DSP

DIGITAL SIGNAL AND IMAGE PROCESSING SERIES

Digital Signal Processing Using MATLAB

Edited by André Quinquis

ISTE

WILEY

Digital Signal Processing using MATLAB®

Edited by
André Quinquis

ISTE

WILEY

First published in France in 2007 by Hermes Science/Lavoisier entitled "Le traitement du signal sous Matlab®: pratique et applications", 2nd edition
First published in Great Britain and the United States in 2008 by ISTE Ltd and John Wiley & Sons, Inc.

ISTE Ltd
6 Fitzroy Square
London W1T 5DX
UK

www.iste.co.uk

John Wiley & Sons, Inc.
111 River Street
Hoboken, NJ 07030
USA

www.wiley.com

Library of Congress Cataloging-in-Publication Data

Quinquis, André.
[Traitement du signal sous MATLAB. English]
Digital signal processing using MATLAB / André Quinquis.
p. cm.
Includes bibliographical references and index.
ISBN 978-1-84821-011-0
1. Signal processing--Digital techniques. I. Title.
TK5102.9.Q853 2008
621.382'2--dc22

2007043209

British Library Cataloguing-in-Publication Data
A CIP record for this book is available from the British Library
ISBN: 978-1-84821-011-0

Printed and bound in Great Britain by Antony Rowe Ltd, Chippenham, Wiltshire.

Digital Signal Processing using MATLAB®

Preface

Why and How this Book was Written

Sometimes it is easier to say what a book is not than what it exactly represents. It may be also better to resume the authors' motivations than to explain the book content itself.

From this point of view, our book is certainly not a traditional course, although it recalls many theoretical signal processing concepts. Indeed, we emphasize a limited number of important ideas instead of making a detailed description of the involved concepts. Intuitive manners have been used to link these concepts to physical aspects. Hence, we hope that reading this book will be much more exciting than studying a traditional signal processing course.

This book is also not a physics course, although a major purpose of most proposed exercises is to link abstract signal processing concepts to real-life problems. These connections are illustrated in a simple and comprehensive manner through MATLAB® simulations.

The main topics of this book cover the usual program of an undergraduate signal processing course. It is especially written for language and computer science students, but also for a much larger scientific community who may wish to have a comprehensive signal processing overview. Students will certainly find here what they are looking for, while others will probably find new and interesting knowledge.

This book is also intended to illustrate our pedagogical approach, which is based on three major reasons:

1. Students need to know how the teaching provided can be useful for them; it is their customer attitude.

2. Students have good potential for doing independent work; their interest and curiosity should be continuously stimulated by:

– using a diversified pedagogical approach that combines the two sides of a complete presentation methodology: from components to the system and vice versa;

– encouraging them to take advantage of their creativity through interactive educational tools; they should be allowed to make changes and even contribute to their development.

3. Students have to improve and validate their knowledge through written work; writing is still the best way to focus someone's concentration.

The role of simulations is becoming more and more important in the framework of a scientific education because it is an effective way to understand many physical phenomena, some of them less known or mastered, and to take into account their complexity. Simulations may be thus very useful for:

– *understanding* working principles and deriving behavior laws;

– *learning* about processing methods and systems running using algorithms to reproduce them off-line;

– *evaluating* the performance and robustness of various algorithms and estimating the influence of different parameters.

Simulations in signal processing education enable students to learn faster and facilitate the comprehension of the involved physical principles. From a teaching point of view, simulation tools lead to lower costs and time efficiency.

This book is based on a signal processing course, which has been successfully given for many years in several universities. According to our experience, signal theory abstract concepts and signal processing practical potentialities can be linked only through tutorial classes and simulation projects. In this framework, simulations appear to be the necessary complement for the classical tripod theory – modeling – experimentation.

This book brings together into a clear and concise presentation the main signal processing methods and the related results, using MATLAB software as a simulation tool. Why MATLAB? Because it is:

– simple to learn and to use;

– powerful and flexible;

– accurate, robust and fast;

– widespread in both academic and industrial environments;

– continuously updated by professionals.

The word "signal" stands for a physical entity, most often of an electrical nature, like that observed at a microphone output. It is submitted to various transformations when it goes through a system. Thus, in a communication chain, the signal is subject to some changes (distortion, attenuation, etc.), which can make it unrecognizable. The aim is to understand this evolution in order to properly recover the initial message.

In other words, a signal is a physical support of information. It may carry the orders in a control and command equipment or multimedia (speech and image) over a network. It is generally very weak and it has to be handled with much caution in order to reach the signal processing final goal, i.e. information extraction and understanding.

Signal processing is widely used in many industrial applications such as: telecommunications, audio and speech signal processing, radar, sonar, non-destructive control, vibrations, biomedicine, imagery, etc. Standard signal processing functions include signal analysis, improvement, synthesis, compression, detection, classification, etc., which depend on and interact with each other in an integrated information processing chain.

The digital signal processing methods provide noteworthy capabilities: accurate system design, excellent equipment reproducibility, high stability of their exploitation characteristics and an outstanding supervision facility.

The digital signal processing boom is related to the development of fast algorithms to calculate the discrete Fourier transform. Indeed, this is the equivalent of the Fourier transform in the discrete domain and so it is a basic tool to study discrete systems. However, related concepts are generally considered highly theoretical and accessible to scientific researchers rather than to most engineers. This book aims to overcome this difficulty by putting the most useful results of this domain within the understanding of the engineer.

Chapter 1 briefly describes essential concepts of MATLAB software, which is an interactive software tailored for digital signal processing. Language rules, elementary operations as well as basic functions are presented. Chapter 2 illustrates the generation of 1D or 2D (image) digital signals as data vectors and matrices respectively.

Finding the solution of a signal processing problem involves several distinct phases. The first phase is the modeling: the designer chooses a representation model for an observed data. When it can be done very accurately the signals are said to be deterministic. A powerful tool for analyzing them is provided by the Fourier transform, also called frequency representation, which is presented in Chapter 5. Its equivalent in the discrete domain is represented by the z-transform, which is developed in Chapter 6.

There are many other processes, which give different and apparently unpredictable results, although they are observed using identical experimental conditions. They are known as random processes, such as the receiver's thermal noise. The wide sense stationary random processes, which form a particularly interesting class of these signals, are presented in Chapter 3. Some useful statistical tools for testing different hypothesis about their parameters behavior are provided in Chapter 4.

From a very general point of view, digital signal processing covers all the operations, arithmetical calculations and number handling performed on the signal to be processed, defined by a number series, in order to obtain a transformed number series representing the processed signal. Very different functions can be carried out in this way, such as classical spectral analysis (Chapter 10), time-frequency analysis (Chapters 11 and 12), linear filtering (Chapters 7 and 8), detection and estimation (Chapter 9), and feature extraction for information classification or compression (Chapters 13 and 14).

Theoretical developments have been reduced to the necessary elements for a good understanding and an appropriate application of provided results. A lot of MATLAB programs, solved examples and proposed exercises make it possible to directly approach many practical applications. The reader interested in some more complementary information will find this in the references cited at the end of this book.

Finally, I would like to acknowledge all the members of my team, Emanuel Radoi, Cornel Ioana, Ali Mansour and Hélène Thomas, for their contributions to this book.

André QUINQUIS

Table of Contents

Chapter 1

Introduction

1.1. Brief introduction to MATLAB

1.1.1. *MATLAB software presentation*

MATLAB (MATrix LABoratory) is an interactive software, developed by Math Works Inc. and intended especially for digital signal processing. It is particularly effective when the data format is vector or matrix.

MATLAB integrates digital calculus, data visualization and open environment programming. MATLAB exists under both Windows and UNIX. Many demonstrations are available using the command `demo`.

This digital simulation software enables a fast and simple visualization of the obtained results.

MATLAB was primarily written in FORTRAN and C. However, MATLAB knows to interpret commands, while a compilation of the source code is required by FORTRAN and C.

MATLAB is especially designed for digital signal processing and for complex digital system modeling and simulation. It is also suitable for processing data series, images or multidimensional data fields.

MATLAB software general structure is provided in Figure 1.1.

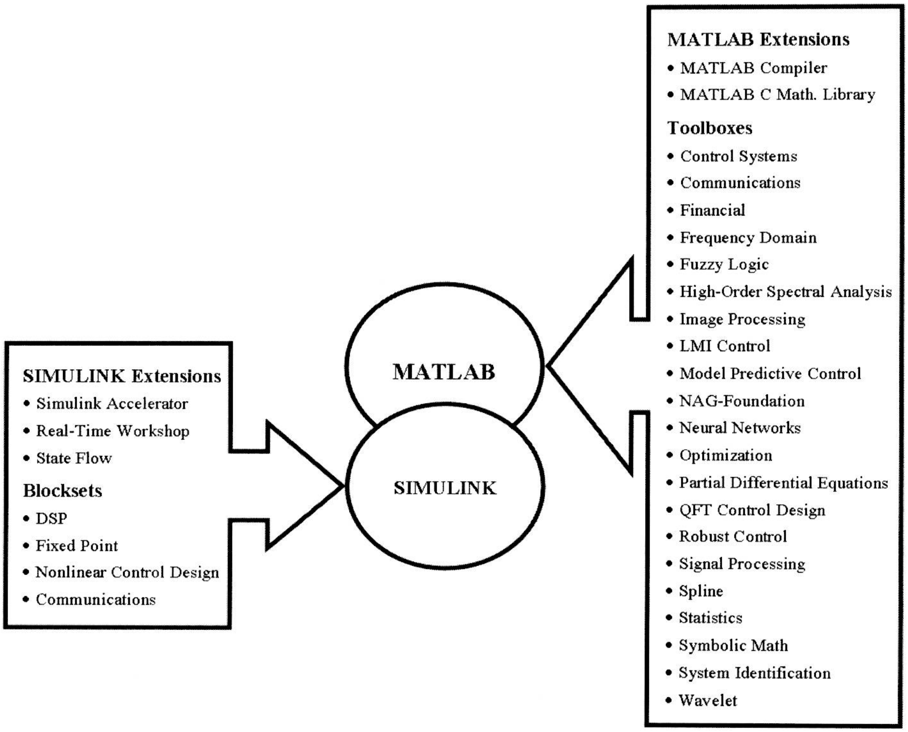

Figure 1.1. *MATLAB software general structure*

The toolboxes extend the basic MATLAB functions and perform specific tasks corresponding to different digital processing fields, such as image processing, optimization, statistics, system control and identification, neural networks, fuzzy systems, etc.

SIMULINK is an interactive software designed for modeling and simulating continuous-time or discrete-time dynamical systems or hybrid structures containing both analog and digital systems. It makes use of a mathematical equation set and provides a large variety of predefined or user-defined functional blocks.

MATLAB has been developed for several years, especially as a consequence of its use in the academic environment as an excellent education tool in mathematics, engineering and science. In addition, MATLAB has already proven its utility for scientific research and technological development.

In order to run MATLAB, type the command `matlab` with UNIX shell (if a MATLAB license under UNIX is available) or double click on the MATLAB icon if the operating system is Windows. To exit MATLAB, type `exit` or `quit`. If MATLAB is running under UNIX, you may have access to all UNIX commands using just before the symbol! (example: `!ls -l`).

1.1.2. *Important MATLAB commands and functions*

`who`	lists the variables in the current workspace
`whos`	the same as previous, but lists more information about each variable
`what`	lists MATLAB-specific files in directory
`size`	provides the size of a data array
`length`	provides the size of a data vector
`help`	displays help text in Command Window
`exit, quit`	exits from MATLAB

Table 1.1. *General commands*

`dir, chdir, delete, load, save, type`	similar to the corresponding DOS commands
`pack`	consolidates workspace memory

Table 1.2. *Commands related to the workspace*

`+, -, *, /, ^`	usual arithmetical operators
`.`	followed by an arithmetical operator for applying it to each array element
`'`	Hermitian operator
`.'`	transpose operator

Table 1.3. *Arithmetical operators*

<, <=, >, >=	usual relational operators
==	equality operator
~=	inequality operator
&	element-wise logical AND
|	element-wise logical OR
~	logical complement (NOT)

Table 1.4. *Relational and logical operators*

=	variable assignment operator
,	used to separate the arguments of a function or the elements of a data array
[]	used to build data arrays
()	used in arithmetical expressions
:	used for indexing variables
;	used at the end of a statement to cancel displaying any output
...	used to continue a command on the next line
%	used to enter a comment

Table 1.5. *Special characters*

ans	default name of a variable or a result
eps	spacing of floating point numbers
pi	value of $\pi = 3.14159...$
i,j	value of $\sqrt{-1}$
Inf	IEEE arithmetic representation for positive infinity (1/0)
NaN	IEEE arithmetic representation for Not-a-Number (0/0)
nargin	returns the number of function input arguments
nargout	returns the number of function output arguments

Table 1.6. *Special variables and constants*

abs	absolute value function
sqrt	square root function
real	real part of a complex variable
imag	imaginary part of a complex variable
angle	returns the phase angles, in radians, of a complex variable
conj	complex conjugate operator
sign	signum function
rem	returns the remainder after division
exp	exponential function
log	natural logarithm function
log10	base 10 logarithm function

Table 1.7. *Elementary mathematical functions*

sin, cos, tan, cot, sec	usual trigonometric functions
asin, acos, atan, acot, asec	inverse trigonometric functions
sinh, cosh, tanh, coth, sech	hyperbolic functions
asinh, acosh, atanh, acoth, asech	inverse hyperbolic functions

Table 1.8. *Trigonometric functions*

max	largest component
min	smallest component
mean	average or mean value
std	standard deviation
sum	sum of elements
cumsum	cumulative sum of elements
prod	product of elements
cumprod	cumulative product of elements

Table 1.9. *Data analysis functions*