The Discrete Fourier Transform

Theory, Algorithms and Applications

D. Sundararajan

World Scientific

The Discrete Fourier Transform

Theory, Algorithms and Applications

D. Sundararajan

World Scientific
Singapore • New Jersey • London • Hong Kong

Published by

World Scientific Publishing Co. Pte. Ltd.

P O Box 128, Farrer Road, Singapore 912805

USA office: Suite 1B, 1060 Main Street, River Edge, NJ 07661

UK office: 57 Shelton Street, Covent Garden, London WC2H 9HE

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library.

THE DISCRETE FOURIER TRANSFORM
Theory, Algorithms and Applications

ISBN 981-02-4521-1

Printed in Singapore by Uto-Print

The Discrete Fourier Transform

To my mother Dhanabagyam and my late father Duraisamy

Preface

Fourier transform is one of the most widely used transforms for the analysis and design of signals and systems in several fields of science and engineering. The primary objective of writing this book is to present the discrete Fourier transform theory, practically efficient algorithms, and basic applications using a down-to-earth approach. The computation of discrete cosine transform and discrete Walsh-Hadamard transforms are also described.

The book is addressed to senior undergraduate and graduate students in engineering, computer science, mathematics, physics, and other areas who study the discrete transforms in their course work or research. This book can be used as a textbook for courses on Fourier analysis and as a supplementary textbook for courses such as digital signal processing, digital image processing, digital communications engineering, and vibration analysis. The second group to whom this book is addressed is the professionals in industry and research laboratories involved in the design of general- and special-purpose signal processors, and in the hardware and software applications of the discrete transforms in various areas of engineering and science. For these professionals, this book will be useful for self study and as a reference book.

As the discrete transforms are used in several fields by users with different mathematical backgrounds, I have put considerable effort to make things simpler by providing physical explanations in terms of real signals, and through examples, figures, signal-flow graphs, and flow charts so that the reader can understand the theory and algorithms fully with minimum effort. Along with other forms of description, the reader can easily understand that the mathematical version presents the same information in a more abstract and compact form. In addition, I have deliberately made an attempt to present the material quite explicitly and describing only practically more useful methods and algorithms in very simple terms.

With the arrival of more and more new computers, the user needs a deep understanding of the algorithms and the architecture of the computer used to achieve an efficient implementation of algorithms for a given application. By going through the mathematical derivations, signal-flow graphs, flow charts, and the numerical examples presented in this book, the reader can get the necessary understanding of the algorithms. Large number of exercises are given, analytical and programming, that will further consolidate the readers' confidence. Answers to selected analytical exercises marked $*$ are given at the end of the book. Answers are given to all the programming exercises on the Internet at www.wspc.com/others/software/4610/. Important terms and expressions are defined in the glossary. A list of abbreviations is also given. For readers with little or no prior knowledge of discrete Fourier analysis, it is recommended that they read the chapters in the given order.

I assume the responsibility for all the errors in this book and would very much appreciate receiving readers' suggestions and pointing of any errors (email address: d_sundararajan@yahoo.com). I thank my friend Dr. A. Pedar for his help and encouragement during the preparation of this book. I thank my family for their support during this endeavor.

D. Sundararajan

Abbreviations

dc Constant
DCT Discrete cosine transform
DFT Discrete Fourier transform
DIF Decimation-in-frequency
DIT Decimation-in-time
DWT Discrete Walsh transform
FT Fourier transform
FS Fourier Series
IDFT Inverse discrete Fourier transform
Im Imaginary part of a complex number
lsb Least significant bit
LTI Linear time-invariant
msb Most significant bit
NDHT Naturally ordered discrete Hadamard transform
PM Plus-minus
RDFT Discrete Fourier transform of real data
Re Real part of a complex number
RIDFT Inverse discrete Fourier transform of the transform of real data
SDHT Sequency ordered discrete Hadamard transform
SFG Signal-flow graph
1-D One-Dimensional
2-D Two-Dimensional

The Discrete Fourier Transform

Contents

Chapter 1
Introduction

Fourier analysis is the representation of signals in terms of sinusoidal waveforms. This representation provides efficiency in the manipulation of signals in a large number of practical applications in science and engineering. Although the Fourier transform has been a valuable mathematical tool in the linear time-invariant (LTI) system analysis for a long time, it is the advent of digital computers and fast numerical algorithms that has made the Fourier transform the single most important practical tool in many areas of science and engineering. Fourier representation of signals is extremely useful in spectral analysis as well as a frequency-domain tool.

In this book, we will be dealing mostly with the discrete Fourier transform (DFT), which is the discrete version of the Fourier transform. The main purpose of this book is to present: (i) the DFT theory and some basic applications using a down-to-earth approach and (ii) practically efficient DFT algorithms and their software implementations. In the rest of this chapter, we explain the transform concept and describe the organization of this book.

1.1 The Transform Method

Transform methods are used to reduce the complexity of an operation by changing the domain of the operands. The transform method gives the solution of a problem in an indirect way more efficiently than direct methods. For example, multiplication operation is more complex than addition operation. In using logarithms, we find the logarithm of the two operands to be multiplied, add them, and find the antilogarithm to get the product.

By computing the common logarithm of a number, for example, we find the exponent to which 10 must be raised to produce that number. When numbers are represented in this form, by a law of exponents, the multiplication of numbers reduces to the addition of their exponents. In addition to providing faster implementation of operations, the transformed values give us a better understanding of the characteristics of a signal. The reader might have used the log-magnitude plot for better representation of certain functions.

The output of an LTI system can be found by using the convolution operation, which is more complex than the multiplication operation. When a given signal is represented in terms of complex exponentials (a functionally equivalent mathematical representation of sinusoidal waveforms), the response of a system is found by multiplying the complex coefficients of the complex exponentials representing the input signal by the corresponding complex coefficients representing the system impulse response. This is because a complex exponential input signal is a scaled version of itself at the output of an LTI system. Note that this procedure is very similar to the use of logarithms just described: in using common logarithms, we represent numbers as powers of ten to get advantages in number manipulation whereas, in using the Fourier transform, we represent signals in terms of complex exponentials to get advantages in signal manipulation and understanding. We use transform methods quite often in system analysis. Apart from logarithms, we usually prefer to use the Laplace transform to solve a differential equation rather than using a direct approach. Similarly, we prefer to use the z-transform to solve a difference equation.

The time- and frequency-domain approaches are two different ways of presenting the interaction between signals and systems. An arbitrary signal can be considered as a linear combination of frequency components. The time-domain representation is the superposition sum of the frequency components. The DFT is the tool that separates the frequency components. Viewing the signal in terms of its frequency components gives us a better understanding of its characteristics. In addition, it is easier to manipulate the signal. After manipulation, the inverse DFT (IDFT) operation can be used to sum all the frequency components to get the processed time-domain signal. Obviously, this procedure of manipulating signals is efficient only if the effort required in all the steps is less than that of the direct signal manipulation. The manipulation of signals, using DFT, is efficient because of the availability of fast algorithms.

1.2 The Organization of this Book

In Fourier analysis, the principal object is the sinusoidal waveform. Therefore, it is imperative to have a good understanding of its representation and properties. In Chapter 2, **The Discrete Sinusoid**, we describe the discrete sinusoidal waveform and, its representation and properties. The two principal operations, in Fourier analysis, are the decomposition of an arbitrary waveform into its constituent sinusoids and the building of an arbitrary waveform by summing a set of sinusoids. The first operation is called signal analysis and the second operation is called signal synthesis. The discrete mathematical formulation of these two operations are, respectively, called DFT and IDFT operations. In Chapter 3, **The Discrete Fourier Transform**, we derive the DFT and the IDFT expressions and provide examples of finding the DFT of some simple signals analytically. The advantages of sinusoidal representation of signals are also listed. The existence of fast algorithms and the usefulness of the DFT in applications is due to its advantageous properties. In Chapter 4, **Properties of the DFT**, we present the various properties and theorems of the DFT.

In Chapter 5, **Fundamentals of the PM DFT Algorithms**, we present the fundamentals of the practically efficient PM family of DFT algorithms. The classification of the PM DFT algorithms is also presented. In Chapter 6, **The $u \times 1$ PM DFT Algorithms**, the subset of $u \times 1$ PM DFT algorithms for complex data are derived and the software implementation of an algorithm is presented. In Chapter 7, **The 2×2 PM DFT Algorithms**, the 2×2 PM DFT algorithms for complex data are derived. When the data is real, usually it is, there are more efficient ways of computing the DFT and IDFT rather than using the algorithms for complex data directly. In Chapter 8, **DFT Algorithms for Real Data – I**, the efficient use of DFT algorithms for complex data for the computation of the DFT of real data (RDFT) and for the computation of the IDFT of the transform of real data (RIDFT) is described. In Chapter 9, **DFT Algorithms for Real Data – II**, the PM DFT and IDFT algorithms, specifically suited for real data, are deduced from the corresponding algorithms for complex data.

In the analysis of a 1-D signal, the signal, which is an arbitrary curve, is decomposed into a set of sinusoidal waveforms. In the analysis of a 2-D signal, typically an image, the signal, which is an arbitrary surface, is decomposed into a set of sinusoidal surfaces. In Chapter 10, **Two-Dimensional**

Discrete Fourier Transform, the theory and properties of the 2-D DFT is presented. The practically efficient way of computing the 2-D DFT is to compute the row DFTs followed by the computation of the column DFTs and vice versa. Using this approach, the 2-D PM DFT algorithms are derived.

In practice, most of the naturally occurring signals are continuous-time signals. It is by representing this signal by a set of finite samples, we are able to use the DFT. This creation of a set of samples to represent a continuous-time signal necessitates sampling and truncation operations. These operations introduce some errors in the signal representation but, fortunately, these errors can be reduced to a desired level by using an appropriate number of samples of the signal taken over proper record length. Therefore, the level of truncation and the number of samples used are a trade-off between accuracy and computational effort. A good understanding of the effects of truncation and sampling is essential in order to analyze a signal with minimum computational effort while meeting the required accuracy level. In Chapter 11, **Aliasing and Other Effects**, the problems created by sampling and truncation operations, namely aliasing, leakage, and picket-fence effects, are discussed.

The continuous-time Fourier series (FS) is the frequency-domain representation of a periodic continuous-time signal by an infinite set of harmonically related sinusoids. In Chapter 12, **The Continuous-Time Fourier Series**, the approximation of the continuous-time Fourier Series, 1-D and 2-D, by the DFT coefficients is described. The inability of the Fourier representation to provide uniform convergence in the vicinity of a discontinuity of a signal is also discussed. The continuous-time Fourier transform (FT) is the frequency-domain representation of an aperiodic continuous-time signal by an infinite set of sinusoids with continuum of frequencies. In Chapter 13, **The Continuous-Time Fourier Transform**, the approximation of the samples of the continuous-time Fourier transform, 1-D and 2-D, by the DFT coefficients is described.

A major application of the DFT is the fast implementation of fundamentally important operations such as convolution and correlation. In Chapter 14, **Convolution and Correlation**, the fast implementation of the convolution and correlation operations, 1-D and 2-D, using the DFT is presented.

The even extension of a signal eliminates discontinuity at the edges, if present, thereby enabling the signal to be represented by a smaller set