCT or DWT. Fig.10.19 shows the histogram of the pixels in the image over all 256 gray scales (top panel), and the histograms of the transform coefficients over the range of all values (both positive and negative) after the DCT (middle panel) and DWT (bottom panel). We see clearly that the energy is relatively evenly distributed among all gray scale levels before the transform, but it is highly concentrated in the transform coefficients with values around zero after the transform, i.e., most of the coefficients after the transform take very low values. This fact lends itself very well to a lossless compression method based an entropy encoding algorithm called Huffman coding. Essentially, Huffman coding assigns variable code lengths to different symbols to be transmitted according to how frequent or probable each symbol occurs. By assigning shorter code length to more frequent symbols, Huffman coding can minimize the total code length. The effectiveness of Huffman coding is directly related to the entropy of the data defined as:

$$H = -\sum_k p_k log_2 p_k \tag{10.168}$$

where p_k is the frequency or probability of the kth symbol, in our case, for all pixels or transform coefficients taking the kth value in the histogram. The entropy values of the original image and its DCT and DWT transform coefficients are computed to be 6.05, 0.41 and 0.37, respectively. Correspondingly, Huffman coding will be most effective for the DWT coefficients, less so for the DCT coefficients, and not very effective at all for the pixels in the original image.

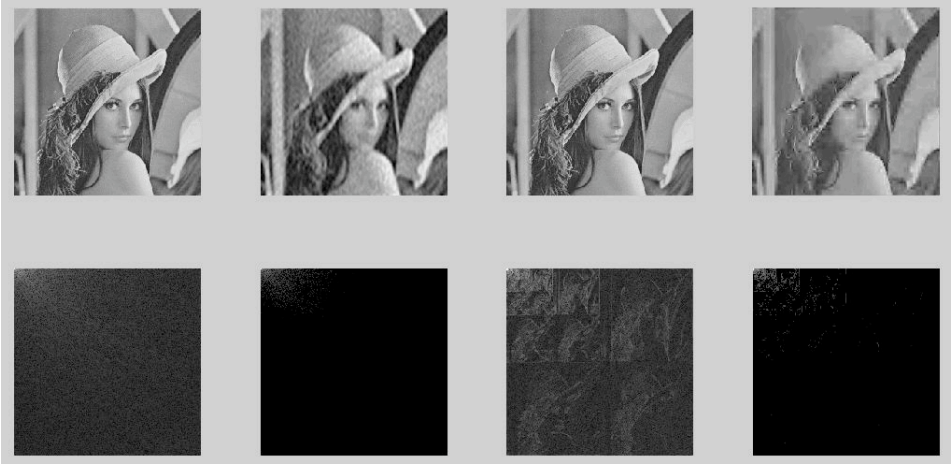

Figure 10.18 Image compression based on DCT and DWT
A nonlinear mapping $y = x^{0.3}$ is applied to all DCT and DWT coefficients for all of them to be visible.

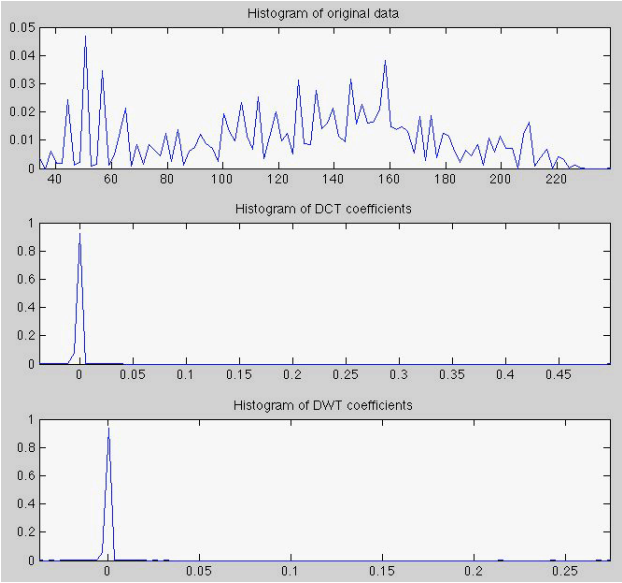

Figure 10.19 Signal histogram before and after DCT and DWT
The entropy values for the three histograms are 6.05, 0.41 and 0.37, respectively.

11 Appendix 1: Review of Linear Algebra

11.1 Basic Definitions

- **Matrix**
  An $m \times n$ matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ or $\mathbb{C}^{m\times n}$ is an array of m rows and n columns

$$\boldsymbol{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}_{m\times n} \tag{11.1}$$

  where $a_{ij} \in \mathbb{R}$ or $\mathbb{C}$ is the element in the ith (first index) row and jth (second index) column. In particular,
  – if $m = n$, $\boldsymbol{A}$ becomes a square matrix;
  – if $m = 1$, $\boldsymbol{A}$ becomes an n-dimensional (1 by n) row vector;
  – if $n = 1$, $\boldsymbol{A}$ becomes an m-dimensional (m by 1) column vector.
  Through out the book, a vector $\boldsymbol{a}$ 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

$$\boldsymbol{A} = [\boldsymbol{a}_1, \cdots, \boldsymbol{a}_n] \tag{11.2}$$

  where $\boldsymbol{a}_j$ $(j = 1, \cdots, n)$ is an m-dimensional column vector:

$$\boldsymbol{a}_j = \begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{mj} \end{bmatrix} \tag{11.3}$$

  The ith 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 $\boldsymbol{A}$, denoted by $\boldsymbol{A}^T$, is an $n \times m$ matrix obtained by swapping elements a_{ij} and a_{ji} for all $i, j \in \{1, \cdots, n\}$. In other words, the jth column of $\boldsymbol{A}$ becomes the jth row of $\boldsymbol{A}^T$, and at the same time,

the ith row of $\boldsymbol{A}$ becomes the ith column of $\boldsymbol{A}^T$:

$$\boldsymbol{A}^T = [\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_n]^T = \begin{bmatrix} \boldsymbol{a}_1^T \\ \boldsymbol{a}_2^T \\ \vdots \\ \boldsymbol{a}_n^T \end{bmatrix} = \begin{bmatrix} a_{11} & a_{21} & \cdots & a_{m1} \\ a_{12} & a_{22} & \cdots & a_{m2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1n} & a_{2n} & \cdots & a_{mn} \end{bmatrix}_{n \times m} \tag{11.4}$$

where $\boldsymbol{a}_j$ is the jth column of $\boldsymbol{A}$ and its transpose $\boldsymbol{a}_j^T$ is the jth row of $\boldsymbol{A}^T$:

$$\boldsymbol{a}_j^T = \begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{nj} \end{bmatrix}^T = [a_{1j}, a_{2j}, \cdots, a_{nj}] \tag{11.5}$$

Here are some important properties related to transpose:

$$(\boldsymbol{A}^T)^T = \boldsymbol{A}, \quad (\boldsymbol{AB})^T = \boldsymbol{B}^T \boldsymbol{A}^T \tag{11.6}$$

The *conjugate transpose* of an $m \times n$ complex matrix $\boldsymbol{A}$, denoted by $\boldsymbol{A}^*$, is the complex conjugate of its transpose, i.e.,

$$\boldsymbol{A}^* = \overline{\boldsymbol{A}^T} = \overline{\boldsymbol{A}}^T \tag{11.7}$$

i.e., the element in the ith row and jth column of $\boldsymbol{A}^*$ is the complex conjugate of the element in the jth row and ith column of $\boldsymbol{A}$. We obviously have:

$$(\boldsymbol{A}^*)^* = \boldsymbol{A}, \quad (\boldsymbol{AB})^* = \boldsymbol{B}^* \boldsymbol{A}^* \tag{11.8}$$

- **Identity Matrix**
  The *identity matrix* $\boldsymbol{I}$ is a special $n \times n$ square matrix with all elements being zero except those along the main diagonal which are 1:

$$\boldsymbol{I} = diag[1, \cdots, 1] = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}_{n \times n} \tag{11.9}$$

  The identity matrix can also be expressed in terms of its column vectors:

$$\boldsymbol{I} = [\boldsymbol{e}_1, \cdots, \boldsymbol{e}_n] \tag{11.10}$$

  where $\boldsymbol{e}_i$ is an n-dimensional column vector with all elements equal to zero except the ith one which is 1:

$$\boldsymbol{e}_i = [0, \cdots, 0, 1, 0, \cdots, 0]^T \tag{11.11}$$

- **Scalar Multiplication**

A matrix $\boldsymbol{A}$ can be multiplied by a scalar c to get

$$c\boldsymbol{A} = c\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} = \begin{bmatrix} ca_{11} & ca_{12} & \cdots & ca_{1n} \\ ca_{21} & ca_{22} & \cdots & ca_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ ca_{m1} & ca_{m2} & \cdots & ca_{mn} \end{bmatrix} \tag{11.12}$$

- **Dot Product**
  The *dot product*, also called *inner product*, of two real column vectors $\boldsymbol{x} = [x_1, \cdots, x_n]^T$ and $\boldsymbol{y} = [y_1, \cdots, y_n]^T$ is defined as

$$\boldsymbol{x} \cdot \boldsymbol{y} = <\boldsymbol{x}, \boldsymbol{y}> = \boldsymbol{x}^T\overline{\boldsymbol{y}} = \boldsymbol{y}^*\boldsymbol{x} = [x_1, \cdots, x_n]\begin{bmatrix} \overline{y}_1 \\ \vdots \\ \overline{y}_n \end{bmatrix} = \sum_{i=1}^{n} x_i\overline{y}_i \tag{11.13}$$

  where $\overline{u + jv} = u - jv$ is complex conjugate of $u + jv$. If the inner product of $\boldsymbol{x}$ and $\boldsymbol{y}$ is zero, then the two vectors are said to be *orthogonal*, denoted by $\boldsymbol{x} \perp \boldsymbol{y}$. If particular when $\boldsymbol{x} = \boldsymbol{y}$, we have:

$$\boldsymbol{x} \cdot \boldsymbol{x} = ||\boldsymbol{x}||^2 = \sum_{i=1}^{n} x_i\overline{x}_i = \sum_{i=1}^{n} |x_i|^2 > 0 \tag{11.14}$$

  where

$$||\boldsymbol{x}|| = \sqrt{\sum_{i=1}^{n} |x_i|^2} \tag{11.15}$$

  is called the *norm* of $\boldsymbol{x}$. When $||\boldsymbol{x}|| = 1$, $\boldsymbol{x}$ is *normalized*.
- **Matrix Multiplication**
  The product of an $m \times k$ matrix $\boldsymbol{A}$ and a $k \times n$ matrix $\boldsymbol{B}$ is

$$\boldsymbol{A}_{m\times k}\boldsymbol{B}_{k\times n} = \boldsymbol{C}_{m\times n} \tag{11.16}$$

  where the element in the ith row and jth column of $\boldsymbol{C}$ is the dot product of the ith row vector of $\boldsymbol{A}$ and the jth column of $\boldsymbol{B}$:

$$c_{ij} = [a_{i1}, \cdots, a_{ik}]\begin{bmatrix} b_{k1} \\ \vdots \\ b_{kn} \end{bmatrix} = \sum_{l=1}^{k} a_{il}b_{lj} \tag{11.17}$$

  For this multiplication to be possible, the number of columns of $\boldsymbol{A}$ must be equal to the number of rows of $\boldsymbol{B}$, so that the dot product can be carried out. Otherwise, the two matrices can not be multiplied.
- **Trace**
  The *trace* of $\boldsymbol{A}$ is defined as the sum of the element along the main diagonal:

$$tr(\boldsymbol{A}) = \sum_{i=1}^{n} a_{ii} \tag{11.18}$$

- **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 $\boldsymbol{A}$, denoted by $rank\boldsymbol{A}$, is the maximum number of linearly independent columns of $\boldsymbol{A}$, 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:

$$rank\boldsymbol{A} \leq min(m, n) \tag{11.19}$$

  If the equation holds, matrix $\boldsymbol{A}$ has a *full rank*.
- **Determinant**
  The *determinant* of an $n \times n$ matrix $\boldsymbol{A}$, denoted by $det\boldsymbol{A}$ or $|\boldsymbol{A}|$, is a scalar that can be recursively defined as

$$det\boldsymbol{A} = \sum_{j=1}^{n} (-1)^{j+1}\ a_{1j}\ det\boldsymbol{A}_{1j} \tag{11.20}$$

  where $\boldsymbol{A}_{1j}$ is an $n-1 \times n-1$ matrix obtained by deleting the first row and jth column of $\boldsymbol{A}$, and the determinant of a 1 by 1 matrix is $det(a) = a$. In particular, when $n = 2$,

$$det \begin{bmatrix} a & b \\ c & d \end{bmatrix} = ad - bc \tag{11.21}$$

  and when $n = 3$,

$$det \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} = a\ det \begin{bmatrix} e & f \\ h & i \end{bmatrix} - b\ det \begin{bmatrix} d & f \\ g & i \end{bmatrix} + c\ det \begin{bmatrix} d & e \\ g & h \end{bmatrix}$$
$$= aei - afh - bdi + bfg + cdh - ceg = (aei + bfg + cdh) - (gec + hfa + idb) \tag{11.22}$$

  Here are some important properties related to determinant:

$$det(\boldsymbol{AB}) = det\boldsymbol{A}\ det\boldsymbol{B}, \quad det(\boldsymbol{A}^T) = det\boldsymbol{A}, \quad det(c\boldsymbol{A}) = c^n det\boldsymbol{A} \tag{11.23}$$

- **Inverse Matrix**
  If $\boldsymbol{A}$ is an $n \times n$ square matrix and there exists another $n \times n$ matrix $\boldsymbol{B}$ so that $\boldsymbol{A}B = \boldsymbol{B}A = \boldsymbol{I}$, then $\boldsymbol{B} = \boldsymbol{A}^{-1}$ is the *inverse* of $\boldsymbol{A}$, which can be obtained by:

$$\boldsymbol{A}^{-1} = \frac{1}{det\boldsymbol{A}} \begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1n} \\ c_{21} & c_{22} & \cdots & c_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mn} \end{bmatrix}^T \tag{11.24}$$

  where c_{ij} is the ij-th *cofactor* defined as

$$c_{ij} = (-1)^{i+j}\ det\boldsymbol{\mu}_{ij} \tag{11.25}$$

with $\boldsymbol{\mu}_{ij}$ being an $n-1 \times n-1$ *minor matrix* obtained by removing the ith row and jth column $\boldsymbol{A}$. Obviously if $det\boldsymbol{A} = 0$, $\boldsymbol{A}^{-1}$ does not exist.
The following statements are equivalent:
– $\boldsymbol{A}$ is invertible, i.e., inverse matrix $\boldsymbol{A}^{-1}$ exists.
– $rank\boldsymbol{A} = n$ (full rank).
– $det\boldsymbol{A} \neq 0$.
– All column and row vectors are linearly independent.
– All eigenvalues of $\boldsymbol{A}$ are nonzero (to be discussed later).
These are some basic properties related to inverse of a matrix $\boldsymbol{A}$:

$$(\boldsymbol{A}^{-1})^{-1} = \boldsymbol{A}, \quad (c\boldsymbol{A})^{-1} = \frac{1}{c}\boldsymbol{A}^{-1}, \quad (\boldsymbol{A}B)^{-1} = \boldsymbol{B}^{-1}\boldsymbol{A}^{-1}, \quad (\boldsymbol{A}^{-1})^T = (\boldsymbol{A}^T)^{-1} \tag{11.26}$$

- **Pseudo-Inverse Matrix**
  Let $\boldsymbol{A}$ be an $m \times n$ matrix. If $m \neq n$, then $\boldsymbol{A}$ is not a square matrix and its inverse does not exist. However, we can find its *pseudo-inverse* $\boldsymbol{A}^{-}$, an $n \times m$ matrix, as shown below.
  – If $\boldsymbol{A}$ has more rows than columns, i.e., $m > n$, then

$$\boldsymbol{A}^{-} = (\boldsymbol{A}^{*}\boldsymbol{A})^{-1}\boldsymbol{A}^{*} \tag{11.27}$$

  We can verify that $\boldsymbol{A}^{-}\boldsymbol{A} = \boldsymbol{I}$:

$$\boldsymbol{A}^{-}\boldsymbol{A} = (\boldsymbol{A}^{*}\boldsymbol{A})^{-1}\boldsymbol{A}^{*}\boldsymbol{A} = \boldsymbol{I}_{n\times n} \tag{11.28}$$

  Note that $\boldsymbol{A}\boldsymbol{A}^{-} \neq \boldsymbol{I}$:
  – If $\boldsymbol{A}$ has more columns than rows, i.e., $m < n$, then

$$\boldsymbol{A}^{-} = \boldsymbol{A}^{*}(\boldsymbol{A}\boldsymbol{A}^{*})^{-1} \tag{11.29}$$

  We can verify that $\boldsymbol{A}\boldsymbol{A}^{-} = \boldsymbol{I}$:

$$\boldsymbol{A}\boldsymbol{A}^{-} = \boldsymbol{A}\boldsymbol{A}^{*}(\boldsymbol{A}\boldsymbol{A}^{*})^{-1} = \boldsymbol{I}_{m\times m} \tag{11.30}$$

  Note that $\boldsymbol{A}^{-}\boldsymbol{A} \neq \boldsymbol{I}$:
  Note that the pseudo-inverses in Eq.11.27 ($m > n$) and Eq.11.29 ($m < n$) are essentially the same. Assume $\boldsymbol{A}$ has more rows than columns ($m > n$), then another matrix defined as $\boldsymbol{B} = \boldsymbol{A}^{*}$ has more columns than rows. Taking conjugate transpose on both sides of Eq.11.27, we get:

$$(\boldsymbol{A}^{-})^{*} = (\boldsymbol{A}^{*})^{-1} = [(\boldsymbol{A}^{*}\boldsymbol{A})^{-1}\boldsymbol{A}^{*}]^{*} = \boldsymbol{A}(\boldsymbol{A}^{*}\boldsymbol{A})^{-1} \tag{11.31}$$

  i.e.,

$$\boldsymbol{B}^{-} = \boldsymbol{B}^{*}(\boldsymbol{B}\boldsymbol{B}^{*})^{-1} \tag{11.32}$$

  which is the same as Eq.11.29.

We can also show that $(\boldsymbol{A}^-)^- = \boldsymbol{A}$. If $m > n$, then we have:

$$\begin{aligned}(\boldsymbol{A}^-)^- &= [(\boldsymbol{A}^*\boldsymbol{A})^{-1}\boldsymbol{A}^*]^- = [(\boldsymbol{A}^*\boldsymbol{A})^{-1}\boldsymbol{A}^*]^* \left[(\boldsymbol{A}^*\boldsymbol{A})^{-1}\boldsymbol{A}^*[(\boldsymbol{A}^*\boldsymbol{A})^{-1}\boldsymbol{A}^*]^*\right]^{-1} \\ &= \boldsymbol{A}(\boldsymbol{A}^*\boldsymbol{A})^{-1}\left[(\boldsymbol{A}^*\boldsymbol{A})^{-1}\boldsymbol{A}^*\boldsymbol{A}(\boldsymbol{A}^*\boldsymbol{A})^{-1}\right]^{-1} \\ &= \boldsymbol{A}(\boldsymbol{A}^*\boldsymbol{A})^{-1}(\boldsymbol{A}^*\boldsymbol{A}) = \boldsymbol{A} \end{aligned} \tag{11.33}$$

Similarly we can show the same is true if $m < n$.
Specially when $m = n$, $\boldsymbol{A}$ is invertible and the pseudo-inverse in either Eq.11.27 or Eq.11.29 becomes the regular inverse $\boldsymbol{A}^- = \boldsymbol{A}^{-1}$.

11.2 Eigenvalues and Eigenvectors

For any $n \times n$ matrix $\boldsymbol{A}$, if there exists an n by 1 vector $\boldsymbol{\phi}$ and a scalar λ satisfying

$$\boldsymbol{A}_{n\times n}\boldsymbol{\phi}_{n\times 1} = \lambda\boldsymbol{\phi}_{n\times 1} \tag{11.34}$$

then λ and $\boldsymbol{\phi}$ are called the *eigenvalue* and *eigenvector* of $\boldsymbol{A}$, respectively. To obtain λ, we rewrite the above equation as

$$(\lambda\boldsymbol{I} - \boldsymbol{A})\boldsymbol{\phi} = 0 \tag{11.35}$$

This is a homogeneous algebraic equation system (of n equations) for n unknowns, the elements in vector $\boldsymbol{\phi}$. This equation system has non-zero solutions if and only if

$$det(\lambda\boldsymbol{I} - \boldsymbol{A}) = 0 \tag{11.36}$$

This nth order equation of λ is the *characteristic equation* of the matrix $\boldsymbol{A}$, which can be solved to get n solutions, the n eigen values $\{\lambda_1, \cdots, \lambda_n\}$ of $\boldsymbol{A}$. Substituting each λ_i back into the equation system, we can obtain the non-zero solution, the eigenvector $\boldsymbol{\phi_i}$corresponding to eigenvalue λ_i:

$$\boldsymbol{A}\boldsymbol{\phi}_i = \lambda_i\boldsymbol{\phi}_i, \qquad (i = 1, \cdots, n) \tag{11.37}$$

Putting all n such equations together, we get

$$\boldsymbol{A}\left[\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_n\right] = \left[\lambda_1\boldsymbol{\phi}_1, \cdots, \lambda_n\boldsymbol{\phi}_n\right] = \left[\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_n\right]\begin{bmatrix} \lambda_1 & 0 & \cdot & 0 \\ 0 & \lambda_2 & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & \cdot & \lambda_n \end{bmatrix} \tag{11.38}$$

Defining

$$\boldsymbol{\Phi} = \left[\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_n\right], \quad \text{and} \quad \boldsymbol{\Lambda} = diag\left[\lambda_1, \cdots, \lambda_n\right] \tag{11.39}$$

we can write the equation above in a more compact form:

$$\boldsymbol{A}\boldsymbol{\Phi} = \boldsymbol{\Phi}\boldsymbol{\Lambda}, \quad \text{or} \quad \boldsymbol{\Phi}^{-1}\boldsymbol{A}\boldsymbol{\Phi} = \boldsymbol{\Lambda} \tag{11.40}$$

The trace and determinant of $\boldsymbol{A}$ can be obtained in terms of its eigenvalues

$$tr\boldsymbol{A} = \sum_{k=1}^{n} \lambda_k, \qquad det\boldsymbol{A} = \prod_{k=1}^{n} \lambda_k \tag{11.41}$$

$\boldsymbol{A}^T$ has the same eigenvalues and eigenvectors as $\boldsymbol{A}$:

$$\boldsymbol{A}^T \boldsymbol{\phi}_i = \lambda_i \boldsymbol{\phi}_i, \quad (i = 1, \cdots, n) \tag{11.42}$$

$\boldsymbol{A}^m$ has the same eigenvectors as $\boldsymbol{A}$, but its eigenvalues are $\{\lambda_1^m, \cdots, \lambda_n^m\}$:

$$\boldsymbol{A}^m \boldsymbol{\phi}_i = \lambda_i^m \boldsymbol{\phi}_i, \quad (i = 1, \cdots, n) \tag{11.43}$$

where m is a positive integer. When $m = -1$, the relation still holds, i.e., the eigenvalues of $\boldsymbol{A}^{-1}$ are $\{1/\lambda_1, \cdots, 1/\lambda_n\}$:

$$\boldsymbol{A}^{-1} \boldsymbol{\phi}_i = \frac{1}{\lambda_i} \boldsymbol{\phi}_i, \quad (i = 1, \cdots, n) \tag{11.44}$$

A Hermitian matrix $\boldsymbol{A}$ is *positive definite*, denoted by $\boldsymbol{A} > 0$, if and only if for any nonzero $\boldsymbol{x} = [x_1, \cdots, x_n]^T$, the quadratic form $\boldsymbol{x}^* \boldsymbol{A} \boldsymbol{x}$ is greater than zero:

$$\boldsymbol{x}^* \boldsymbol{A} \boldsymbol{x} > 0 \tag{11.45}$$

In particular, if we let $\boldsymbol{x} = \boldsymbol{\phi}_i$ be eigenvector corresponding to the ith eigenvalue λ_i, then the above becomes:

$$\boldsymbol{\phi}_i^* \boldsymbol{A} \boldsymbol{\phi}_i = \lambda_i \boldsymbol{\phi}_i^* \boldsymbol{\phi}_i > 0 \tag{11.46}$$

as $\boldsymbol{\phi}_i^* \boldsymbol{\phi}_i > 0$, we know $\lambda_i > 0$ for all $i = 1, \cdots, n$, i.e., $\boldsymbol{A} > 0$ if and only if all of its eigenvalues are greater than zero. Also, as the eigenvalues of $\boldsymbol{A}^{-1}$ are $1/\lambda_i$, $i = (1, \cdots, n)$, we have $\boldsymbol{A} > 0$ if and only if $\boldsymbol{A}^{-1} > 0$.

11.3 Hermitian Matrix and Unitary Matrix

If a matrix $\boldsymbol{A}$ is equal to its *conjugate transpose*, i.e., $\boldsymbol{A}^* = \boldsymbol{A}$, then it is a *Hermitian matrix*. When a Hermitian matrix $\boldsymbol{A}$ is real ($\overline{\boldsymbol{A}} = \boldsymbol{A}$), it becomes a *symmetric matrix*, $\boldsymbol{A}^T = \boldsymbol{A}$. All eigenvalues of a Hermitian matrix are real. Eigenvectors corresponding to distinct eigenvalues are orthogonal.

$\boldsymbol{A}$ is a *unitary matrix* if and only if $\boldsymbol{A}^* \boldsymbol{A} = \boldsymbol{I}$, i.e., $\boldsymbol{A}^* = \boldsymbol{A}^{-1}$. When a unitary matrix $\boldsymbol{A}$ is real ($\overline{\boldsymbol{A}} = \boldsymbol{A}$), it becomes an *orthogonal matrix*, $\boldsymbol{A}^T = \boldsymbol{A}^{-1}$. The eigenvalues of a unitary matrix are complex numbers of absolute value 1 (i.e. they lie on the unit circle centered at 0 in the complex plane). The determinant of a unitary matrix $\boldsymbol{A}$ is

$$det\boldsymbol{A} = \pm 1 \tag{11.47}$$

The columns (and rows) of a unitary matrix $\boldsymbol{A}$ are *orthonormal*, i.e. they are both orthogonal and normalized:

$$< \boldsymbol{a}_i, \boldsymbol{a}_j >= \sum_k a_{ik}\overline{a}_{jk} = \delta_{ij} = \begin{cases} 1 \text{ if } i = j \\ 0 \text{ else} \end{cases} \tag{11.48}$$

where $\boldsymbol{a}_i$ and $\boldsymbol{a}_j$ are the ith and jth columns of $\boldsymbol{A}$, respectively.

Consider the eigenvalues and eigenvectors of a Hermitian matrix (symmetric if real) $\boldsymbol{A}$:

$$\boldsymbol{A\Phi} = \boldsymbol{\Phi\Lambda}, \quad \text{i.e.,} \quad \boldsymbol{\Phi}^{-1}\boldsymbol{A\Phi} = \boldsymbol{\Lambda} \tag{11.49}$$

where $\boldsymbol{\Phi} = [\phi_1, \cdots, \phi_n]$. As the eigenvectors corresponding to distinct eigenvalues are orthogonal, $\boldsymbol{\Phi}$ is unitary, i.e., $\boldsymbol{\Phi}^{-1} = \boldsymbol{\Phi}^*$ and we have

$$\boldsymbol{\Phi}^{-1}\boldsymbol{A\Phi} = \boldsymbol{\Phi}^*\boldsymbol{A\Phi} = \boldsymbol{\Lambda} \tag{11.50}$$

In other words, a Hermitian matrix $\boldsymbol{A}$ can be diagonalized to become $\boldsymbol{\Lambda}$ by its unitary eigenvector matrix $\boldsymbol{\Phi}$.

Based on any unitary matrix $\boldsymbol{A} = [\boldsymbol{a}_1 \cdots, \boldsymbol{a}_n]$ (where the jth column vector is $\boldsymbol{a}_j = [a_{1j}, \cdots, a_{nj}]^T$), a *unitary transform* of a vector $\boldsymbol{x} = [x_1, \cdots, x_n]^T$ can be defined:

$$\begin{cases} \boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \boldsymbol{Ay} = \begin{bmatrix} \boldsymbol{a}_1 \, \boldsymbol{a}_2 \cdots \boldsymbol{a}_n \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \sum_{j=1}^n y_j \, \boldsymbol{a}_j & \text{(inverse transform)} \\ \boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \boldsymbol{A}^{-1}\boldsymbol{x} = \boldsymbol{A}^*\boldsymbol{x} = \begin{bmatrix} \boldsymbol{a}_1^* \\ \boldsymbol{a}_2^* \\ \vdots \\ \boldsymbol{a}_n^* \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} & \text{(forward transform)} \end{cases} \tag{11.51}$$

In particular, when $\boldsymbol{A} = \overline{\boldsymbol{A}}$ is real, $\boldsymbol{A}^{-1} = \boldsymbol{A}^T$ is an orthogonal matrix and the corresponding transform is an *orthogonal transform*.

The forward transform can also be written as component form:

$$y_j = \boldsymbol{a}_j^*\boldsymbol{x} = \sum_{i=1}^n \overline{a}_{ij}x_i, \quad (j = 1, \cdots, n) \tag{11.52}$$

where the transform coefficient $y_j = \boldsymbol{a}_j^*\boldsymbol{x}$ is the dot product of the two vectors, representing the projection of vector $\boldsymbol{x}$ onto the ith column vector $\boldsymbol{a}_i$ of the transform matrix $\boldsymbol{A}$. The *inverse transform* can also be written as:

$$\boldsymbol{x} = \sum_{j=1}^n y_j \, \boldsymbol{a}_j \quad \text{or in component form:} \quad x_i = \sum_{j=1}^n a_{ij} \, y_j \quad (i = 1, \cdots, n) \tag{11.53}$$

By this transform, vector $\boldsymbol{x}$ is represented as a linear combination (weighted sum) of the n column vectors $\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_n$ of matrix $\boldsymbol{A}$. Geometrically, $\boldsymbol{x}$ is a

point in the n-dimensional space spanned by these n orthonormal basis vectors. Each coefficient (coordinate) y_i is the projection of $\boldsymbol{x}$ onto the corresponding basis vector $\boldsymbol{a}_i$.

The n-dimensional space can be spanned by the column vectors of *any* n by n unitary (or orthogonal) matrix, and a vector $\boldsymbol{x}$ in the space can be represented by any of such matrices, each defining a different transform.

- When $\boldsymbol{A} = \boldsymbol{I} = [\boldsymbol{e}_1, \cdots, \boldsymbol{e}_n]$ is an identity matrix, the transform becomes:

$$\boldsymbol{x} = \sum_{j=1}^{n} y_j \boldsymbol{a}_j = \sum_{j=1}^{n} x_j \boldsymbol{e}_j \tag{11.54}$$

  where $\boldsymbol{e}_i = [0, \cdots, 0, 1, 0, \cdots, 0]^T$ is the ith column of $\boldsymbol{I}$ with the ith element equal 1 and all other 0.
- When $a_{m,n} = w[m,n] = e^{-j2\pi mn/N}$, the corresponding transform is the discrete Fourier transform. The nth column vector $\boldsymbol{w}_n$ of the transform matrix $\boldsymbol{W} = [\boldsymbol{w}_0, \cdots, \boldsymbol{w}_{N-1}]$ represents a sinusoid of a frequency nf_0, and the corresponding complex coordinate $y_n = \boldsymbol{w}_n^* \boldsymbol{x}$ represents the magnitude $|y_n|$ and phase $\angle y_n$ of this nth frequency component. The Fourier transform $\boldsymbol{y} = \boldsymbol{W}\boldsymbol{x}$ represents a rotation of the coordinate system.

A unitary (orthogonal) transform $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$ 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 $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$ does not change the vector's length:

$$||\boldsymbol{y}||^2 = \boldsymbol{y}^*\boldsymbol{y} = (\boldsymbol{A}^*\boldsymbol{x})^*(\boldsymbol{A}^*\boldsymbol{x}) = \boldsymbol{x}^*\boldsymbol{A}A^*\boldsymbol{x} = \boldsymbol{x}^*\boldsymbol{x} = ||\boldsymbol{x}||^2 \tag{11.55}$$

as $\boldsymbol{A}\boldsymbol{A}^* = \boldsymbol{A}\boldsymbol{A}^{-1} = \boldsymbol{I}$. This is the Parseval's relation. If $\boldsymbol{x}$ is interpreted as a signal, then its length $||\boldsymbol{x}||^2 = ||\boldsymbol{y}||^2$ represents the total energy or information contained in the signal, which is preserved during any unitary transform. However, some other features of the signal may be changed, e.g., the signal may be decorrelated after the transform, which may be desirable in many applications.

If $\boldsymbol{x}$ is a random vector with mean vector $\boldsymbol{\mu}_x$ and covariance matrix $\boldsymbol{\Sigma}_x$:

$$\boldsymbol{\mu}_x = E(\boldsymbol{x}), \qquad \boldsymbol{\Sigma}_x = E(\boldsymbol{x}\boldsymbol{x}^*) - \boldsymbol{\mu}_x\boldsymbol{\mu}_x^* \tag{11.56}$$

then its transform $\boldsymbol{y} = \boldsymbol{A}^*\boldsymbol{x}$ has the following mean vector and covariance matrix:

$$\boldsymbol{\mu}_y = E(\boldsymbol{y}) = E(\boldsymbol{A}^*\boldsymbol{x}) = \boldsymbol{A}^*E(\boldsymbol{x}) = \boldsymbol{A}^*\boldsymbol{\mu}_x \tag{11.57}$$

$$\begin{aligned}\boldsymbol{\Sigma}_y &= E(\boldsymbol{y}\boldsymbol{y}^*) - \boldsymbol{\mu}_y\boldsymbol{\mu}_y^* = E[(\boldsymbol{A}^*\boldsymbol{x})(\boldsymbol{A}^*\boldsymbol{x})^*] - (\boldsymbol{A}^*\boldsymbol{\mu}_x)(\boldsymbol{A}^*\boldsymbol{\mu}_x)^* \\ &= E[\boldsymbol{A}^*(\boldsymbol{x}\boldsymbol{x}^*)\boldsymbol{A}] - \boldsymbol{A}^*\boldsymbol{\mu}_x\boldsymbol{\mu}_x^*\boldsymbol{A} = \boldsymbol{A}^*[E(\boldsymbol{x}\boldsymbol{x}^*) - \boldsymbol{\mu}_x\boldsymbol{\mu}_x^*]\boldsymbol{A} \\ &= \boldsymbol{A}^*\boldsymbol{\Sigma}_x\boldsymbol{A}\end{aligned} \tag{11.58}$$

If $\boldsymbol{A}$ is Hermitian (symmetric if $\boldsymbol{A}$ is real), then all of its eigenvalues λ_i' are real and all eigenvectors $\boldsymbol{\phi}_i$ are orthogonal:

$$\boldsymbol{\phi}_i^*\boldsymbol{\phi}_j = \delta_{ij} \tag{11.59}$$

If all $\boldsymbol{\phi}_i$'s are normalized, matrix $\boldsymbol{\Phi}$ is unitary (orthogonal if $\boldsymbol{A}$ is real):

$$\boldsymbol{\Phi}^{-1} = \boldsymbol{\Phi}^* \tag{11.60}$$

and we have

$$\boldsymbol{\Phi}^{-1}\boldsymbol{A}\boldsymbol{\Phi} = \boldsymbol{\Phi}^*\boldsymbol{A}\boldsymbol{\Phi} = \boldsymbol{\Lambda} \tag{11.61}$$

On the other hand, the matrix $\boldsymbol{A}$ can be decomposed to be expressed as

$$\boldsymbol{A} = \boldsymbol{\Phi}\boldsymbol{\Lambda}\boldsymbol{\Phi}^* = [\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_n] \begin{bmatrix} \lambda_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \lambda_n \end{bmatrix} \begin{bmatrix} \boldsymbol{\phi}_1^* \\ \vdots \\ \boldsymbol{\phi}_n^* \end{bmatrix} = \sum_{i=1}^{n} \lambda_i \boldsymbol{\phi}_i \boldsymbol{\phi}_i^* \tag{11.62}$$

11.4 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:

$$A_T = \begin{bmatrix} a & b & c & d & e & f \\ g & a & b & c & d & e \\ h & g & a & b & c & d \\ i & h & g & a & b & c \\ j & i & h & g & a & b \\ k & j & i & h & g & a \end{bmatrix} \tag{11.63}$$

An N by N Toeplitz matrix can be formed by a sequence $\cdots x_{-2}, x_{-1}, x_0, x_1, x_2, \cdots$:

$$A_T = \begin{bmatrix} a_0 & a_1 & a_2 & \cdots & a_{N-3} & a_{N-2} & a_{N-1} \\ a_{-1} & a_0 & a_1 & \cdots & a_{N-4} & a_{N-3} & a_{N-2} \\ a_{-2} & a_{-1} & a_0 & \cdots & a_{N-5} & a_{N-4} & a_{N-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ a_{3-N} & a_{4-N} & a_{5-N} & \cdots & a_0 & a_1 & a_2 \\ a_{2-N} & a_{3-N} & a_{4-N} & \cdots & a_{-1} & a_0 & a_1 \\ a_{1-N} & a_{2-N} & a_{3-N} & \cdots & a_{-2} & a_{-1} & a_0 \end{bmatrix} \tag{11.64}$$

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:

$$A_T = \begin{bmatrix} a_0 & a_1 & a_2 \cdots & a_{N-3} & a_{N-2} & a_{N-1} \\ a_{N-1} & a_0 & a_1 \cdots & a_{N-4} & a_{N-3} & a_{N-2} \\ a_{N-2} & a_{N-1} & a_0 \cdots & a_{N-5} & a_{N-4} & a_{N-3} \\ \vdots & \vdots & \vdots \ddots & \vdots & \vdots & \vdots \\ a_3 & a_4 & a_5 \cdots & a_0 & a_1 & a_2 \\ a_2 & a_3 & a_4 \cdots & a_{N-1} & a_0 & a_1 \\ a_1 & a_2 & a_3 \cdots & a_{N-2} & a_{N-1} & a_0 \end{bmatrix} \tag{11.65}$$

When the period N of the sequence is increased to approach infinity $N \to \infty$, the periodic sequence approaches aperiodic, correspondingly, the circulant matrix asymptotically becomes a Toeplitz matrix.

11.5 Vector and Matrix Differentiation

A vector differentiation operator is defined as

$$\frac{d}{d\boldsymbol{x}} = [\frac{\partial}{\partial x_1}, \cdots, \frac{\partial}{\partial x_n}]^T \tag{11.66}$$

which can be applied to any scalar function $f(\boldsymbol{x})$ to find its derivative with respect to $\boldsymbol{x}$:

$$\frac{d}{d\boldsymbol{x}} f(\boldsymbol{x}) = [\frac{\partial f}{\partial x_1}, \cdots, \frac{\partial f}{\partial x_n}]^T \tag{11.67}$$

Vector differentiation has the following properties:

$$\frac{d}{d\boldsymbol{x}}(\boldsymbol{b}^T\boldsymbol{x}) = \frac{d}{d\boldsymbol{x}}(\boldsymbol{x}^T\boldsymbol{b}) = \boldsymbol{b} \tag{11.68}$$

$$\frac{d}{d\boldsymbol{x}}(\boldsymbol{x}^T\boldsymbol{x}) = 2\boldsymbol{x} \tag{11.69}$$

$$\frac{d}{d\boldsymbol{x}}(\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x}) = 2\boldsymbol{A}\boldsymbol{x} \quad (if\ \boldsymbol{A}^T = \boldsymbol{A}) \tag{11.70}$$

To prove the third one, consider the kth element of the vector:

$$\frac{\partial}{\partial x_k}(\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x}) = \frac{\partial}{\partial x_k}\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}x_i x_j = \sum_{i=1}^{n} a_{ik}x_i + \sum_{j=1}^{n} a_{kj}x_j = 2\sum_{i=1}^{n} a_{ik}x_i \tag{11.71}$$

for $(k = 1, \cdots, n)$.

Note that here we have used the assumption that $a_{ik} = a_{ki}$, i.e., $\boldsymbol{A}^T = \boldsymbol{A}$. Putting all n elements in vector form, we have the above.

When $\boldsymbol{A} = \boldsymbol{I}$, we have

$$\frac{d}{d\boldsymbol{x}}(\boldsymbol{x}^T\boldsymbol{x}) = 2\boldsymbol{x} \tag{11.72}$$

You can compare these results with the familiar derivatives in the scalar case:

$$\frac{d}{dx}(ax^2) = 2ax \tag{11.73}$$

A matrix differentiation operator is defined as

$$\frac{d}{d\boldsymbol{A}} = \begin{bmatrix} \frac{\partial}{\partial a_{11}} & \cdots & \frac{\partial}{\partial a_{1n}} \\ \vdots & \ddots & \vdots \\ \frac{\partial}{\partial a_{m1}} & \cdots & \frac{\partial}{\partial a_{mn}} \end{bmatrix} \tag{11.74}$$

which can be applied to any scalar function of $f(\boldsymbol{A})$:

$$\frac{d}{d\boldsymbol{A}} f(\boldsymbol{A}) = \begin{bmatrix} \frac{\partial}{\partial a_{11}} f(\boldsymbol{A}) & \cdots & \frac{\partial}{\partial a_{1n}} f(\boldsymbol{A}) \\ \vdots & \ddots & \vdots \\ \frac{\partial}{\partial a_{m1}} f(\boldsymbol{A}) & \cdots & \frac{\partial}{\partial a_{mn}} f(\boldsymbol{A}) \end{bmatrix} \tag{11.75}$$

Specifically, consider $f(\boldsymbol{A}) = \boldsymbol{u}^T \boldsymbol{A} \boldsymbol{v}$, where $\boldsymbol{u}$ and $\boldsymbol{v}$ are $m \times 1$ and $n \times 1$ constant vectors, respectively, and $\boldsymbol{A}$ is an $m \times n$ matrix. Then we have:

$$\frac{d}{d\boldsymbol{A}}(\boldsymbol{u}^T \boldsymbol{A} \boldsymbol{v}) = \boldsymbol{u}\boldsymbol{v}^T \tag{11.76}$$

12 Appendix 2: Review of Random Variables

12.1 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 $A_i \in \mathcal{F}$ $(i = 1, 2, \cdots)$, 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 $(\Omega, \mathcal{F})$ 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 $(\Omega, \mathcal{F}, P)$ is called the *probability space*.

- **Random Variables**
  A random variable $x(\omega)$ is a real-valued function $x : \Omega \to \mathbb{R}$ that maps every outcome $\omega \in \Omega$ into a real number x. Formally, the function $x(\omega)$ is a random variable if

$$\{\omega : x(\omega) \leq r\} \in \mathcal{F}, \quad \forall r \in \mathbb{R} \tag{12.1}$$

  Random variables x can be either continuous or discrete.
- **Cumulative Distribution Function**
  The *cumulative distribution function* of a random variable x is defined as

$$F_x(\xi) = P(x < \xi) \tag{12.2}$$

  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

$$p_x(\xi) = \frac{d}{d\xi} F_x(\xi), \quad \text{i.e.,} \quad F_x(u) = \int_{-\infty}^{u} p_x(\xi) d\xi \tag{12.3}$$

  We have

$$P(a \leq x < b) = F_x(b) - F_x(a) = \int_a^b p_x(\xi) d\xi \tag{12.4}$$

  In particular

$$P(x < \infty) = F_x(\infty) - F_x(-\infty) = \int_{-\infty}^{\infty} p_x(\xi) d\xi = 1 \tag{12.5}$$

  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 $\{x_i \quad i = 1, \cdots, n\}$, then its *probability distribution* is

$$P(x = x_i) = p_i \quad (i = 1, \cdots, n) \tag{12.6}$$

  where

$$0 \leq p_i \leq 1, \quad \text{and} \quad \sum_{i=1}^{n} p_i = 1 \tag{12.7}$$

  The cumulative distribution function is

$$F_x(\xi) = P(x < \xi) = \sum_{x_i < \xi} p_i \tag{12.8}$$

- **Expectation**
  The *expectation* is the mathematical mean of a random variable x. If x is continuous,

$$\mu_x = E(x) = \int_{-\infty}^{\infty} \xi \, p(\xi) d\xi \tag{12.9}$$

If x is discrete,

$$\mu_x = E(x) = \sum_{i=1}^{n} x_i p_i \tag{12.10}$$

- **Variance**
  The *variance* represents the statistical variability of a random variable x. If x is continuous,

$$\sigma_x^2 = Var(x) = E[(x-\mu_x)^2] = \int_{-\infty}^{\infty} (x-\mu_x)^2 p(x) dx \tag{12.11}$$

If x is discrete,

$$\sigma_x^2 = Var(x) = E[(x-\mu_x)^2] = \sum_{i=1}^{n} (x_i-\mu_x)^2 p_i \tag{12.12}$$

We also have

$$\sigma_x^2 = Var(x) = E[(x-\mu_x)^2] = E(x^2) - 2\mu_x E(x) + \mu_x^2 = E(x^2) - \mu_x^2 \tag{12.13}$$

The *standard deviation* of x is defined as

$$\sigma_x = \sqrt{Var(x)} \tag{12.14}$$

- **Normal (Gaussian) Distribution**
  Random variable x has a *normal distribution* if its density function is

$$p(x) = N(x, \mu_x, \sigma_x) = \frac{1}{\sqrt{2\pi\sigma_x^2}} e^{-\frac{(x-\mu_x)^2}{\sigma_x^2}} \tag{12.15}$$

It can be shown that

$$\int_{-\infty}^{\infty} N(x, \mu_x, \sigma_x) dx = 1 \tag{12.16}$$

$$E(x) = \int_{-\infty}^{\infty} x\, N(x, \mu_x, \sigma_x) dx = \mu_x \tag{12.17}$$

and

$$Var(x) = \int_{-\infty}^{\infty} (x-\mu_x)^2 N(x, \mu_x, \sigma_x) dx = \sigma_x^2 \tag{12.18}$$

12.2 Multivariate Random Variables

- **Multivariate Random Variables**
  A *multivariate random variable* or *random vector* is a vector $\boldsymbol{x} = [x_1, \cdots, x_n]^T$ with each component x_i being a random variable. When a *stochastic process* or *random process* (discussed later) $x(t)$ is sampled, its discrete time samples can be considered as a random vector $\boldsymbol{x}$.

- **Joint Distribution Function and Density Function**
  The *joint distribution function* of a random vector $\boldsymbol{x}$ is defined as

$$\begin{aligned} F_x(u_1, \cdots, u_n) &= P(x_1 < u_1, \cdots, x_n < u_n) \\ &= \int_{-\infty}^{u_1} \cdots \int_{-\infty}^{u_n} p(\xi_1, \cdots, \xi_n) d\xi_1 \cdots d\xi_n \end{aligned} \tag{12.19}$$

  where $p(\xi_1, \cdots, \xi_n)$ is the *joint density function* of the random vector $\boldsymbol{x}$.
- **Independent Variables**
  A set of n random variables are independent if

$$p(x_1, \cdots, x_n) = p(x_1)\, p(x_2) \cdots p(x_n) \tag{12.20}$$

- **Mean Vector**
  The *expectation* or *mean* of random variable x_i is defined as

$$\mu_i = E(x_i) = \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \xi_i\, p(\xi_1, \cdots, \xi_n)\, d\xi_1 \cdots d\xi_n \tag{12.21}$$

  The *mean vector* of random vector $\boldsymbol{x}$ is defined as

$$\boldsymbol{\mu}_x = E(\boldsymbol{x}) = [E(x_1), \cdots, E(x_n)]^T = [\mu_1, \cdots, \mu_n]^T \tag{12.22}$$

  which can be interpreted as the center of gravity of an n-dimensional object with $p(x_1, \cdots, x_n)$ being the density function.
- **Covariance Matrix**
  The *variance* of random variable x_i measures its variability and is defined as

$$\begin{aligned} \sigma_i^2 &= Var(x_i) = E[(x_i - \mu_i)^2] = E(x_i^2) - \mu_i^2 \\ &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} (\xi_i - \mu_i)^2\, p(\xi_1, \cdots, \xi_n)\, d\xi_1 \cdots d\xi_n \end{aligned} \tag{12.23}$$

  The *covariance* of x_i and x_j measures their similarity and is defined as

$$\begin{aligned} \sigma_{ij}^2 &= Cov(x_i, x_j) = E[(x_i - \mu_i)(x_j - \mu_j)] = E(x_i x_j) - \mu_i \mu_j \\ &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} (\xi_i - \mu_i)(\xi_j - \mu_j)\, p(\xi_1, \cdots, \xi_n)\, d\xi_1 \cdots d\xi_n - \mu_i \mu_j \end{aligned} \tag{12.24}$$

$$\tag{12.25}$$

  The *covariance matrix* of a random vector $\boldsymbol{x}$ is defined as

$$\begin{aligned} \boldsymbol{\Sigma}_x &= E[(\boldsymbol{x} - \boldsymbol{\mu}_x)(\boldsymbol{x} - \boldsymbol{\mu}_x)^T] = E(\boldsymbol{x}\boldsymbol{x}^T) - \boldsymbol{\mu}_x \boldsymbol{\mu}_x^T \\ &= \begin{bmatrix} \sigma_{11}^2 & \cdots & \sigma_{1n}^2 \\ \vdots & \ddots & \vdots \\ \sigma_{n1}^2 & \cdots & \sigma_{nn}^2 \end{bmatrix}_{n \times n} \end{aligned} \tag{12.26}$$

  When $i = j$, $\sigma_i^2 = E(x_i^2) - \mu_i^2$ is the variance of x_i, which can be interpreted as the amount of information, or energy, contained in the ith component x_i

of the signal $\boldsymbol{x}$. Therefore the total information or energy contained in $\boldsymbol{x}$ is:

$$tr\ \boldsymbol{\Sigma}_x = \sum_{i=1}^{n} \sigma_i^2 \tag{12.27}$$

$\boldsymbol{\Sigma}$ is symmetric as $\sigma_{ij}^2 = \sigma_{ji}^2$. Moreover, it can be shown that $\boldsymbol{\Sigma}$ is also *positive definite*, and all its eigenvalues $\{\lambda_1, \cdots, \lambda_n\}$ are greater than zero and we have

$$tr\ \boldsymbol{\Sigma}_x = \sum_{i=1}^{n} \lambda_i > 0, \quad \text{and} \quad det\ \boldsymbol{\Sigma}_x = \prod_{i=1}^{n} \lambda_i > 0 \tag{12.28}$$

- **Correlation Coefficient**
  The covariance σ_{ij}^2 of two random variables x_i and x_j represents the statistical similarity between them. If $\sigma_{ij}^2 > 0$, x_i and x_j are positively correlated; if $\sigma_{ij}^2 < 0$, they are negatively correlated, if $\sigma_{ij}^2 = 0$, they are *uncorrelated* or *decorrelated*. The normalized covariance is called the *correlation coefficient*:

$$r_{ij} = \frac{\sigma_{ij}^2}{\sigma_i \sigma_j} = \frac{E(x_i x_j) - \mu_i \mu_j}{\sqrt{E(x_i^2) - \mu_i^2}\sqrt{E(x_j^2) - \mu_j^2}} \tag{12.29}$$

  Now if the two variables are identical, i.e., $x_i = x_j$, then $c_{ij} = 1$, indicating they are one hundred percent similar to each other.
  A random vector $\boldsymbol{x} = [x_1, \cdots, x_n]^T$ is said to be decorrelated if $r_{ij} = 0$ for all $i \neq j$, and its covariance matrix $\boldsymbol{\Sigma}$ becomes a diagonal matrix with only non-zero σ_i^2 $(i = 1, \cdots, n)$ on its diagonal.
  If x_i $(i = 1, \cdots, n)$ are independent, i.e., $p(x_1, \cdots, x_n) = p(x_1) \cdots p(x_n)$, then they are also uncorrelated, i.e., $E(x_1, \cdots, x_n) = E(x_1) \cdots E(x_n)$. On the other hand, uncorrelated variables are not necessarily independent. (But uncorrelated variables with normal distribution are also independent.)
  Note that the term corration is also used to describe the similarity of two deterministic time functions $x(t)$ and $y(t)$
- **Mean and Covariance under Orthogonal Transforms**
  If the inverse of a matrix is the same as its transpose: $\boldsymbol{A}^{-1} = \boldsymbol{A}^T$, then it is an orthogonal matrix. Given any orthogonal matrix $\boldsymbol{A}$, an orthogonal transform of a random vector $\boldsymbol{x}$ can be defined as

$$\begin{cases} \boldsymbol{y} = \boldsymbol{A}^T \boldsymbol{x} \\ \boldsymbol{x} = \boldsymbol{A}\boldsymbol{y} \end{cases} \tag{12.30}$$

  The mean vector $\boldsymbol{\mu}_y$ and the covariance matrix $\boldsymbol{\Sigma}_y$ of $\boldsymbol{y}$ are related to the $\boldsymbol{\mu}_x$ and $\boldsymbol{\Sigma}_x$ of $\boldsymbol{x}$ by:

$$\boldsymbol{\mu}_y = E(\boldsymbol{y}) = E(\boldsymbol{A}^T \boldsymbol{x}) = \boldsymbol{A}^T E(\boldsymbol{x}) = \boldsymbol{A}^T \boldsymbol{\mu}_x \tag{12.31}$$

$$\begin{aligned} \boldsymbol{\Sigma}_y &= E(\boldsymbol{y}\boldsymbol{y}^T) - \boldsymbol{\mu}_y \boldsymbol{\mu}_y^T = E(\boldsymbol{A}^T \boldsymbol{x}\boldsymbol{x}^T \boldsymbol{A}) - \boldsymbol{A}^T \boldsymbol{\mu}_x \boldsymbol{\mu}_x^T \boldsymbol{A} \\ &= \boldsymbol{A}^T E(\boldsymbol{x}\boldsymbol{x}^T)\boldsymbol{A} - \boldsymbol{A}^T \boldsymbol{\mu}_x \boldsymbol{\mu}_x^T \boldsymbol{A} = \boldsymbol{A}^T [E(\boldsymbol{x}\boldsymbol{x}^T) - \boldsymbol{\mu}_x \boldsymbol{\mu}_x^T]\boldsymbol{A} \\ &= \boldsymbol{A}^T \boldsymbol{\Sigma}_x \boldsymbol{A} \end{aligned} \tag{12.32}$$

Orthogonal transform does not change the trace of $\mathbf{\Sigma}$:

$$\begin{aligned} tr\ \mathbf{\Sigma}_y &= tr\ [E(\boldsymbol{y}\boldsymbol{y}^T) - \boldsymbol{\mu}_y\boldsymbol{\mu}_y^T] = E[tr\ (\boldsymbol{y}\boldsymbol{y}^T)] - tr\ (\boldsymbol{\mu}_y\boldsymbol{\mu}_y^T) \\ &= E(\boldsymbol{y}^T\boldsymbol{y}) - \boldsymbol{\mu}_y^T\boldsymbol{\mu}_y = E(\boldsymbol{x}^T\boldsymbol{A}A^T\boldsymbol{x}) - \boldsymbol{\mu}_x^T\boldsymbol{A}A^T\boldsymbol{\mu}_x \\ &= E(\boldsymbol{x}^T\boldsymbol{x}) - \boldsymbol{\mu}_x^T\boldsymbol{\mu}_x = tr\ \mathbf{\Sigma}_x \end{aligned} \tag{12.33}$$

which means the total amount of energy or information contained in $\boldsymbol{x}$ is not changed after a unitary transform $\boldsymbol{y} = \boldsymbol{A}^T\boldsymbol{x}$ (although the distribution of energy among the components may be changed).

- **Normal Distribution**

The density function of a normally distributed random vector $\boldsymbol{x}$ is:

$$p(\boldsymbol{x}) = N(\boldsymbol{x}, \boldsymbol{\mu}_x, \mathbf{\Sigma}_x) = \frac{1}{(2\pi)^{n/2}\left|\mathbf{\Sigma}\right|^{1/2}} exp[-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu}_x)^T\mathbf{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_x)] \tag{12.34}$$

When $n = 1$, $\mathbf{\Sigma}_x$ 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

$$N(\boldsymbol{x}, \boldsymbol{\mu}_x, \mathbf{\Sigma}_x) = c_0 \tag{12.35}$$

where c_0 is a constant. This equation can be written as

$$(\boldsymbol{x} - \boldsymbol{\mu}_x)^T\mathbf{\Sigma}_x^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_x) = c_1 \tag{12.36}$$

where c_1 is another constant related to c_0, $\boldsymbol{\mu}_x$ and $\mathbf{\Sigma}_x$. For $n = 2$ variables x and y, we have

$$\begin{aligned} (\boldsymbol{x} - \boldsymbol{\mu}_x)^T\mathbf{\Sigma}_x^{-1}(\boldsymbol{x} - \boldsymbol{\mu}_x) &= [x_1 - \mu_1, x_2 - \mu_2] \begin{bmatrix} a & b/2 \\ b/2 & c \end{bmatrix} \begin{bmatrix} x_1 - \mu_1 \\ x - \mu_2 \end{bmatrix} \\ = a(x_1 - \mu_1)^2 + b(x_1 - \mu_1)(x_2 - \mu_2) + c(x_2 - \mu_2)^2 &= c_1 \end{aligned} \tag{12.37}$$

Here we have assumed

$$\begin{bmatrix} a & b/2 \\ b/2 & c \end{bmatrix} = \Sigma^{-1} \tag{12.38}$$

The above quadratic equation represents an ellipse (instead of any other quadratic curve) centered at $\boldsymbol{\mu}_x = [\mu_1, \mu_2]^T$, because $\mathbf{\Sigma}_x^{-1}$, as well as $\mathbf{\Sigma}_x$, is positive definite:

$$\left|\Sigma^{-1}\right| = ac - b^2/4 > 0 \tag{12.39}$$

When $n > 2$, the equation $N(\boldsymbol{x}, \boldsymbol{\mu}_x, \mathbf{\Sigma}_x) = c_0$ represents a hyper ellipsoid in the n-dimensional space. The center and spatial distribution of this ellipsoid are determined by $\boldsymbol{\mu}_x$ and $\mathbf{\Sigma}_x$, respectively.

In particular, when $\boldsymbol{x} = [x_1, \cdots, x_n]^T$ is decorrelated, i.e., $\sigma_{ij} = 0 \ \ for\ all \ \ i \neq j$, $\boldsymbol{\Sigma}_x$ becomes a diagonal matrix

$$\boldsymbol{\Sigma}_x = diag[\sigma_1^2, \cdots, \sigma_n^2] = \begin{bmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_n^2 \end{bmatrix} \tag{12.40}$$

and equation $N(\boldsymbol{x}, \boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x) = c_0$ can be written as

$$(\boldsymbol{x} - \boldsymbol{\mu}_x)^T \boldsymbol{\Sigma}_x^{-1} (\boldsymbol{x} - \boldsymbol{\mu}_x) = \sum_{i=1}^{n} \frac{(x_i - \mu_i)^2}{\sigma_i^2} = c_1 \tag{12.41}$$

which represents a standard ellipsoid with all its axes parallel to those of the coordinate system.

- **Estimation of $\boldsymbol{\mu}_x$ and $\boldsymbol{\Sigma}_x$**
  When $p(\boldsymbol{x}) = p(x_1, \cdots, x_n)$ is not known, $\boldsymbol{\mu}_x$ and $\boldsymbol{\Sigma}_x$ cannot be found by their definitions, but they can be estimated if a set of K outcomes ($\boldsymbol{x}^{(k)}, \ \ k = 1, \cdots, K$) of the random experiment can be observed. Then the mean vector can be estimated as

$$\hat{\boldsymbol{\mu}}_x = \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{x}^{(k)} \tag{12.42}$$

  i.e., the ith element of $\hat{\boldsymbol{\mu}}_x$ is estimated as

$$\hat{\mu}_i = \frac{1}{K} \sum_{k=1}^{K} x_i^{(k)}, \qquad (i = 1, \cdots, n) \tag{12.43}$$

  where $x_i^{(k)}$ is the kth element of $\hat{\boldsymbol{x}}_k$. The covariance matrix $\boldsymbol{\Sigma}_x$ can be estimated as

$$\hat{\boldsymbol{\Sigma}}_x = \frac{1}{K} \sum_{k=1}^{K} (\boldsymbol{x}^{(k)} - \hat{\boldsymbol{\mu}}_x)(\boldsymbol{x}^{(k)} - \hat{\boldsymbol{\mu}}_x)^T = \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{x}^{(k)} \boldsymbol{x}^{(k)T} - \hat{\boldsymbol{\mu}}_x \hat{\boldsymbol{\mu}}_x^T \tag{12.44}$$

  i.e., the ijth element of $\hat{\boldsymbol{\Sigma}}_x$ is

$$\hat{\sigma}_{ij} = \frac{1}{K} \sum_{k=1}^{K} (x_i^{(k)} - \mu_i)(x_j^{(k)} - \mu_j) = \frac{1}{K} \sum_{k=1}^{K} x_i^{(k)} x_j^{(k)} - \hat{\mu}_i \hat{\mu}_j, \qquad (i, j = 1, \cdots, n) \tag{12.45}$$

  However, note that in order for this estimation to be unbiased, i.e., $E(\hat{\boldsymbol{\Sigma}}_x) = \boldsymbol{\Sigma}_x$, the coefficient $1/K$ needs to be replace by $1/(K-1)$.

12.3 Stochastic Model of Signals

A physical signal can be modeled as a time function $x(t)$ which takes a real or complex value $x(t_0)$ 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.,

$$\mu_x(t) = \mu_x(t-\tau), \qquad R_x(t,\tau) = R_x(t-\tau), \qquad \Sigma_x(t,\tau) = \Sigma_x(t-\tau) \tag{12.46}$$

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 considered stationary processes.

- The *mean function* of $x(t)$ is the expectation defined as:

$$\mu_x(t) = E[x(t)] \tag{12.47}$$

  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

$$\sigma_x^2(t,t') = Cov[x(t), x(t')] = E[(x(t) - \mu_x(t))(x(t') - \mu_x(t'))]$$
$$= E[x(t)x(t')] - \mu_x(t)\mu_x(t') \tag{12.48}$$

  If the stochastic process is stationary, then $\sigma_x^2(t) = \sigma_x^2(t') = \sigma_x^2$, $\mu_x(t) = \mu_x(t') = \mu_x$, and $\sigma_x^2(t,t') = \sigma_x^2(t-t')$, the above can be expressed as

$$\sigma_x^2(t-t') = E[(x(t) - \mu_x(t))(x(t') - \mu_x(t'))] = E[x(t)x(t')] - \mu_x^2 \tag{12.49}$$

- The *autocorrelation function* of $x(t)$ is defined as

$$r_x(t,t') = \frac{\sigma_x^2(t,t')}{\sigma_x(t)\sigma_x(t')} \tag{12.50}$$

  If the stochastic process is stationary, then $\sigma_x^2(t) = \sigma_x^2(t') = \sigma_x^2$, and $\sigma_x^2(t,t') = \sigma_x^2(t-t')$, the above can be expressed as

$$r_x(t-t') = \frac{\sigma_x^2(t-t')}{\sigma_x^2} \tag{12.51}$$

- When two stochastic processes $x(t)$ and $y(t)$ are of interest, then their *cross-covariance* and *cross-correlation functions* are defined respectively as:

$$\sigma_{xy}^2(t,\tau) = Cov[x(t), y(\tau)] = E[(x(t) - \mu_x(t))(y(\tau) - \mu_y(\tau))]$$
$$= E[x(t)y(\tau)] - \mu_x(t)\mu_y(\tau) \tag{12.52}$$

and

$$r_{xy}(t,\tau) = \frac{\sigma_{xy}^2(t,\tau)}{\sigma_x(t)\sigma_y(\tau)} \tag{12.53}$$

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:

$$x'(t) = x(t) - \mu_x(t) \tag{12.54}$$

so that $\mu_{x'} = 0$. Without loss of generality, any stochastic process can be assumed to be centered. In this case, its covariance becomes

$$\sigma_x^2 = E[x^2(t)] \tag{12.55}$$

A *Markov process* $x(t)$ is a particular type of stochastic process whose future values depend only on its present value, independent of any past values. In other words, the probability of a certain future value conditioned on present and all past values is equal to the probability conditioned only on the present value:

$$Pr[x(t+h) = y | x(s) = \xi(s),\ \forall s \le t] = Pr[x(t+h) = y | x(t) = \xi(t)], \qquad \forall h > 0 \tag{12.56}$$

The discrete version of a Markov process is called a *Markov chain* where the random variable $x[n]$ is only defined over a set of discrete time moments, and it can only take a set of finite or countable values. A Markov chain has the similar property:

$$Pr(x[n] = y | x[m] = \xi[m],\ \forall m < n) = Pr(x[n] = y | x[n-1] = \xi[n-1]) \tag{12.57}$$

Sometimes this definition of Markov chain can be modified to become:

$$Pr(x[n] = y | x[m] = \xi[m],\ \ \forall m < n) = Pr(x[n+1] = y | x[n-m] = \xi[n-m],\ m = 1, \cdots, k) \tag{12.58}$$

This is called a Markov chain of order k, i.e., the future value depends on the k prior values.

In particular, when $k = 0$, we get a memoryless 0 order Markov chain of which random variables at all points are totally independent . Assuming the Markov chain is stationary with N

, and its covariance matrix is diagonal:

$$\boldsymbol{\Sigma}_x = \begin{bmatrix} \sigma^2 & 0 & 0 & \cdots & 0 \\ 0 & \sigma^2 & 0 & \cdots & 0 \\ 0 & 0 & \sigma^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & 0 & \cdots & \sigma^2 \end{bmatrix} = \sigma^2 \boldsymbol{I} \tag{12.59}$$

The covariance matrix of a stationary first order Markov process ($k = 1$) is

$$\mathbf{\Sigma}_x = \sigma^2 \begin{bmatrix} 1 & \rho & \rho^2 & \cdots & \rho^{N-1} \\ \rho & 1 & \rho & \cdots & \rho^{N-2} \\ \rho^2 & \rho & 1 & \cdots & \rho^{N-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \rho^{N-1} & \rho^{N-2} & \rho^{N-3} & \cdots & 1 \end{bmatrix} \tag{12.60}$$

where $\rho = \sigma^2_{n,n-1}/\sigma^2$ is the correlation (normalized covariance) between two consecutive variables, i.e., $0 \leq \rho \leq 1$. We see that the correlation between two variables $x[n]$ and $x[\nu]$ is $\rho^{|n-\nu|}$, which decays exponentially as a function of the distance $|n - \nu|$ between the two variables. Note that here $\mathbf{\Sigma}_x$ is a Toeplitz matrix.