



Applied and Numerical Harmonic Analysis

Series Editor

John J. Benedetto
University of Maryland

Editorial Advisory Board

Akram Aldroubi
NIH, Biomedical Engineering/
Instrumentation

Ingrid Daubechies
Princeton University

Christopher Heil
Georgia Institute of Technology

James McClellan
Georgia Institute of Technology

Michael Unser
NIH, Biomedical Engineering/
Instrumentation

Victor Wickerhauser
Washington University

Douglas Cochran
Arizona State University

Hans G. Feichtinger
University of Vienna

Murat Kunt
Swiss Federal Institute
of Technology, Lausanne

Wim Sweldens
Lucent Technologies
Bell Laboratories

Martin Vetterli
Swiss Federal Institute
of Technology, Lausanne

Anthony Teolis

Computational
Signal Processing
with
Wavelets

Birkhäuser
Boston • Basel • Berlin

Anthony Teolis
Naval Research Laboratory
4555 Overlook Avenue
Bldg. 210, Code 5720
Washington, DC 20375

AIMS, Inc.
6110 Executive Blvd.
Rockville, Maryland 20375

Library of Congress Cataloging-in-Publication Data

Teolis, Anthony, 1964—

Computational signal processing with wavelets / Anthony Teolis.

p. cm. — (Applied and numerical harmonic analysis)

Includes bibliographical references and index.

ISBN-13: 978-1-4612-8672-1

(Basel)

1. Signal processing—Mathematics. 2. Wavelets (Mathematics)

I. Series.

TK5102.9.T44 1998

621.382'2' 015152433—dc21

97-52001

CIP

Printed on acid-free paper
© 1998 Birkhäuser Boston

Birkhäuser ®

Softcover reprint of the hardcover 1st edition 1998

Copyright is not claimed for works of U.S. Government employees.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without prior permission of the copyright owner.

Permission to photocopy for internal or personal use of specific clients is granted by Birkhäuser Boston for libraries and other users registered with the Copyright Clearance Center (CCC), provided that the base fee of \$6.00 per copy, plus \$0.20 per page is paid directly to CCC, 222 Rosewood Drive, Danvers, MA 01923, U.S.A. Special requests should be addressed directly to Birkhäuser Boston, 675 Massachusetts Avenue, Cambridge, MA 02139, U.S.A.

ISBN-13: 978-1-4612-8672-1 e-ISBN-13: 978-1-4612-4142-3

DOI: 10.1007/978-1-4612-4142-3

SPIN 10844448

Typeset by T & T Techworks Inc., Coral Springs, FL

MATLAB® is a registered trademark of The MathWorks Incorporated

9 8 7 6 5 4 3 2

Dedication

To my two favorite playmates, Spencer and Trevor, who relinquished, without their consent or pardon, many hours, Saturdays, and weeknights spent with Dad.

Contents

Preface	xiii
List of Tables	xvii
List of Figures	xix
1 Introduction	1
1.1 Motivation and Objectives	1
1.2 Core Material and Development	1
1.3 Hybrid Media Components	2
1.4 Signal Processing Perspective	3
1.4.1 Analog Signals	3
1.4.2 Digital Processing of Analog Signals	3
1.4.3 Time–Frequency Limitedness	5
2 Mathematical Preliminaries	7
2.1 Basic Symbols and Notation	7
2.2 Basic Concepts	8
2.2.1 Norm	8
2.2.2 Inner Product	8
2.2.3 Convergence	9
2.2.4 Hilbert Spaces	9
2.3 Basic Spaces	10
2.3.1 Bounded Functions	10
2.3.2 Absolutely Integrable Functions	10
2.3.3 Finite Energy Functions	11
2.3.4 Finite Energy Periodic Functions	11
2.3.5 Time–Frequency Concentrated Functions	11
2.3.6 Finite Energy Sequences	12
2.3.7 Bandlimited Functions	12
2.3.8 Hardy Spaces	12
2.4 Operators	13
2.4.1 Bounded Linear Operators	13

2.4.2	Properties	14
2.4.3	Useful Unitary Operators	15
2.5	Bases and Completeness in Hilbert Space	16
2.6	Fourier Transforms	17
2.6.1	Continuous Time Fourier Transform	18
2.6.2	Continuous Time–Periodic Fourier Transform	18
2.6.3	Discrete Time Fourier Transform	20
2.6.4	Discrete Fourier Transform	20
2.6.5	Fourier Dual Spaces	20
2.7	Linear Filters	21
2.7.1	Continuous Filters and Fourier Transforms	21
2.7.2	Discrete Filters and Z-Transforms	22
2.8	Analog Signals and Discretization	22
2.8.1	Classical Sampling Theorem	23
2.8.2	What Can Be Computed Exactly?	24
	Problems	27
3	Signal Representation and Frames	29
3.1	Inner Product Representation (Atomic Decomposition)	29
3.2	Orthonormal Bases	30
3.2.1	Parseval and Plancherel	31
3.2.2	Reconstruction	31
3.2.3	Examples	32
3.3	Riesz Bases	33
3.3.1	Reconstruction	34
3.3.2	Examples	35
3.4	General Frames	36
3.4.1	Basic Frame Theory	37
3.4.2	Frame Representation	40
3.4.3	Frame Correlation and Pseudo-Inverse	42
3.4.4	Pseudo-Inverse	46
3.4.5	Best Frame Bounds	48
3.4.6	Duality	49
3.4.7	Iterative Reconstruction	50
	Problems	55
4	Continuous Wavelet and Gabor Transforms	59
4.1	What Is a Wavelet?	60
4.2	Example Wavelets	62
4.2.1	Haar Wavelet	62
4.2.2	Shannon Wavelet	62
4.2.3	Frequency B-spline Wavelets	63
4.2.4	Morlet Wavelet	65
4.2.5	Time–Frequency Tradeoffs	66
4.3	Continuous Wavelet Transform	67

4.3.1	Definition	67
4.3.2	Properties	69
4.4	Inverse Wavelet Transform	70
4.4.1	The Idea Behind the Inverse	70
4.4.2	Derivation for $L^2(\mathbb{R})$	73
4.4.3	Analytic Signals	74
4.4.4	Admissibility	77
4.5	Continuous Gabor Transform	78
4.5.1	Definition	78
4.5.2	Inverse Gabor Transform	79
4.6	Unified Representation and Groups	80
4.6.1	Groups	80
4.6.2	Weighted Spaces	81
4.6.3	Representation	82
4.6.4	Reproducing Kernel	83
4.6.5	Group Representation Transform	84
	Problems	88
5	Discrete Wavelet Transform	89
5.1	Discretization of the CWT	90
5.2	Multiresolution Analysis	92
5.2.1	Multiresolution Design	92
5.2.2	Resolution and Dilation Invariance	93
5.2.3	Definition	94
5.3	Multiresolution Representation	94
5.3.1	Projection	95
5.3.2	Fourier Transforms	95
5.3.3	Between Scale Relations	97
5.3.4	Haar MRA	99
5.4	Orthonormal Wavelet Bases	101
5.4.1	Characterizing W_0	102
5.4.2	Wavelet Construction	104
5.4.3	The Scaling Function	107
5.5	Compactly Supported (Daubechies) Wavelets	108
5.5.1	Main Idea	109
5.5.2	Trigonometric Half-Band Filters	109
5.5.3	Examples	112
5.6	Fast Wavelet Transform Algorithm	116
5.6.1	Filter Bank Decomposition of the Identity	116
5.6.2	Down- and Up-Sampling	117
5.6.3	Examples	120
	Problems	125
6	Overcomplete Wavelet Transform	127
6.1	Discretization of the CWT Revisited	129

6.1.1	Definition	129
6.1.2	Semilog Regular Time-Scale Sampling	130
6.1.3	Invertibility	131
6.1.4	Overcompleteness and Redundancy	132
6.2	Filter Bank Implementation	136
6.2.1	Analysis	137
6.2.2	Synthesis	138
6.3	Time-Frequency Localization and Wavelet Design	145
6.3.1	The Uncertainty Principle	145
6.3.2	Parametric Bandlimited Wavelets	148
6.4	OCWT Examples	150
6.5	Irregular Sampling and Frames	156
6.5.1	Paley-Wiener Frames	156
6.5.2	Wavelet Frames	160
6.5.3	Irregular Sampling in the Time-Scale Domain	163
	Problems	167
7	Wavelet Signal Processing	171
7.1	Noise Suppression	171
7.1.1	Noise Domains	172
7.1.2	Problem	172
7.1.3	Approach	173
7.1.4	Coherence	178
7.1.5	Frame Localization and Coherence	184
7.1.6	Reconstruction Error from Thresholded OCWTs . .	190
7.1.7	Numerical Experiment	192
7.1.8	Coefficient Noise and Overcompleteness	206
7.1.9	Measuring Noise (SNR)	210
7.2	Compression	212
7.2.1	Problem	212
7.2.2	Performance Measures	212
7.2.3	Approach	213
7.2.4	Numerical Experiment	216
7.2.5	Remarks	223
7.3	Digital Communication	224
7.3.1	Problem	224
7.3.2	Objectives	225
7.3.3	Approach	226
7.3.4	Performance Measures	229
7.3.5	Computer Implementation	232
7.3.6	Wavelet Communication Testbed	236
7.3.7	Numerical Experiments	241
7.3.8	Remarks	253
7.4	Identification	254
7.4.1	Approach	254

7.4.2	Performance Indicators	256
7.4.3	Numerical Experiment	256
7.4.4	Remarks	260
7.5	Conclusion	260
	Problems	261
8	Object-Oriented Wavelet Analysis with MATLAB 5	263
8.1	Wavelet Signal Processing Workstation	264
8.2	MATLAB Coding	267
8.2.1	Object-Oriented Programming	267
8.2.2	Designed Classes	268
8.3	The <code>sampled_signal</code> Object	268
8.3.1	Class Construction	268
8.3.2	<code>sampled_signal</code> Methods	271
8.4	Wavelet Transform Implementation	295
8.4.1	Overcomplete Wavelet Transform (OCWT)	295
8.4.2	GUI for Filter Bank Specification	296
8.5	The <code>wavelet</code> Object	298
8.5.1	Class Construction	298
8.5.2	<code>wavelet</code> Methods	300
8.6	Processing Example	304
8.6.1	Forward OCWT	304
8.6.2	Inverse OCWT	307
8.7	Supporting Functions and Globals	308
8.7.1	Frequency and Time Units	308
8.7.2	Graphical	309
8.7.3	Wavelet	311
References		313
Index		321

Preface

Overview

For over a decade now, wavelets have been and continue to be an evolving subject of intense interest. Their allure in signal processing is due to many factors, not the least of which is that they offer an intuitively satisfying view of signals as being composed of *little* pieces of *waves*. Making this concept mathematically precise has resulted in a deep and sophisticated wavelet theory that has seemingly limitless applications.

This book and its supplementary hands-on electronic component are meant to appeal to both students and professionals. Mathematics and engineering students at the undergraduate and graduate levels will benefit greatly from the introductory treatment of the subject. Professionals and advanced students will find the overcomplete approach to signal representation and processing of great value. In all cases the electronic component of the proposed work greatly enhances its appeal by providing interactive numerical illustrations.

A main goal is to provide a bridge between the theory and practice of wavelet-based signal processing. Intended to give the reader a balanced look at the subject, this book emphasizes both theoretical and practical issues of wavelet processing. A great deal of exposition is given in the beginning chapters and is meant to give the reader a firm understanding of the basics of the discrete and continuous wavelet transforms and their relationship. Later chapters promote the idea that overcomplete systems of wavelets are a rich and largely unexplored area that have demonstrable benefits to offer in many applications.

In addition to the text, there is also supporting MATLAB based software that is graphically oriented and provides a computational platform for exploration and illustration of many of the ideas and algorithms presented here. The software includes comprehensive graphical interfaces for high level interaction as well as hundreds of low-level object-oriented methods for general signal processing.

Organization and Features

The book while written for senior or beginning graduate students in mathematics or engineering, is also accessible to professionals and practitioners in the signal processing community. Technical prerequisites include an undergraduate level knowledge of linear algebra, linear systems theory, Fourier transform theory, and a working knowledge of MATLAB basic functionality. Additional familiarity with operator theory and real analysis is helpful but not required.

Beginning chapters are expository in nature and describe basic notation, concepts, orthonormal wavelets, and frames. Later chapters depart slightly from the mainstream of wavelet theory and instead emphasize *overcomplete* representations of signals as opposed to the more widely used *orthonormal* representations associated with the discrete wavelet transform. Finally, the presentation becomes more numerically oriented in the last chapters where the benefits of overcomplete wavelet representations are explored in various applications. These numerical explorations are fully reproducible and extensible using the available software. The impatient and/or curious reader is encouraged to start there.

This work is geared towards practical application and numerical implementation of wavelet-based algorithms supported by a solid mathematical foundation. Some of its main features are listed as follows.

- An expository treatment of the following topics are included:
 - continuous and discrete Fourier transforms,
 - orthonormal and biorthogonal bases,
 - frames, wavelet frames, and reconstruction,
 - discrete wavelet transform and orthonormal wavelets,
 - classical sampling theorem, and
 - regular and irregular sampling and reconstruction.
- A frame-based theory of the discretization and reconstruction of analog signals is developed in terms of the sampling of a continuous transform.
- The continuous wavelet and Gabor transforms are introduced in a unified group-theoretic setting.
- Concepts and techniques are numerically demonstrated through
 - software reproducible examples,
 - interactive graphical user interfaces, and
 - over 120 traditional static figures.
- Problem exercises are given at the end of each major chapter to reinforce concepts and ideas.
- A new and efficient *overcomplete* wavelet transform is introduced and applied to the tasks of

- noise suppression,
- compression,
- digital communication, and
- identification.

Chapter 1 describes the motivations and objectives of the entire book and provides an overall perspective to the material. Chapter 2 introduces the notation and basic mathematical concepts used throughout the text. Chapter 3 discusses mathematical frames and their use as signal representations as well as algorithms for reconstruction of signals from their frame representations. Chapter 4 presents the continuous wavelet and Gabor transforms and also provides a unified view of them in terms of frame representations. Chapter 5 reviews the discrete wavelet transform, multiresolution analysis, and the construction of compactly supported orthonormal wavelet bases. The fast wavelet transform is also described there. Chapter 6 introduces the overcomplete wavelet transform, its inverse, and their filter bank implementations. Chapter 7 presents several applications of wavelet-based signal processing including noise suppression, signal compression, and identification. Finally, Chapter 8 describes the supporting object-oriented MATLAB code which has been used to numerically illustrate the material.

Computational Aspects

Numerical examples presented in this book have all been computed using a suite of object-oriented tools developed in MATLAB 5 called the Wavelet Signal Processing Workstation (WSPW). The material includes a demonstration copy of the WSPW and is available through the Internet. This object-oriented wavelet signal processing software (MATLAB) is available at the Web site

www.birkhauser.com/book/ISBN/0-8176-3909-8

The software requires MATLAB version 5.0 or later to run and has been tested on operating systems including Windows 95 and the various flavors of UNIX and LINUX.

Acknowledgments

Many people have aided the production of this project and I am greatly indebted to all. There are several individuals and organizations whose support demands special mention and they are listed in the following.

First, and foremost, is my wife, Carole. In addition to being my wife and best friend, she has served as my main supporter, in both the technical and emotional senses. She has tolerated my frequent absences from the affairs of

family and friends all for the benefit of this project. Balancing the demands of our two children, her own career, and myself during this time has been no small sacrifice. To her I cannot be too grateful.

I am grateful to the Naval Research Laboratory (NRL) in Washington, DC, where I have worked as a contractor for the past four years in the Tactical Electronic Warfare Division. Many of the aspects of my work there are reflected in this book. In particular, I am indebted to Dr. Charles Heider, Mr. Roger Oxley, and Dr. Joseph Lawrence for providing me with the intellectual freedom, resources, and general support needed to pursue and complete this project. Specifically, NRL branches 5740 and 5720 have provided a stimulating and exciting working environment that has contributed positively in many ways to the production of this work.

I thank my company, AIMS, Inc., and its founders John and Mary Baras, as well as all my fellow employees for their continued support during this project. Deserving of special note is Rob Mentle who, in addition to being my coworker and friend, has served as proofreader, reviewer, system administrator, and stock advisor.

For introducing me to frames and irregular sampling from a mathematically sophisticated viewpoint, John Benedetto deserves special thanks. Much of the material presented here on noise suppression and compression has grown from joint work on those subjects.

This work has also benefited greatly from the comments of numerous anonymous and nonanonymous reviewers. To them all I am very much indebted. Of special note is Hans Feichtinger of the University of Vienna. His deep understanding, unique perspective, and pervasive and penetrating commentary on early versions of the manuscript have resulted in this much improved version of the current work.

Any errors which may remain are the sole responsibility of the author. Any comments, suggestions, bug reports, errors, or thoughts of any kind, would be most welcome by the author. Please send your email to

`tonyt@palindrome.nrl.navy.mil`

Special thanks is owed to all the people at Birkhäuser who helped make this project become a reality. These include Wayne Yuhasz, who first suggested this project to me and has provided invaluable guidance throughout its preparation, and Lauren Lavery who has handled many of the details of its production.

In addition, the contributing support of the following contracts is gratefully acknowledged: (NRL) N00014-C-93-2093, (NRL/DARPA) N00014-C-95-2043, (ONR/DARPA) N00014-C-97-0364, and (CECOM) DAAB07-97-C-A753.

List of Tables

2.1	Unitary operators $U : L^2(\mathbb{R}) \mapsto L^2(\mathbb{R})$, their inverses (equal to their adjoints), and their Fourier transforms. Here $f \in L^2(\mathbb{R})$, $a, b \in \mathbb{R}$, and $s \in \mathbb{R}^+$	16
2.2	Fourier dual space relations.	21
3.1	Relation of frame objects and their duals.	50
5.1	Scaling function and related filter properties for orthonormal wavelet bases.	108
6.1	Wavelet transform relations.	166
7.1	Computed CSNRs for different signals. Here the DWT representation is computed using the Daubechies D3 wavelet and the OCWT representation is computed with the Morlet(40,20) filter bank, ($a_0 = 1.09$, #filters = 32).	185
7.2	Parameters for Numerical Experiment 7.10.	194
7.3	Coherence, error bound, measured error, and absolute and relative thresholds for the signal $e_{10}\text{chirp}@1$	198
7.4	Coherence, error bound, measured error, and absolute and relative thresholds for the signal $e_5\text{chirp}@4$	199
7.5	Coherence, error bound, measured error, and absolute and relative thresholds for the signal $e_{10}\text{packet}$	199
7.6	Expected power due to noise contributions.	211
7.7	Parameters for Numerical Experiment 7.14.	217
7.8	Filter banks used in Numerical Experiment 7.14.	217
7.9	Compression results for Numerical Experiment 7.14.	222
7.10	Parameters for Numerical Experiment 7.15.	247
7.11	Decoding stability with respect to uncertainty of synthesis filter bank and <i>full</i> knowledge of cell template.	253
7.12	Parameters for Numerical Experiment 7.16.	257
7.13	Morlet OCWT match distances for SNR = 0dB.	259
7.14	Time domain match distances for SNR = 0dB.	259

7.15	Morlet OCWT match distances for SNR = -8dB.	259
7.16	Matched filter match distances for SNR = -8dB.	259
8.1	Summary of some <code>sampled_signal</code> methods.	272
8.2	Summary of the main <code>wavelet</code> methods.	300

List of Figures

2.1	Continuous linear system output as convolution.	21
2.2	Discrete linear system output as convolution.	22
3.1	The mappings L and L^*	41
4.1	Members of a family of wavelets shown in both the time (left) and frequency (right) domains. All members are generated by the dilation and translation in time of a single function called the analyzing wavelet.	61
4.2	The Haar wavelet in the time and frequency domains. . . .	63
4.3	A second-order real B-spline wavelet in the time and frequency domains.	65
4.4	Morlet analyzing wavelet in the time and frequency domains (center frequency at 1MHz and unit variance).	67
4.5	The CWT as the output of a continuous bank of linear filters. .	68
4.6	The “continuous” wavelet transform magnitude of a chirp signal.	69
4.7	Filter and its “inverse” implementation.	71
4.8	Continuous filter bank and its “inverse” implementation. . .	72
5.1	The dyadic lattice in the time-scale plane.	91
5.2	Two different resolutions in the Haar MRA.	100
5.3	A graphical depiction of the dilation equation for the Haar MRA.	101
5.4	(a) Daubechies 4 tap filter ($N = 2$) low and high pass filter coefficients and their corresponding scaling and wavelet functions; (b) magnitude squared (power) spectra of Daubechies scaling (solid) and wavelet (dashed) functions for $N = 2$. .	113
5.5	(a) Daubechies 8 tap filter ($N = 4$) low and high pass filter coefficients and their corresponding scaling and wavelet functions; (b) magnitude squared (power) spectra of Daubechies scaling (solid) and wavelet (dashed) functions for $N = 4$. .	114

5.6	(a) Daubechies 14 tap filter ($N = 7$) low and high pass filter coefficients and their corresponding scaling and wavelet functions; (b) magnitude squared (power) spectra of Daubechies scaling (solid) and wavelet (dashed) functions for $N = 7$	115
5.7	Two-band decomposition and synthesis bank suggested by Equation (5.17).	117
5.8	A single level of the fast wavelet decomposition cascaded with its corresponding fast wavelet reconstruction filters.	118
5.9	A three-level wavelet decomposition (analysis) with signal lengths indicated.	119
5.10	Frequency modulation and corresponding time domain signal for the signal “FMramp.”	121
5.11	Synthetic signal “packet” in the time and frequency domains. The top plot shows the real part and magnitude of the time signal and the bottom plot shows the Fourier transform magnitude.	121
5.12	Daubechies 4 tap ($N = 2$) discrete wavelet transform magnitude of the signal “packet.”	122
5.13	Daubechies 8 tap ($N = 4$) discrete wavelet transform magnitude of the signal “packet.”	122
5.14	Daubechies 14 tap ($N = 7$) discrete wavelet transform magnitude of the signal “packet.”	123
5.15	Synthetic signal “FMramp” in the time and frequency domains. The top plot shows the real part and magnitude of the time signal and the bottom plot shows the Fourier transform magnitude.	123
5.16	Daubechies 4 tap ($N = 2$) discrete wavelet transform magnitude of the signal “FMramp.”	124
5.17	Daubechies 8 tap ($N = 4$) discrete wavelet transform magnitude of the signal “FMramp.”	124
5.18	Daubechies 14 tap ($N = 7$) discrete wavelet transform magnitude of the signal “FMramp.”	125
6.1	A signal with varying time-frequency content and two sampling strategies: (a) uniform sampling in accordance with the Nyquist criterion (allows reconstruction), and (b) irregular sampling with local extrema (may not allow reconstruction).	133
6.2	An overcomplete wavelet transform as an analysis filter bank.	138
6.3	Inverse overcomplete wavelet transform as a synthesis filter bank.	140
6.4	A time-scale semilog regular sampling set of the form of Equation (6.5) with a time sample period of 0.25 and a dilation factor of 1.1.	143

6.5	Parametric bandlimited wavelets in time and frequency for fixed center frequency of 20Hz and bandwidth of 20Hz and values of $N = 0, 3, 6$, and 9.	149
6.6	Frequency modulations of the six-signal synthetic test set.	150
6.7	Frequency domain transfer functions of a Morlet filter bank ($M = 64$ filters) and the associated frequency support function of Equation (6.3).	151
6.8	Frequency domain transfer functions of a PBL filter bank ($M = 64$ filters) and the associated frequency support function of Equation (6.3).	152
6.9	FM signal 1: (Left) Morlet OCWT; (right) PBL OCWT (signal upshifted in frequency by 10).	153
6.10	FM signal 2: (left) Morlet OCWT; (right) PBL OCWT (signal upshifted in frequency by 10).	153
6.11	FM signal 3: (left) Morlet OCWT; (right) PBL OCWT (signal upshifted in frequency by 10).	153
6.12	FM signal 4: (left) Morlet OCWT; (right) PBL OCWT (signal upshifted in frequency by 10).	154
6.13	FM signal 5: (left) Morlet OCWT; (right) PBL OCWT (signal upshifted in frequency by 10).	154
6.14	FM signal 6: (left) Morlet OCWT; (right) PBL OCWT (signal upshifted in frequency by 10).	154
6.15	Morlet OCWT of FM signal “WAVE.”	155
6.16	B-spline OCWT of FM signal “happy.”	156
6.17	A uniformly dense sequence generated as a tessellation of a uniform sequence on a compact interval.	159
6.18	A Kadec–Levinson sequence.	160
6.19	The wavelet transform W_g on different domains.	165
7.1	Noise model and incoherence suppression processing.	177
7.2	Absolute and relative coherence (CSNR) measures of normalized chirp and Gaussian noise signals with respect to four different representation domains.	183
7.3	Filter bank implementation schematic for a fixed threshold reconstruction from a tight frame representation.	191
7.4	Equivalent filter bank implementation of a fixed threshold reconstruction which acts as a variable threshold on a weighted tight frame.	191
7.5	Morlet Filter bank with 32 channels used in Numerical Experiment 7.10.	193
7.6	Coherence bound and measured values as function of threshold for the signal $e_{10}\text{chirp}@1$	195
7.7	Reconstructions associated with relative threshold values of 0.1, 0.2, and 0.3 for the signal $e_{10}\text{chirp}@1$	196

7.8	Coherence bound and measured values as function of threshold for the signal $e_{10}\text{chirp}@4$	196
7.9	Reconstructions associated with relative threshold values of 0.1, 0.2, and 0.3 for the signal $e_5\text{chirp}@4$	197
7.10	Coherence bound and measured values as function of threshold for the signal $e_{10}\text{packet}$	197
7.11	Reconstructions associated with relative threshold values of 0.1, 0.2, and 0.3 for the signal $e_{10}\text{packet}$	198
7.12	Wavelet transform magnitude for concatenation of three versions of $e_{10}\text{chirp}@1$ with SNRs of ∞ , 10, and 5dB.	200
7.13	Measured (16 trial avg.) relative error as a function of SNR for signal $e_{10}\text{chirp}@1$ using relative threshold values of 0.1, 0.2, and 0.3.	200
7.14	Time reconstruction of $e_{10}\text{chirp}@1$ from its SNR = 10dB noisy observation using a relative threshold of 0.1.	201
7.15	Time reconstruction of $e_{10}\text{chirp}@1$ from its SNR = 5dB noisy observation using a relative threshold of 0.2.	201
7.16	Wavelet transform magnitude for concatenation of three versions of $e_5\text{chirp}@4$ with SNRs of ∞ , 10, and 5dB.	202
7.17	Measured (16 trial avg.) relative error as a function of SNR for signal $e_5\text{chirp}@4$ using relative threshold values of 0.1, 0.2, and 0.3.	202
7.18	Time reconstruction of $e_5\text{chirp}@4$ from its SNR = 10dB noisy observation using a relative threshold of 0.1.	203
7.19	Time reconstruction of $e_5\text{chirp}@4$ from its SNR = 5dB noisy observation using a relative threshold of 0.2.	203
7.20	Wavelet transform magnitude for concatenation of three versions of $e_{10}\text{packet}$ with SNRs of ∞ , 10, and 5dB.	204
7.21	Measured (16 trial avg.) relative error as a function of SNR for signal $e_{10}\text{packet}$ using relative threshold values of 0.1, 0.2, and 0.3.	204
7.22	Time reconstruction of $e_{10}\text{packet}$ from its SNR = 10dB noisy observation using a relative threshold of 0.1.	205
7.23	Time reconstruction of $e_{10}\text{packet}$ from its SNR = 5dB noisy observation using a relative threshold of 0.2.	205
7.24	Computed power increase in signal due to additive noise with a SNR of 3dB.	210
7.25	Uniform quantization of a line using three bits.	214
7.26	Overcomplete wavelet transform-based scheme for compression.	215
7.27	Filter bank support functions G of Equation (6.3) for the three filter banks used in Numerical Experiment 7.14.	218
7.28	OCWT compression for signal FM1.	219
7.29	OCWT compression for signal FM2.	219
7.30	OCWT compression for signal FM3.	220

7.31 OCWT compression for signal FM4.	220
7.32 OCWT compression for signal FM5.	221
7.33 OCWT compression for signal FM6.	221
7.34 Compression for FM signal set using Daub6 orthonormal wavelet representation	222
7.35 Digital communication system using a wavelet encoder in the digital transmitter and a wavelet decoder in the digital receiver.	227
7.36 Various approaches to channel allocation: TDMA, FDMA, WPMA, and NWMA.	228
7.37 The Morlet wavelet used in the numerical illustrations of this section.	237
7.38 The wavelet modulation of a simple four-bit carrying communication signal.	238
7.39 The corresponding magnitude wavelet transform of the communication signal.	239
7.40 Modulation of a 52-bit carrying information signal.	239
7.41 Bit recovery process for the 52-bit communication signal. .	240
7.42 Magnitude wavelet transform associated with the 52-bit communication signal.	240
7.43 A 7-bit cell wavelet cluster in the time–frequency domain configured such that the center cell carries a value of 1 and adjacent cells carry values of 0 (i.e., the 1 0 configuration). .	243
7.44 A 7-bit cell wavelet cluster in the time–frequency domain configured such that the center cell carries a value of 0 and adjacent cells carry values of 1 (i.e., the 0 1 configuration). .	243
7.45 A more typical arrangement of time–frequency rectangles that contains an example of the 1 0 configuration.	244
7.46 Wavelet communication interface showing a 0 1 cluster and its corresponding time domain signal.	244
7.47 The 0 1 cluster with the wavelet transform magnitude overlaid. .	245
7.48 Noisy version, 0dB SNR, of the 0 1 cluster (overlaid with CWT).	245
7.49 Full-scale view of BER versus SNR for different noise thresholds.	247
7.50 Zoomed to transition region view of BER versus SNR for different noise thresholds.	248
7.51 Receiver operating characteristics for the wavelet communication scheme.	249
7.52 Histogram of energy distributions for the center test cell and the surrounding cells: (left) SNR = 12dB (right) SNR = 6dB. .	250
7.53 Histogram of energy distributions for the center test cell and the surrounding cells: (left) SNR = 2dB (right) SNR = 0dB. .	250
7.54 Histogram of energy distributions for the center test cell and the surrounding cells: (left) SNR = -2dB (right) SNR = -6dB. .	251

7.55	Block diagram of the identification process with a provision for determining whether an observed signal is outside the database.	255
7.56	Performance indicators for (left) the matched filter and (right) the OCWT identification schemes.	258
8.1	Architecture of the Wavelet Signal Processing Workstation.	264
8.2	WSPW startup screen.	266
8.3	Loading a FM signal into the WSPW.	267
8.4	A symmetric about 0 uniform sample set	270
8.5	Displaying using plot: A 2×1 sampled signal consisting of the chirp and packet signals.	274
8.6	Arithmetic combinations of members of p	275
8.7	Relational operations on members of p	277
8.8	FFT of the 2×1 sampled signal matrix.	278
8.9	Resampling the chirp and packet signals with the sample structure [1/5 5] (severely undersampled).	280
8.10	Resampling the chirp and packet signals with the sample structures [1/500 5].	280
8.11	Convolution of the packet and chirp signals.	282
8.12	Translated by 2 version of p	283
8.13	Noisy versions of the sampled signal object p , SNRs = ∞ , 6, and 0.	285
8.14	Time domain alignment of the chirp and packet signals.	288
8.15	Bandpassed versions of the chirp and packet signals.	289
8.16	Derivative of the chirp and packet signals.	290
8.17	Integral of the real parts of the chirp and packet signals.	291
8.18	Fourier transform of the 20MHz upshifted chirp and packet signals.	292
8.19	Power as a function of index (0-2) for the perturbed signals chirp (top) and packet (bottom).	293
8.20	Graphical user interface for specifying wavelet filter banks.	297
8.21	Wavelet transform magnitude of the chirp and packet signals.	302
8.22	Magnitude of the Morlet filter bank transfer functions.	303
8.23	Wavelet transform of the chirp signal p	306
8.24	Wavelet transform of upshifted by 10 chirp signal $e_{10}p$	307
8.25	Reconstruction of chirp signal having significant energy outside the filter bank frequency support; error is on the order of 10^{-4}	308
8.26	Reconstruction of filtered chirp signal having almost all of its energy inside the filter bank frequency support; error drops to below 10^{-13}	309

1

Introduction

1.1 Motivation and Objectives

Although the theory of wavelet analysis is a relatively new and still evolving discipline, there is a deep and sophisticated body of work currently available. Much of this work, however, requires a fairly in-depth knowledge of several areas of advanced mathematics and hence limits its accessibility. It is a main objective of this work to strike a balance between accessibility and mathematical rigor that sacrifices as little as possible of both. To help achieve this goal, the dissemination of the material is provided by a hybrid combination of traditional (text) and nontraditional (Internet and electronic) media.

Despite the project's multifaceted nature, the traditional text component is designed as the primary vehicle for delivery of the material. This has been done with the intent that the text be useful as a standalone reference. Supporting the text is the electronic component of the material that provides a dynamic and interactive aspect. It consists of both software and (Web-accessible) hypertext documents. In this way interactive illustration of signal processing concepts and techniques are provided in an effective, compelling, and practically useful way.

An underlying goal of the material presented in this work is to provide a bridge between the theory and practice of signal processing with particular emphasis on generally overcomplete wavelet techniques. Despite the fact that the *practice* of signal processing is a wide and varied one, the term *practical signal processing* is used in the context of this material with a very specific meaning: the numerical implementation of techniques for signal manipulation on a finite precision digital machine.

1.2 Core Material and Development

A main thread of this work is the idea that overcomplete systems of wavelets are a rich and largely unexplored area that have great benefits to offer in

many applications. Starting from the continuous wavelet transformation, a mathematically sound theory of the discretization of analog signals is developed. The development yields a rich family of signal representations and leads naturally to computer implementations. The discussion is at that of a senior or beginning graduate student level and is accessible to professionals in the signal processing community.

Numerical illustration of concepts and techniques are facilitated through software reproducible examples, interactive graphical user interfaces, as well as traditional static figures in the text. This work is geared towards practical application and numerical implementation of wavelet-based algorithms. As such it includes working interactive software demonstrations available from the Internet-accessible Web site:

www.birkhauser.com/book/ISBN/0-8176-3909-8

1.3 Hybrid Media Components

This text is but one piece of a larger body of material presented in a hybrid media form consisting of print, electronic, and software components. Specifically the material consists of

- an expository theoretical treatment of the discrete and continuous wavelet transformations with an emphasis on discretization through sampling of the continuous wavelet transform;
- an applications-oriented presentation illustrated via numerical examples on synthetic and real data; and
- an electronic component in the form of an Internet-accessible Web page that includes down-loadable MATLAB-based code with the following capabilities:
 - (i) reproducing examples presented in the text,
 - (ii) conducting numerical experiments as suggested in the text,
 - (iii) applying algorithms described in the text on data provided by the user, and
 - (iv) designing new algorithms from component modules on user-defined signal processing tasks.

Despite the fact that this text is just part of the entire hybrid media work, it is meant to be a self-contained document. The electronic components, however, are an invaluable supplement to the text and shall be updated and modified continually.

1.4 Signal Processing Perspective

1.4.1 Analog Signals

A main philosophy adhered to in this book is the idea that the fundamental underlying objects of interest are analog (as opposed to discrete or digital) signals.

In most real-world applications the fundamental signals of interest are analog in nature. For example, the variation in air pressure caused by a sound source or the intensity of electromagnetic energy reflected by an object illuminated by an active source may be considered as analog signals. An analog signal is one whose domain has no measurable gaps, for example, $f(t)$ where t may take arbitrary values over the whole real line. In contrast a discrete signal is one whose domain is restricted to a countable set of points, for example, $f(t_n)$ where n is restricted to be an integer.

Whether signals come from the audio, visible and/or infrared, or microwave areas of the electromagnetic spectrum, *signal processing* generally involves a prescribed manipulation in order to achieve some useful goal such as communication, compression, or information extraction. Depending on the area of the electromagnetic spectrum of interest, such manipulations may be identified with sound, image, and radar signal processing.

1.4.2 Digital Processing of Analog Signals

Although the signals of interest in many applications are inherently analog in nature, digital platforms have fast become the primary vehicle for implementation of signal processing algorithms and techniques. This situation is due not only to the proliferation and ever-increasing computational power to cost ratio of digital platforms, but also to the established and demonstrable benefits of digital processing. Perhaps the best example of this comes from the audio (voice and music) reproduction industries ([Gib93]) in which the superiority of digital coding for both communication and archival (via compact disc) has been firmly established. Reproductions based on digital techniques attain a level of fidelity unsurpassed by former analog techniques. Other benefits associated with digital based signal processing include the abilities to

- manipulate signals via digital processors,
- store/archive signals on digital media,
- propagate signals via digital networks (e.g., the Internet), and
- achieve a high degree of noise robustness.

Digital techniques and processing offer a host of desirable qualities. On the other hand, signals of interest are fundamentally analog in nature.

Clearly, there exists a gap that must be bridged in order to process analog signals via digital platforms. In order to manipulate a signal via a digital platform it is not only necessary to discretize the domain and range of the signal, but also to restrict the extent of those discretizations to some finite interval. The process of doing so yields a discrete representation of the underlying analog signal.

In practice, the analog-to-digital gap is routinely bridged by the direct digital sampling of analog signals. Theoretically, the direct sampling and reconstruction (of bandlimited signals) is completely understood via the classical sampling theorem (viz. Theorem 2.6 on page 23); and what's more, the theory is successfully and widely implemented in practical systems.

Because of these facts there is strong justification for focusing attention purely on the digital domain processing of discrete signals under the three-step processing model of

1. sample all analog signals as prescribed by the sampling theorem (A/D),
2. manipulate data in the digital domain, and,
3. (possibly) transform back to the analog domain (D/A).

Taking this view necessarily has the consequence of ignoring the analog origins of the signals. A contention held here is that there are both practical benefits and theoretical insights to be gained by considering the process as a whole from its analog domain of origin. From a theoretical point of view, in many respects, it is easier to deal directly with the original analog space than its discrete counterpart. In fact, there is a wealth of existing theory and understanding associated with analog spaces, for example, spaces of bandlimited or finite energy functions.

In this more general context, methods other than direct sampling may be considered for discretizing an analog signal. One drawback of direct digital sampling is that a fixed frequency extent (bandwidth) is supposed on the signal over its (infinite) duration. Accordingly, the sampling theorem requires that the signal be (uniformly) sampled at a rate inversely proportional to this fixed frequency extent.

Thus, both periods of high frequency content and low frequency content are sampled at a rate that is governed by the highest frequency content of the signal. This leads to the intuitively unappealing situation that many signals may be “oversampled” over most of their duration. As an example, consider a signal that has a burst of high frequency energy highly concentrated in time. The direct sampling of this signal is deficient in the sense that the sampling rate is required to be constant, resulting in critical sampling over the burst and oversampling elsewhere. Stated succinctly, direct sampling is insensitive to fluctuations over time in the frequency content of a signal.

Signals that exhibit structured time–frequency behaviors abound in natural and manmade systems. For such signals the fluctuation of their frequency content over time varies coherently. As an example consider the sound of a gong or steam engine whistle. In both cases their characteristic sound may be understood in terms of a coherent fluctuation of frequency content over time. Similar statements can be made for speech, radar, and many signals generated by mechanical means. Such signals are said to be *time-frequency coherent*.

For the discretization (and processing) of time–frequency coherent signals, the wavelet transform offers itself as a natural tool. For one-dimensional signals the (continuous) wavelet transform yields a two dimensional function of time and frequency. At any fixed value of time, the magnitude of the wavelet transform indicates the frequencies present in the signal. In this way, a wavelet transform has the ability to expose the time–frequency content of a signal. An alternative to the direct sampling of a time–frequency coherent signal is to first (in the analog domain) *continuously* wavelet transform the signal and then sample the wavelet transform. Sampling strategies may then be prescribed that are sensitive to the signal’s time–frequency content.

1.4.3 Time-Frequency Limitedness

A basic assumption of direct sampling is that the analog signal to be sampled is bandlimited. On the one hand, it is intuitive that practical signals can have neither infinite duration nor infinite bandwidth; yet, on the other hand, fundamental mathematical considerations preclude the existence of simultaneously timelimited and bandlimited signals. This is the so-called paradox of simultaneously timelimited and bandlimited signals. One cause of this paradox comes from the very concept of limitedness itself, that is, the idea that a signal is *exactly* zero outside some finite interval. From a practical viewpoint, it is not possible to measure a signal to enough accuracy to determine if it is exactly zero and, hence, assuming so is nothing more than a mathematical convenience. An assumption of limitedness has ramifications that may lead to various paradoxes and must therefore be used with caution. Even so, it is undeniable that real-world signals are of finite duration. A possible resolution to this dilemma may be attained by taking a more practical, less stringent, definition of duration that allows the signal to be nonzero outside a finite interval yet requires that the most significant portion of the signal be resident in a finite interval.

Such a view is presented by David Slepian in [Sle76] who introduces the idea of ϵ -distinguishability between two signals. In particular, two signals of time $f(t)$ and $g(t)$ are *distinguishable* if their difference has sufficient energy; that is,

$$\int_{-\infty}^{\infty} |f(t) - g(t)|^2 > \epsilon.$$

This concept leads, in turn, to obvious definitions of ϵ timelimitedness and bandlimitedness (cf. Section 7.3.4). In this realm, the prolate spheroidal wave functions ([SP61],[SL61]) play a key role. The interested reader is referred to [Sle76].

2

Mathematical Preliminaries

This chapter lays the basic mathematical foundations of the book, establishes notation, and defines various spaces and operators necessary for the development of the material. It is meant more as a reference and index of notation than a comprehensive mathematical introduction. Most of the basic notation and concepts that are used may be found in standard texts on real analysis ([Roy68], [Ben76]), operator theory ([GG80]), and discrete signal processing ([OS75], [Tre76]). The reader is assumed to have some familiarity with complex analysis and linear algebra.

Although generalization to multidimensional signals is possible, the analysis throughout the text is confined to one-dimensional signals.

2.1 Basic Symbols and Notation

\mathbb{Z} denotes the integers.

\mathbb{R} denotes the real numbers.

$\mathbb{R}^+ \triangleq \{t \in \mathbb{R} : t > 0\}$ denotes the strictly positive real numbers.

$\mathbb{R}^- \triangleq \{t \in \mathbb{R} : t < 0\}$ denotes the strictly negative real numbers.

\mathbb{C} denotes the complex numbers and $j = \sqrt{-1}$.

The complex conjugate of $z \in \mathbb{C}$ is denoted \bar{z} and the magnitude of z is denoted $|z|$.

The support, $\text{supp } f$, of a function f is the closure of the set of domain values for which the function is nonzero; that is, $\text{supp } f \triangleq \overline{\{t : |f(t)| > 0\}}$, where \overline{S} denotes the closure of a set S .

If S is a set then the *characteristic* function 1_S of the set S is defined as

$$1_S(x) = \begin{cases} 1, & x \in S, \\ 0, & \text{otherwise.} \end{cases}$$

The function $\delta_{m,n}$ is the Kronecker delta function defined as

$$\delta_{m,n} = \begin{cases} 1, & m = n, \\ 0, & m \neq n. \end{cases}$$

A standard “continuous” counterpart to the Kronecker delta function is the Dirac delta distribution or “function.” For a mathematically precise formulation of the Dirac delta the reader is referred to [Ben96, Chapter 2] for an comprehensive treatment. Here $\delta(t)$ is formally defined¹ in the limit as

$$\delta(t) = \lim_{\Delta \rightarrow 0} \frac{1}{\Delta} \cdot 1_{(-\Delta/2, \Delta/2)}(t),$$

where convergence is in the distributional sense and has the so-called sifting property

$$f(0) = \int f(t) \delta(t) dt.$$

2.2 Basic Concepts

Several fundamental mathematical concepts key to the discussion are listed briefly in this section. Throughout this section, \mathcal{H} denotes a linear space (or vector space) with an inner product. An example of such an inner product space is $\mathcal{H} = \mathbb{C}^n$ under the usual inner product $\sum x_k \bar{y}_k$ for $x, y \in \mathbb{C}^n$.

2.2.1 Norm

A *norm* $\|\cdot\|$ is a mapping $\mathcal{H} \mapsto \mathbb{R}^+ \cup \{0\}$ that assigns to any element $f \in \mathcal{H}$ a nonnegative real number $\|f\|$. With $x, y \in \mathcal{H}$ and $a \in \mathbb{C}$ a norm satisfies the properties:

1. (Nonnegativity) $\|x\| \geq 0$ and $\|x\| = 0 \iff x = 0$,
2. (Scaling) $\|ax\| = |a| \cdot \|x\|$, and
3. (Triangle Inequality) $\|x + y\| \leq \|x\| + \|y\|$.

As a consequence of these properties a norm satisfies the parallelogram law.

Parallelogram Law

$$\|x + y\|^2 + \|x - y\|^2 \leq 2(\|x\|^2 + \|y\|^2).$$

2.2.2 Inner Product

An inner product $\langle \cdot, \cdot \rangle$ is a mapping $\mathcal{H} \times \mathcal{H} \mapsto \mathbb{C}$ that satisfies the conditions:

¹This is but one possibility. In fact any approximate identity may be used here ([Ben96]).

1. $\langle ax + by, z \rangle = a \langle x, z \rangle + b \langle y, z \rangle,$
2. $\langle x, y \rangle = \overline{\langle y, x \rangle},$ and
3. $\langle x, x \rangle = \|x\|^2,$

for all $x, y, z \in \mathcal{H}$ and $a, b \in \mathbb{C}.$

If $\langle x, y \rangle = 0$ then x and y are said to be *orthogonal* and may be written as $x \perp y.$ As a consequence of these properties an inner product satisfies the Cauchy–Schwarz inequality and the Pythagorean theorem.

Cauchy–Schwarz Inequality

$$|\langle x, y \rangle| \leq \|x\| \cdot \|y\|.$$

Pythagorean Theorem

$$\langle x, y \rangle = 0 \implies \|x + y\|^2 = \|x\|^2 + \|y\|^2.$$

2.2.3 Convergence

Let $x_n \in \mathcal{H}, n \in \mathbb{Z}$ be a sequence in $\mathcal{H}.$ That the sequence $\{x_n\}$ converges to x means

$$\|x - x_n\|_{\mathcal{H}} \rightarrow 0$$

as $n \rightarrow \infty,$ and may be written as $x_n \rightarrow x$ in \mathcal{H} or $x = \lim x_n.$ If $\lim x_n$ is an element of the space \mathcal{H} for all converging sequences $\{x_n\}$ then \mathcal{H} is said to be *closed*.

The sequence $\{x_n\}$ is a *Cauchy* sequence if $\|x_n - x_m\| \rightarrow 0$ as $n, m \rightarrow \infty.$ Although all convergent sequences are Cauchy in a normed vector space $\mathcal{H},$ all Cauchy sequences need not converge. If all Cauchy sequences do converge in \mathcal{H} then \mathcal{H} is said to be *complete.* Complete normed vector spaces are *Banach* spaces. Furthermore, Banach spaces that admit an inner product are *Hilbert* spaces.

2.2.4 Hilbert Spaces

The main spaces of interest here are Hilbert spaces.² The reader unfamiliar with the notion of Hilbert spaces may think of them as a generalization of ordinary three-dimensional Euclidean space in which the concepts of distance (norm) and “angle” (inner product) between two of its elements are extended. Mathematically, a Hilbert space \mathcal{H} is a complete³ normed (with norm $\|\cdot\|_{\mathcal{H}}$) vector space that has an inner product.

²An implicit assumption on the Hilbert spaces which are considered is that they admit dense countable subsets, that is, that they be *separable.*

³A vector space is complete if all Cauchy sequences converge to a limit in the space. (Recall that complete, normed vector spaces are *Banach* spaces.) See [Roy68] for precise mathematical details or [Tre76], [VK95] for an overview.

Along with some general spaces, the main Hilbert spaces of interest and their associated inner products are given in the following section.

2.3 Basic Spaces

Of particular interest are the following function spaces: $L^1(\mathbb{R})$, $L^\infty(\mathbb{R})$, the general Hilbert spaces $L^2(\mathbb{R})$, $\ell^2(\mathbb{Z})$, $\ell^2(J)$ ($J \subseteq \mathbb{Z}$ finite), and PW_Ω (bandlimited space) with specified inner products, and the Banach space of all linear bounded operators mapping one Hilbert space to another. These are all discussed in the following subsections.

2.3.1 Bounded Functions

$L^\infty(\mathbb{R})$ is the space of complex-valued (essentially) bounded signals⁴ defined on the real line \mathbb{R} . The *norm* of an element $f \in L^\infty(\mathbb{R})$ is

$$\|f\| \triangleq \text{ess sup } |f(t)| < \infty,$$

where the essential supremum is over the whole real line.

2.3.2 Absolutely Integrable Functions

$L^1(\mathbb{R})$ is the space of complex-valued absolutely integrable signals defined on the real line. The *norm* of an element $f \in L^1(\mathbb{R})$ is

$$\|f\| \triangleq \int |f(t)|dt < \infty,$$

where integration⁵ is over the whole real line \mathbb{R} .

The Fourier transform \hat{f} of a function $f \in L^1(\mathbb{R})$ is well-defined as

$$\hat{f}(\gamma) = \int f(t)e^{-j2\pi\gamma t}dt$$

since the integral converges absolutely. Fourier transforms are discussed further in Section 2.6.

⁴The terms *signal* and *function* are used interchangeably.

⁵As a convention, unspecified limits on integrations, summations, or sequences are implicitly defined to cover the entire domain associated with the function in question; for example, $\{\phi_n\}$ means $\{\phi_n\}_{n \in \mathbb{Z}}$ and $\int \phi_n$ means $\int_{-\infty}^{\infty} \phi_n$ for $\phi_n \in L^2(\mathbb{R})$.

2.3.3 Finite Energy Functions

$L^2(\mathbb{R})$ is the space of complex-valued finite energy signals defined on the real line \mathbb{R} . The *norm* of an element $f \in L^2(\mathbb{R})$ is

$$\|f\| \triangleq \left(\int |f(t)|^2 dt \right)^{1/2} < \infty,$$

where integration is over the whole real line \mathbb{R} , and the *inner product* of $f, g \in L^2(\mathbb{R})$ is $\langle f, g \rangle = \int f(t)\bar{g}(t)dt$.

2.3.4 Finite Energy Periodic Functions

Often it is desirable to consider the T -periodization, $\sum f(t + nT)$, of functions f such that attention need only be given to the symmetric interval $(-T/2, T/2)$. The concept of finite energy may be extended to periodic signals by limiting consideration of the signal to a single period. $L^2(-T/2, T/2)$ is the space of functions with finite energy over $(-T/2, T/2)$. The *norm* of an element $f \in L^2(-T/2, T/2)$ is

$$\|f\| \triangleq \left(\int_{-T/2}^{T/2} |f(t)|^2 dt \right)^{1/2} < \infty.$$

\mathcal{P}_T is the space of complex-valued finite energy signals defined on the real line whose T -periodizations are again of finite energy, i.e.,

$$\mathcal{P}_T \triangleq \left\{ f \in L^2(\mathbb{R}) : \sum f(t + nT) \in L^2(-T/2, T/2) \right\}.$$

Here, convergence of the sum to an element in $L^2(-T/2, T/2)$ means

$$\lim_{N \rightarrow \infty} \sum_{n=-N}^N f(t + nT)$$

converges absolutely and uniformly in L^2 on compact sets. Recall that a *compact* set is one that is closed and bounded.

2.3.5 Time-Frequency Concentrated Functions

$\text{TF}(\mathbb{R})$ is the space of complex-valued finite energy signals defined on the real line that decay faster than $1/t$ simultaneously in the time and frequency domains defined explicitly as

$$\begin{aligned} \text{TF}(\mathbb{R}) \triangleq \{ f \in L^2(\mathbb{R}) : & |f(t)| < C(1+|t|)^{-(1+\epsilon)} \text{ and} \\ & |\widehat{f}(\gamma)| < C(1+|\gamma|)^{-(1+\epsilon)} \text{ for } C < \infty, \epsilon > 0 \}. \end{aligned}$$

The inner product on $\text{TF}(\mathbb{R})$ is inherited from $L^2(\mathbb{R})$. It is not difficult to show that $\text{TF}(\mathbb{R}) \subseteq L^1(\mathbb{R}) (\subseteq L^2(\mathbb{R}))$ and that for all $f \in \text{TF}(\mathbb{R})$ both f and \widehat{f} are continuous.

2.3.6 Finite Energy Sequences

$\ell^2(\mathbb{Z})$ is the space of complex-valued finite energy sequences defined on the integers \mathbb{Z} . The *norm* of an element $c \in \ell^2(\mathbb{Z})$ is

$$\|c\| \triangleq \left(\sum |c_n|^2 \right)^{1/2} < \infty,$$

where summation is over \mathbb{Z} , and the *inner product* of $c, d \in \ell^2(\mathbb{Z})$ is $\langle c, d \rangle = \sum c_n \bar{d}_n$.

$\ell^2(J)$ is the space of complex-valued finite energy sequences defined on the finite set $J \subseteq \mathbb{Z}$. The *norm* of an element $c \in \ell^2(J)$ is

$$\|c\| \triangleq \left(\sum_{n \in J} |c_n|^2 \right)^{1/2} < \infty,$$

and the *inner product* of $c, d \in \ell^2(J)$ is $\langle c, d \rangle = \sum_{n \in J} c_n \bar{d}_n$. Typically, J is a sequentially ordered set of indices such as $J = \{1, 2, \dots, N\}$, $J = \{0, 1, \dots, N-1\}$, or $J = \{-N/2, -N/2+1, \dots, N/2-1\}$, where N is usually a power of 2.

2.3.7 Bandlimited Functions

Paley and Wiener ([PW34]) have made significant fundamental contributions to Fourier theory and, in particular, the understanding of bandlimited functions. For this reason, spaces of bandlimited functions are labeled as “Paley–Wiener” spaces. Bandlimited functions are defined as those that have compact (closed and bounded) support in the frequency domain (i.e., their Fourier transforms vanish off a compact set). For $\Omega > 0$, PW_Ω is the Paley–Wiener space defined as

$$PW_\Omega \triangleq \left\{ f \in L^2(\mathbb{R}) : \text{supp } \hat{f} \subseteq [-\Omega, \Omega] \right\},$$

with the L^2 inner product.

2.3.8 Hardy Spaces

Two further spaces of interest are the Hardy spaces $H_+^2(\mathbb{R})$ and $H_-^2(\mathbb{R})$ given as

$$H_+^2(\mathbb{R}) = \left\{ f \in L^2(\mathbb{R}) : \text{supp } \hat{f} \subset [0, \infty) \right\}$$

and

$$H_-^2(\mathbb{R}) = \left\{ f \in L^2(\mathbb{R}) : \text{supp } \hat{f} \subset (-\infty, 0] \right\},$$

where $\text{supp } \widehat{f}$ is the support of \widehat{f} . Let f_+ be an arbitrary element of $H_+^2(\mathbb{R})$ and f_- an arbitrary element of $H_-^2(\mathbb{R})$. The Hardy space $H_+^2(\mathbb{R})$ is a Hilbert space with norm

$$\|f_+\|_{H_+^2(\mathbb{R})} = \left(\int_0^\infty |\widehat{f}_+(\gamma)|^2 \right)^{1/2}$$

with the inner product it induces. The Hardy space $H_-^2(\mathbb{R})$ is a Hilbert space with norm

$$\|f_-\|_{H_-^2(\mathbb{R})} = \left(\int_{-\infty}^0 |\widehat{f}_-(\gamma)|^2 \right)^{1/2}$$

with the inner product it induces. Also $H_-^2(\mathbb{R})$ and $H_+^2(\mathbb{R})$ are orthogonal complements in $L^2(\mathbb{R})$; that is, $L^2(\mathbb{R}) = H_-^2(\mathbb{R}) \oplus H_+^2(\mathbb{R})$. Consequently, for an arbitrary element f in $L^2(\mathbb{R})$ there are elements $f_- \in H_-^2(\mathbb{R})$ and $f_+ \in H_+^2(\mathbb{R})$ so that $f = f_- + f_+$ and $\|f\|^2 = \|f_-\|_{H_-^2(\mathbb{R})}^2 + \|f_+\|_{H_+^2(\mathbb{R})}^2$.

Remark 2.1 Any real element of $L^2(\mathbb{R})$ may be associated with either $H_+^2(\mathbb{R})$ or $H_-^2(\mathbb{R})$ since all real functions have Fourier transforms that are involutive; that is, if f is real then $\widehat{\bar{f}}(\gamma) = \widehat{f}(-\gamma)$. Thus, real functions are uniquely determined by their Fourier transforms on $[0, \infty)$ or $(-\infty, 0]$. One such association may be realized via the Hilbert transform.

2.4 Operators

Much of the analysis described in the text is operator-theoretic in nature. Specific operators of interest are defined in this section. They include the translation, dilation, modulation, and involution operators.

2.4.1 Bounded Linear Operators

$\mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$ is the Banach space of bounded linear operators that map the Hilbert space \mathcal{H}_1 to the Hilbert space \mathcal{H}_2 . The *norm* of an element $T \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$ is

$$\|T\| = \sup_{x \in \mathcal{H}_1} \frac{\|Tx\|_{\mathcal{H}_2}}{\|x\|_{\mathcal{H}_1}} < \infty.$$

Two classes of bounded linear operators specifically worthy of mention are the self-adjoint operators and the orthogonal projection operators.

Self-Adjoint

Fact 2.2 ([GG80, Theorem III.4.1]) *If $T \in \mathcal{B}(\mathcal{H}, \mathcal{H})$ is self-adjoint then*

$$\|T\| = \sup_{f \in \mathcal{H}} \frac{|\langle f, Tf \rangle|}{\|f\|^2}.$$

Orthogonal Projection

With \mathcal{H}' a subspace of a Hilbert space \mathcal{H} the operator $P_{\mathcal{H}'} : \mathcal{H} \mapsto \mathcal{H}'$ denotes the orthogonal projection operator onto \mathcal{H}' .

Fact 2.3 ([GG80, Theorem II.13.1]) *$P_{\mathcal{H}} \in \mathcal{B}(\mathcal{H}, \mathcal{H})$ is an orthogonal projection onto \mathcal{H} if and only if $P_{\mathcal{H}}^2 = P_{\mathcal{H}}$ and $P_{\mathcal{H}}$ is self-adjoint.*

2.4.2 Properties

Let \mathcal{H}_1 and \mathcal{H}_2 be arbitrary Hilbert spaces with norms $\|\cdot\|_{\mathcal{H}_1}$ and $\|\cdot\|_{\mathcal{H}_2}$ and inner products $\langle \cdot, \cdot \rangle_{\mathcal{H}_1}$ and $\langle \cdot, \cdot \rangle_{\mathcal{H}_2}$, respectively. Let T be an operator that takes a function in \mathcal{H}_1 to a function in \mathcal{H}_2 ; that is, $T : \mathcal{H}_1 \mapsto \mathcal{H}_2$.

1. The *range* of T is $T(\mathcal{H}_1) \triangleq \{Tf : f \in \mathcal{H}_1\}$.
2. The *kernel* of T is $\ker T \triangleq \{f \in \mathcal{H}_1 : Tf = 0\}$.
3. T is *injective* or *one-to-one* if $Tf = Tg$ if and only if $f = g$. If T is a linear operator then T is injective if and only if $\ker T = \{0\}$.
4. T is *surjective* or *onto* if $T(\mathcal{H}_1) = \mathcal{H}_2$.
5. T is *bijective* if it is both injective and surjective.
6. T has an *inverse* $T^{-1} : \mathcal{H}_2 \mapsto \mathcal{H}_1$ if T is bijective. In this case the inverse of T is defined as $T^{-1}g = f$ if $g = Tf$.
7. T is *continuous* if $x_n \rightarrow x$ implies $Tx_n \rightarrow Tx$. A linear operator T is bounded if and only if T is continuous.
8. The *adjoint* of T is the unique operator $T^* : \mathcal{H}_2 \mapsto \mathcal{H}_1$ so that $\langle Tf, g \rangle_{\mathcal{H}_2} = \langle f, T^*g \rangle_{\mathcal{H}_1}$ for all $f \in \mathcal{H}_1$ and $g \in \mathcal{H}_2$. T is *self-adjoint* if $T^* = T$.
9. $T \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$ is a *compact* operator if for all sequences $\{f_n : \|f_n\| = 1\} \subseteq \mathcal{H}_1$ the sequence $\{Tf_n\}$ has a converging subsequence in \mathcal{H}_2 .

$$\{f_n : \|f_n\| = 1\} \subseteq \mathcal{H}_1$$

the sequence $\{Tf_n\}$ has a converging subsequence in \mathcal{H}_2 .

10. T is a *topological isomorphism* if T is bijective, $T \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$, and $T^{-1} \in \mathcal{B}(\mathcal{H}_2, \mathcal{H}_1)$. Thus, both T and T^{-1} are continuous linear mappings.
11. T is an *isometry* if for all $f \in \mathcal{H}_1$, $\|Tf\|_{\mathcal{H}_2} = \|f\|_{\mathcal{H}_1}$.
12. T is a *unitary* map if it is linear, bijective, and an isometry. If T is unitary then $T^{-1} = T^*$.

Compositions of topological isomorphisms are again topological isomorphisms. Two Hilbert spaces \mathcal{H}_1 and \mathcal{H}_2 that are related by a topological isomorphism are topologically equivalent. Topologically equivalent spaces are in an abstract sense the same. A topological isomorphism that is norm-preserving is a unitary map. Compositions of isometries are again isometries.

2.4.3 Useful Unitary Operators

Let f be an arbitrary element of $L^2(\mathbb{R})$ and take a, b and $s \neq 0$ all to be real numbers.

1. The *translation* operator $\tau_a : L^2(\mathbb{R}) \mapsto L^2(\mathbb{R})$ is a unitary map given by

$$(\tau_a f)(t) \stackrel{\Delta}{=} f(t - a).$$

2. The *modulation* operator $e_b : L^2(\mathbb{R}) \mapsto L^2(\mathbb{R})$ is a unitary map given by

$$(e_b f)(t) \stackrel{\Delta}{=} e^{j2\pi bt} f(t).$$

3. The *dilation*⁶ operator $D_s : L^2(\mathbb{R}) \mapsto L^2(\mathbb{R})$ is a unitary map given by

$$(D_s f)(t) \stackrel{\Delta}{=} |s|^{1/2} f(st).$$

4. The *reflection* operator $\mathcal{R} : L^2(\mathbb{R}) \mapsto L^2(\mathbb{R})$ is a unitary map given by

$$(\mathcal{R} f)(t) = f(-t).$$

5. The *involution* operator $\tilde{\cdot} : L^2(\mathbb{R}) \mapsto L^2(\mathbb{R})$ is a (conjugate) unitary⁷ map given by

$$\tilde{f}(t) \stackrel{\Delta}{=} \overline{f}(-t).$$

Table 2.1 displays in a compact form these unitary operators on $L^2(\mathbb{R})$, their definitions, inverses, and Fourier transforms.

⁶Note that for this definition of dilation operator, values of $|s| > 1$ make $D_s f$ a *contracted* version of f and values of $|s| < 1$ make $D_s f$ a *ilated* version of f .

⁷Due to the conjugation required in its definition, involution is not strictly unitary since $\langle \tilde{f}, g \rangle = \overline{\langle f, g \rangle}$, yet, it is still norm-preserving and invertible.

Name	Uf	$(Uf)(t)$	$U^*f = U^{-1}f$	$(Uf)^\wedge$
Translation	$\tau_a f$	$f(t-a)$	$\tau_{-a} f$	$e_{-a} \hat{f}$
Modulation	$e_a f$	$e^{j2\pi at} f(t)$	$e_{-a} f$	$\tau_a \hat{f}$
Dilation	$D_s f$	$s^{\frac{1}{2}} f(st)$	$D_{s^{-1}} f$	$D_{s^{-1}} \hat{f}$
Reflection	$\mathcal{R} f$	$f(-t)$	$\mathcal{R} f$	$\mathcal{R} \hat{f}$
Involution	\tilde{f}	$\bar{f}(-t)$	\tilde{f}	$\bar{\hat{f}}$

TABLE 2.1. Unitary operators $U : L^2(\mathbb{R}) \mapsto L^2(\mathbb{R})$, their inverses (equal to their adjoints), and their Fourier transforms. Here $f \in L^2(\mathbb{R})$, $a, b \in \mathbb{R}$, and $s \in \mathbb{R}^+$.

Of these operators, the reflection operator \mathcal{R} is the most remarkable in that not only is it its own inverse and its own adjoint, but it is also its own Fourier transform in the sense that $(\mathcal{R}f)^\wedge = \mathcal{R}f$.

2.5 Bases and Completeness in Hilbert Space

Consider a sequence of functions $\{\phi_n\} \subseteq \mathcal{H}$ from a Hilbert space \mathcal{H} . With respect to the representation of arbitrary signals from \mathcal{H} as linear combinations of the functions in $\{\phi_n\}$, there are a number of important properties and concepts associated with such sequences. This section enumerates some of the most important.

1. The linear *span*, $\text{span}\{\phi_n\}$, of a sequence of functions $\{\phi_n\}$ is the set of functions generated from arbitrary linear combinations of $\{\phi_n\}$; that is,

$$\text{span}\{\phi_n\} = \left\{ \sum_{n=1}^N c_n \phi_n : c_n \in \mathbb{C}, N \in \mathbb{Z} \right\}.$$

2. $\{\phi_n\}$ is *dense* in \mathcal{H} if $\overline{\text{span}\{\phi_n\}} = \mathcal{H}$ ⁸.
3. $\{\phi_n\}$ is *complete* in \mathcal{H} if $\forall n$, $\langle f, \phi_n \rangle = 0$ if and only if $f = 0$ a.e.
4. $\{\phi_n\}$ is *orthonormal* in \mathcal{H} if $\forall m, n$, $\langle \phi_m, \phi_n \rangle = \delta_{m,n}$.
5. $\{\phi_n\}$ is a *Schauder basis* or *basis* for \mathcal{H} if for each $f \in \mathcal{H}$ there is a unique sequence $\{c_n\} \in \mathbb{C}$ such that $f = \sum c_n \phi_n$.

⁸Note that \overline{S} denotes the closure of the set S ; that is, all limits of converging sequences in S are included in \overline{S} . In particular, $\overline{\text{span}\{\phi_n\}}$ contains limits of the form $\lim_{N \rightarrow \infty} \sum_{n=1}^N c_n \phi_n$ for some complex scalar sequence $\{c_n\}$.

6. An orthonormal set $\{\phi_n\}$ is an *orthonormal basis* for \mathcal{H} if for every $f \in \mathcal{H}$ there is a unique sequence $\{c_n\} \in \mathbb{C}$ such that $f = \sum c_n \phi_n$, that is, a Schauder basis of orthonormal functions. If $\{\phi_n\}$ is an orthonormal basis then for every $f \in \mathcal{H}$

$$f = \sum_n \langle f, \phi_n \rangle \phi_n$$

and

$$\|f\|^2 = \sum_n |\langle f, \phi_n \rangle|^2 \quad (\text{Parseval's equality}).$$

7. A basis $\{\phi_n\}$ is a *Riesz basis* for \mathcal{H} if it is related to an orthonormal basis by a topological isomorphism; that is, there is a topological isomorphism $T : \mathcal{H} \mapsto \mathcal{H}$ such that $\phi_n = Te_n$ for all n , where $\{e_n\}$ is an orthonormal basis for \mathcal{H} .
8. A basis $\{\phi_n\}$ is an *unconditional basis* for \mathcal{H} if every convergent series of the form $\sum c_n \phi_n$ is unconditionally convergent; that is, every arrangement of its terms converges to the same element.
9. An unconditional basis $\{\phi_n\}$ is a *bounded unconditional basis* for \mathcal{H} if there are constants $0 < A \leq B < \infty$ such that $A < \|\phi_n\| < B$.
10. If $\{\psi_n\}$ is another sequence in \mathcal{H} then $\{\phi_n\}$ and $\{\psi_n\}$ are *biorthogonal* if $\langle \phi_m, \psi_n \rangle = \delta_{m,n}$.

An important set of functions that arise in signal processing and mathematical analysis is the harmonic complex exponentials $\{e^{j2\pi nt}\}$. The fact that a set of harmonic complex exponentials forms an orthonormal basis for a space of periodic finite energy functions is stated formally in the following theorem and forms the cornerstone of digital Fourier techniques and applications.

Theorem 2.4 *The harmonic complex exponentials*

$$\left\{ \frac{1}{\sqrt{T}} e_{n/T} \right\} = \left\{ \frac{1}{\sqrt{T}} e^{j2\pi nt/T} \right\}$$

form an orthonormal basis for $L^2(-T/2, T/2)$.

For a proof the reader is referred to [You80, Section 1.3, Theorem 2].

2.6 Fourier Transforms

Presented in this section are the standard Fourier transforms in their discrete, periodic, continuous, and hybrid forms. Of particular interest is the fast Fourier transform (FFT) which has vast application in digital signal processing. A mathematically sophisticated view of general harmonic analysis may be found in [Ben96].

2.6.1 Continuous Time Fourier Transform

The (continuous time) *Fourier transform* is a mapping $\mathcal{F} : L^2(\mathbb{R}) \mapsto L^2(\mathbb{R})$ and is defined rigorously as follows. For $f \in L^1(\mathbb{R}) \subset L^2(\mathbb{R})$

$$\mathcal{F}f(\gamma) = \widehat{f}(\gamma) = \int f(t)e^{-j2\pi t\gamma} dt, \quad (2.1)$$

and for $f \in L^2(\mathbb{R}) \setminus L^1(\mathbb{R})$

$$\mathcal{F}f(\gamma) = \widehat{f}(\gamma) = \lim_{n \rightarrow \infty} \int_{-n}^n f(t)e^{-j2\pi t\gamma} dt,$$

for $\gamma \in \widehat{\mathbb{R}} (\stackrel{\triangle}{=} \mathbb{R})$. In this case convergence of the integrals to $\mathcal{F}f = \widehat{f}$ is in the L^2 -sense. Hereafter, the “ $\widehat{}$ ” notation is adopted almost exclusively to indicate the forward Fourier transform and “ $\widehat{}^\vee$ ” to indicate the inverse Fourier transform. Both are defined formally in terms of inner products with complex exponentials in the following.

The Fourier transform of $f \in L^2(\mathbb{R})$ may be formally represented as

$$\widehat{f}(\gamma) = \langle f, e_\gamma \rangle_{L^2(\mathbb{R})} \quad (2.2)$$

and the inverse transform may be similarly formally represented as

$$\left(\widehat{f} \right)^\vee(t) = \langle \widehat{f}, e_{-t} \rangle_{L^2(\widehat{\mathbb{R}})}, \quad (2.3)$$

despite the fact that e_t is not an element of $L^2(\mathbb{R})$.

Two fundamental results in Fourier theory are the Plancherel and Parseval relations which assert the fact that the Fourier transform is unitary (norm-preserving, in particular). Let $f, g \in L^2(\mathbb{R})$.

Plancherel for $L^2(\mathbb{R})$

$$\int |f(t)|^2 dt = \int |\widehat{f}(\gamma)|^2 d\gamma. \quad (2.4)$$

Parseval for $L^2(\mathbb{R})$

$$\int f(t)\bar{g}(t) dt = \int \widehat{f}(\gamma)\bar{\widehat{g}}(\gamma) d\gamma. \quad (2.5)$$

These relations are generalized to spaces other than $L^2(\mathbb{R})$ in Section 2.6.5.

2.6.2 Continuous Time-Periodic Fourier Transform

A T -periodic function f satisfies $f = \tau_T f$ almost everywhere; that is,

$$f(t) = f(t + T), \text{ for almost all } t.$$

For functions f defined over the entire real line \mathbb{R} that have sufficient decay, the T -periodization $\sum \tau_{nT} f$ is recoverable from a countable set $\{\widehat{f}(n/T)\}$ consisting of samples of the Fourier transform. This relationship is illuminated by the Poisson Summation Formula (PSF) given in Theorem 2.5 and shows that periodic replication in the time (frequency) domain is equivalent to “sampling” in the frequency (time) domain ([BHW92]).

Theorem 2.5 (Poisson Summation Formula) *If $f \in \mathcal{P}_T$ then*

$$T \cdot \sum \tau_{nT} f = \sum \widehat{f}\left(\frac{n}{T}\right) e_{n/T}. \quad (2.6)$$

Proof: The proof depends on the two facts that:

- (i) $\langle \sum \tau_{nT} f, e_{n/T} \rangle_{L^2(-T/2, T/2)} = \langle f, e_{n/T} \rangle_{L^2(\mathbb{R})} = \widehat{f}(n/T)$; and
- (ii) since (by Theorem 2.4) the harmonic exponentials $\left\{\frac{e_{n/T}}{\sqrt{T}}\right\}$ form an orthonormal basis for $L^2(-T/2, T/2)$

$$\begin{aligned} \forall g \in L^2(-T/2, T/2), \quad g &= \sum \left\langle g, \frac{e_{n/T}}{\sqrt{T}} \right\rangle_{L^2(-T/2, T/2)} \frac{e_{n/T}}{\sqrt{T}} \\ &= \frac{1}{T} \sum \langle g, e_{n/T} \rangle_{L^2(-T/2, T/2)} e_{n/T}. \end{aligned}$$

Fact (i) is left as an exercise for the reader and requires that $f \in \mathcal{P}_T$ and particularly that the T periodization converges absolutely and uniformly on compact sets. Combining (i) and (ii) for $g = \sum \tau_{nT} f$ yields

$$\sum \tau_{nT} f = \frac{1}{T} \sum \widehat{f}\left(\frac{n}{T}\right) e_{n/T}. \quad \blacksquare$$

Via the PSF, a Fourier transform that relates continuous time $2T$ periodic functions to discrete finite energy sequences may be defined. This map $\mathcal{F} : \mathcal{P}_{2T} \subseteq L^2(-T, T) \mapsto \ell^2(\mathbb{Z})$ is called the continuous time periodic Fourier transform (CTPFT) and is defined by

$$(\mathcal{F}f)_n = \frac{1}{\sqrt{2T}} \int_{-T}^T f(t) e^{-j2\pi tn/(2T)} dt \quad (2.7)$$

$$= \frac{1}{\sqrt{2T}} \langle f, e_{n/(2T)} \rangle_{L^2(-T, T)}. \quad (2.8)$$

2.6.3 Discrete Time Fourier Transform

The Discrete Time Fourier Transform (DTFT) is a map $\mathcal{F} : \ell^2(\mathbb{Z}) \mapsto L^2(-1/2, 1/2)$ defined as

$$\mathcal{F}c = \widehat{c} \triangleq \sum_n c_n e^{-jn}, \quad (2.9)$$

where $c = \{c_n\} \in \ell^2(\mathbb{Z})$. Commonly the DTFT of a sequence c is denoted C ; that is, $C = \widehat{c}$. Clearly the DTFT is a 1-periodic function since it is a linear combination of 1-periodic functions (the harmonic complex exponentials $\{e^{-jn}\}$).

2.6.4 Discrete Fourier Transform

One reason that Fourier techniques are so prevalent in digital signal processing is the existence of efficient algorithms (Fast Fourier Transforms (FFTs)) for the computation of the Discrete Fourier Transform (DFT). Formally, a DFT is a map

$$\mathcal{F} : \ell^2(\{0, 1, \dots, N-1\}) \mapsto \ell^2(\{0, 1, \dots, N-1\})$$

defined as

$$(\mathcal{F}x)_k = \widehat{x}_k = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n e^{j2\pi n \cdot k / N}, \quad k = 0, 1, \dots, N-1,$$

where $x, \widehat{x} \in \ell^2(\{0, 1, \dots, N-1\})$.

2.6.5 Fourier Dual Spaces

The Plancherel and Parseval relations introduced in Section 2.6.1 are generalized in this section to spaces other than $L^2(\mathbb{R})$. In particular, pairs of spaces may be identified via appropriate Fourier transforms in which the Plancherel and Parseval relations hold. These pairs of spaces are called Fourier dual spaces and are isomorphic. In an abstract sense isomorphic spaces are the same. Some Fourier dual spaces and the Fourier transform (isomorphism) that relates them are listed in Table 2.2.

In an inner product form, Equations (2.4) and (2.5) are quite general in that they hold for all pairs \mathcal{H} and $\widehat{\mathcal{H}}$ of Fourier dual spaces.

Plancherel

For all $f \in \mathcal{H}$,

$$\|f\|_{\mathcal{H}}^2 = \|\widehat{f}\|_{\widehat{\mathcal{H}}}^2. \quad (2.10)$$

\mathcal{H}	$\widehat{\mathcal{H}}$	$\mathcal{F} : \mathcal{H} \mapsto \widehat{\mathcal{H}}$
$L^2(\mathbb{R})$	$L^2(\mathbb{R})$	
$\text{TF}(\mathbb{R})$	$\text{TF}(\mathbb{R})$	CTFT
PW_Ω	$L^2(-\Omega, \Omega)$	
$L^2(-T, T)$	$\ell^2(\mathbb{Z}/2T)$	CTPFT
$L^2(-1/2, 1/2)$	$\ell^2(\mathbb{Z})$	
$\ell^2(\mathbb{Z}/2\Omega)$	$L^2(-\Omega, \Omega)$	DTFT
$\ell^2(\mathbb{Z})$	$L^2(-1/2, 1/2)$	
$\ell^2(\{0, 1, \dots, N-1\})$	$\ell^2(\{0, 1, \dots, N-1\})$	DFT

TABLE 2.2. Fourier dual space relations.

Parseval

For all $f, g \in \mathcal{H}$,

$$\langle f, g \rangle_{\mathcal{H}} = \langle \widehat{f}, \widehat{g} \rangle_{\widehat{\mathcal{H}}}. \quad (2.11)$$

2.7 Linear Filters

2.7.1 Continuous Filters and Fourier Transforms

The convolution $f * g$ of two functions $f, g \in L^2(\mathbb{R})$ is

$$(f * g)(t) = \int f(x)g(t-x)dx.$$

Linear time invariant systems are characterized by their impulse responses, that is, their output in response to an impulsive input. Given the impulse response $g \in L^2(\mathbb{R})$ of a linear system Σ , the output $y \in L^2(\mathbb{R})$ to an arbitrary input signal f is the convolution $y = f * g$ as depicted in Figure 2.1.

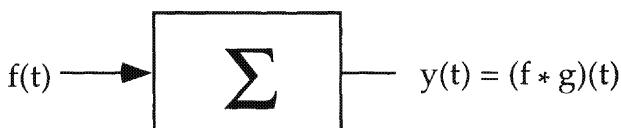


FIGURE 2.1. Continuous linear system output as convolution.

An important property associated with linear time invariant systems is that convolution in the time domain is equivalent to multiplication in the frequency domain:

$$(f * g) \hat{=} \hat{f} \cdot \hat{g}.$$

2.7.2 Discrete Filters and Z-Transforms

The *convolution* $x * h$ of two elements $x, h \in \ell^2(\mathbb{Z})$ is

$$(y * h)_k = \sum_n x_n h_{k-n}.$$

Just as continuous linear systems are characterized by their impulse responses, discrete linear systems are also characterized by their impulse (Dirac-delta) responses. Given the impulse response $h \in \ell^2(\mathbb{Z})$ of a linear system Σ_d , the output $y \in \ell^2(\mathbb{Z})$ to an arbitrary input signal f is the convolution $y = f * h$ as depicted in Figure 2.2.

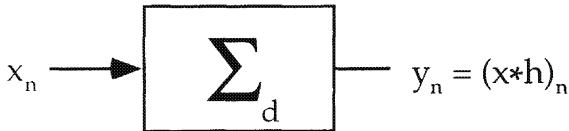


FIGURE 2.2. Discrete linear system output as convolution.

Z-transforms and discrete filters are fundamental to digital signal processing and analysis. The Z-transform $\mathcal{Z}h$ of sequence $h = \{h_n\} \in \ell^2(\mathbb{Z})$ is

$$(\mathcal{Z}h)(z) \triangleq H(z) \triangleq \sum_n h_n z^{-n},$$

where z runs over the entire complex plane \mathbb{C} and the sum on the right-hand side makes sense.

An important property associated with discrete linear systems is that convolution in the time domain is equivalent to multiplication in the Z-domain:

$$\mathcal{Z}(x * h) = \mathcal{Z}x \cdot \mathcal{Z}h.$$

Note that the DTFT $\mathcal{F}c$ of a sequence $c = \{c_n\}$ is its Z-transform evaluated in the unit circle; that is,

$$(\mathcal{F}c)(\gamma) = (\mathcal{Z}c)(e^{j2\pi\gamma}).$$

2.8 Analog Signals and Discretization

Numerically, analog signals are represented through their samples on a discrete set that is dense enough with respect to their frequency content.

This section introduces the computational concept of the sample structure for bandlimited analog signals and indicates its relation to the classical sampling theorem and the software component of the material (viz. Chapter 8).

2.8.1 Classical Sampling Theorem

Uniform sampling in spaces of bandlimited functions (i.e., the Paley–Wiener spaces) has a mature and well-understood theory associated with it. Known to Whittaker ([Whi15]) in 1915, and applied later in 1933 by Kotelnikov ([Kot33]) to communication problems, the so-called “Sampling Theorem” did not find fame until the 1949 landmark paper of Shannon ([Sha49]). The Classical Sampling Theorem establishes that any Ω bandlimited function f may be fully represented in terms of its uniform samples $\{f(n\Delta)\}$ for a sufficiently small sampling interval $\Delta > 0$ and, moreover, may be recovered as a sample weighted linear combination of sinc functions. The theorem is stated precisely as Theorem 2.6.

Theorem 2.6 (Classical Sampling Theorem) *Let $\Omega > 0$ and $2\Omega\Delta < 1$. Then any $f \in PW_\Omega$ may be reconstructed from its uniform samples $\{f(n\Delta)\}$ as*

$$f = \Delta \sum f(n\Delta) \tau_{n\Delta} d_{2\pi\Omega},$$

where the sum converges in L^2 and $d_{2\pi\Omega}$ is a dilated version of the Dirichlet kernel.

The Dirichlet kernel (or sinc function) is given by

$$d(\gamma) = \frac{\sin(\gamma)}{\pi\gamma}$$

and d_λ is defined as

$$d_\lambda(\gamma) \triangleq \lambda d(\lambda\gamma) = \frac{\sin(\lambda\gamma)}{\pi\gamma}.$$

Thus, d_λ has a Fourier transform that is an ideal pulse of bandwidth λ/π ; that is,

$$\widehat{d}_\lambda \triangleq 1_{[-\lambda/(2\pi), \lambda/(2\pi)]}. \quad (2.12)$$

Explicitly,

$$d_{2\pi\Omega}(x) \triangleq \frac{\sin(2\pi\Omega x)}{\pi x}.$$

A proof of the classical sampling theorem may be made that uses the PSF, however, it is deferred to a later section where a more general sampling theorem is proved. The classical sampling theorem answers the question:

what conditions on the set $\{n\Delta\}$ allow the recovery of f from $\{f(n\Delta)\}$? The condition given by the classical sampling theorem is that the sampling density Δ^{-1} must be at least as large as twice the value of the cutoff frequency Ω ; that is, $\Delta^{-1} \geq 2\Omega$. The quantity 2Ω is the so-called *Nyquist density*. Note that in the case $2\Delta\Omega = 1$ the set of functions $\{\sqrt{\Delta}\tau_{n\Delta}d_{2\pi\Omega}\} = \{\sin(2\pi\Omega(x - n\Delta))/(2\sqrt{2\Omega}\pi(x - n\Delta))\}$ is an orthonormal basis for the space of functions bandlimited by Ω .

2.8.2 What Can Be Computed Exactly?

In this section the question of computing the Fourier transform \hat{f} of an analog signal f on a finite precision digital machine is addressed. In all of its generality this task is an impossible one for arbitrary signals $f \in L^2(\mathbb{R})$ for the main reason that a digital machine is inherently constrained to operate on objects represented by a finite number of bits. Because of this, any signal to be digitally processed must first be representable by a finite number of bits. This has the practical consequence of limiting the types of analog signals that may be digitally processed to ones which are well represented by their quantized sample values on some finite interval, for example, $\{Qf(t_n)\}_{n=1}^N$, where $-\infty < a \leq t_n < t_{n+1} \leq b < \infty$ and Q is a quantization⁹ function. An analog function that is recoverable to a good approximation from a representation of this type is said to *admit* a digital representation. With respect to the computation of the Fourier transform of an analog signal there are two main issues:

- determination of a class (or classes) of analog signals that admit digital representations in *both* the time and frequency domains; and
- bounds on the approximation error.

Since it relates an analog signal to its sample values, the classical sampling theorem provides at least a starting point for the resolution of these two issues. In fact, using the classical sampling theorem and the PSF, one may write down an exact formula for the evaluation of the Fourier transform on a uniform sampling set in the frequency domain. Application of these two theorems requires that the class of signal be limited to those which are simultaneously bandlimited and periodizable.

⁹A quantization function $Q : L^2(\mathbb{R}) \mapsto L^2(\mathbb{R})$ is one that maps values of f to their closest member in a finite set of isolated points $\{q_n\}_{n=1}^J$ with the idea that $Qf \approx f$. This will be the case if the range of f is close to the range of $\{q_n\}$ and there are a sufficient number J of the quantization values q_n (i.e., the quantization is fine enough). This assumption is implicitly made in the following.

Thus, suppose that $f \in PW_\Omega \cap \mathcal{P}_T$ for finite $\Omega > 0$ and $T > 0$. With $f \in PW_\Omega$ the classical sampling theorem allows f to be written as

$$f = \Delta \sum_{n=-\infty}^{\infty} f(n\Delta) \tau_{n\Delta} d_{2\pi\Omega}$$

provided $2\Omega\Delta \leq 1$. Assume now that $2\Omega\Delta = 1$. Taking the Fourier transform of each side gives

$$\hat{f} = \Delta \sum_{n=-\infty}^{\infty} f(n\Delta) e^{-jn\Delta} \cdot 1_{(-\Omega, \Omega)}.$$

Because $\hat{f} = 0$ outside $(-\Omega, \Omega)$ it is sufficient to restrict attention to the finite set of frequencies $\{k2\Omega/N\}_{k=-N/2}^{N/2-1}$ where N is chosen to be even (for convenience) and $N \geq 2\Omega T$ (for inversion via the PSF). On this sampling set an exact formula for the Fourier transform is

$$\begin{aligned} \hat{f}\left(k \frac{2\Omega}{N}\right) &= \Delta \sum_{n=-\infty}^{\infty} f(n\Delta) e^{-jn2\pi nk2\Delta\Omega/N} \\ &= \Delta \sum_{n=-\infty}^{\infty} f(n\Delta) e^{-jn2\pi nk/N} \\ &= \Delta \sum_{m=-\infty}^{\infty} \sum_{l=mN}^{mN+N-1} f(l\Delta) e^{-j2\pi lk/N} \\ &= \Delta \sum_{m=-\infty}^{\infty} \sum_{l=0}^{N-1} f((l+mN)\Delta) e^{-j2\pi(lk/N+mk)} \\ &= \Delta \sum_{l=0}^{N-1} \left(\sum_{m=-\infty}^{\infty} f(l\Delta + mT) \right) e^{-j2\pi lk/N} \\ &= \Delta \sum_{l=0}^{N-1} \left[\left(\sum_{m=-\infty}^{\infty} \tau_{mT} f \right) (l\Delta) \right] e^{-j2\pi lk/N} \end{aligned}$$

for $k = -N/2, -N/2 + 1, \dots, N/2 - 1$. This computation shows that the value of the Fourier transform evaluated on the sampling set $\{k2\Omega/N\}$ is *exactly* the discrete Fourier transform of the T -periodized version of the function f evaluated on the sampling set $\{l\Delta\}_{l=0}^{N-1}$. In symbols,

$$\hat{f}\left(k \frac{2\Omega}{N}\right) = \text{DFT}(f^*),$$

where $f^* = \{(\tau_{mT} f)(l\Delta)\}_{l=0}^{N-1} \in \ell^2(\{0, 1, \dots, N-1\})$.

Thus, the exact computation of \hat{f} on the sampling set may be attained by the steps:

- (i) computation of f^* , and
- (ii) computation of the DFT of f^* .

Since the DFT is computable on a finite precision machine to a high degree of precision, computation (ii) is digitally tractable; however, computation (i) requires knowledge of the signal f at an infinite number of times and an infinite summation. In the case where the signal f is known analytically, an analytic expression for f^* may be possible to determine. More likely, an approximation to f^* is necessary.

A common and trivially applied approximation is

$$f_l^* \approx f(l\Delta - T/2), \quad l = 0, 1, \dots, N - 1.$$

For T large enough and for f with sufficient decay this approximation is a reasonable one.

Even for signals $f \in L^2(\mathbb{R})$ that are not bandlimited, approximations to the true Fourier transform may be made via a DFT computation; however, in this case good approximations require very small sampling intervals Δ so that in the limit $\Omega = 1/(2\Delta) \rightarrow \infty$.

Problems

2.1 Suppose $f, g \in L^2(\mathbb{R})$.

- (a) Write the convolution $(f * g)(t)$ in terms of the inner product $\langle \cdot, \cdot \rangle$ and translation operator τ_t .
- (b) Using the Parseval relation (2.11) and the expression previously derived show that $(f * g)^\wedge = \hat{f} \cdot \hat{g}$.

2.2 Show that a function $g \in L^1(\mathbb{R})$ has its Fourier transform \hat{g} bounded by $\|g\|_1$ and that \hat{g} is continuous.

2.3 Let $a \in \mathbb{R}$, $f \in L^2(\mathbb{R})$. Prove the following relations.

- (a) $(\tau_a f)^\wedge = e^{-a} \hat{f}$.
- (b) $(e_a f)^\wedge = \tau_a \hat{f}$.
- (c) $(D_a f)^\wedge = D_{a-1} \hat{f}$.
- (d) $(\mathcal{R}f)^\wedge = \mathcal{R}\hat{f}$.
- (e) $(\tilde{f})^\wedge = \overline{\hat{f}}$.

2.4 Let $f \in L^2(\mathbb{R})$; show that

- (a) if f is purely real then $\hat{f} = \tilde{\hat{f}}$;
- (b) if f is purely imaginary then $\hat{f} = -\tilde{\hat{f}}$.

Hint: Use Problem 2.3e.

2.5 Show that the Fourier transform of the signal

$$f(t) = e^{-st^2}, \quad s \in \mathbb{C}, \operatorname{Re}\{s\} > 0,$$

is

$$\hat{f}(\gamma) = e^{-(\pi\gamma)^2/s} \cdot \sqrt{\frac{\pi}{s}}.$$

Determine a function f such that $\hat{f} = f$.

2.6 Compute the operator norm of the operators mapping $L^2(\mathbb{R}) \mapsto L^2(\mathbb{R})$:

- (a) Translation τ_a ,
- (b) Modulation e_a ,
- (c) Dilation D_a ,
- (d) Involution $\tilde{\cdot}$, and
- (e) Fourier Transform \mathcal{F} (viz. Equation (2.1)).

- 2.7 Let $S \subseteq \mathcal{H}$ be a subspace of the Hilbert space \mathcal{H} and $P_S : \mathcal{H} \mapsto S$ be the orthogonal projection operator onto the subspace S . Using Fact 2.3 show that $\|P_S\| = 1$.

Hint: First, show that if T is self-adjoint then $\|T^2\| = \|T\|^2$.

- 2.8 Consider the Hilbert space $\ell^2(J)$, where $J = \{1, 2\}$. An operator $T : \ell^2(J) \mapsto \ell^2(J)$ may be written as

$$T = \begin{pmatrix} a & b \\ c & d \end{pmatrix},$$

where $a, b, c, d \in \mathbb{C}$.

- (a) Find conditions on a, b, c , and d such that
 - i. T is onto $\ell^2(J)$.
 - ii. $T \in \mathcal{B}(\ell^2(J), \ell^2(J))$.
 - iii. T is unitary.
 - iv. $T = T^*$.
 - (b) Compute $\|T\|$.
 - (c) Compute $\langle x, Tx \rangle$ where $x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \in \ell^2(J)$. Comment on its relationship with $\|T\|$ when T is self-adjoint.
- 2.9 This exercise is intended to give the reader a *feel* for the type of functions that are periodizable. Let T be a given positive constant.
- (a) Show that any compactly supported signal $f \in L^2(\mathbb{R})$ (i.e., a signal that is identically zero off some interval) is also in \mathcal{P}_T .
 - (b) Give an example of a signal that is in $L^2(\mathbb{R})$ but not in \mathcal{P}_T .
- Hint: Consider the function $1/t$ on some infinite interval.

- 2.10 Show that if $f \in \mathcal{P}_T$ then

$$\left\langle \sum \tau_{nT} f, e_{n/T} \right\rangle_{L^2(-T/2, T/2)} = \widehat{f}\left(\frac{n}{T}\right).$$

3

Signal Representation and Frames

A pervasive and useful idea in mathematics is that functions (signals) may be decomposed into elementary atomic functions. Suppose that $\{\phi_n\}$ is such a collection of building blocks or *atoms* from which functions may be constructed. Any such constructed function f has the form

$$f(t) = \sum c_n \phi_n(t) \quad (3.1)$$

for some constants c_n that may be chosen. By specifying both an underlying atomic set $\{\phi_n\}$ and an associated scalar sequence $\{c_n\}$, one may provide a description of the function f that has both practical and analytic value. This is true with the caveat that the atomic set is chosen with some care. In particular, the interpretation of the equality in Equation (3.1) is not straightforward for arbitrary atomic sets. In addition, there are deep and interesting convergence issues concerning the right-hand side of (3.1) when the atomic set has an infinite number of members. In a practical sense, it may be argued that choosing atomic sets that lead to fundamental analytical problems such as these are bad choices and should be avoided. Such problems may be circumvented by placing some modest requirements on the atomic set; namely, that it form a frame for a large enough space of interest.

This chapter describes a general theory for the discrete representation of analog signals using mathematical frames. The presentation starts with familiar cases in which the underlying atomic functions form orthonormal bases and progresses through Riesz bases to the general overcomplete case. Always, functions of interest are assumed to come from a general Hilbert space \mathcal{H} of interest.

3.1 Inner Product Representation (Atomic Decomposition)

The goal of an atomic decomposition is to provide a recipe for constructing a given function $f \in \mathcal{H}$ from a set of atomic functions $\{\phi_n\}$. Analysis is

restricted to recipes (linear combinations) of the form

$$f(t) = \sum c_n \phi_n(t),$$

where $\{c_n\}$ is a countable sequence in $\ell^2(\mathbb{Z})$ and the atoms satisfy some fundamental properties. For instance, a basic requirement is that the atoms span the space \mathcal{H} , that is, that every $f \in \mathcal{H}$ is essentially a linear combination of the atoms. More specifically, the atoms are required to form a frame for \mathcal{H} . Frames may be thought of as a generalization of an orthonormal basis and are a main topic of this chapter.

Given a function $f \in \mathcal{H}$ and a frame $\{\phi_n\}$ for \mathcal{H} , sequences $c = \{c_n(f)\}$ that satisfy Equation (3.1) may be generated through the application of the associated fundamental (linear) representation operator L on the signal f . In fact, it is shown in this chapter that there is a well-defined linear operator R^\dagger such that coefficients satisfying (3.1) are given by $c = R^\dagger L f$. The representation operator L is the main mathematical tool used for discretization and is defined as

$$\begin{aligned} L : \quad \mathcal{H} &\rightarrow \ell^2(\mathbb{Z}) \\ f &\mapsto \{\langle f, \phi_n \rangle\} \end{aligned} \tag{3.2}$$

for some $f \in \mathcal{H}$ of interest. In some contexts the operator L is sometimes called the Bessel operator. L is referred to more generally here as the *representation* operator (associated with $\{\phi_n\}$) or simply *representation*. If $\{\phi_n\}$ is a frame for \mathcal{H} then L is appropriately characterized as a “frame representation.”

3.2 Orthonormal Bases

This section reviews some standard facts of orthonormal bases (ONBs) in the context of the representation operator L where the atoms $\{\phi_n\} \subseteq \mathcal{H}$ are constrained to form an orthonormal basis for a Hilbert space \mathcal{H} of interest. An orthonormal basis is special not only because the representation of a given signal with respect to it consists of unique coefficients but also because these coefficients quantify directly the contribution that each basis function makes to the signal. More to the point, the fact that $\{\phi_n\}$ is an ONB for \mathcal{H} implies that $c = Lf$ is the *only* choice of coefficients that satisfies (3.1) for a given $f \in \mathcal{H}$.

To be precise, an orthonormal sequence of functions $\{\phi_n\} \subseteq \mathcal{H}$ is an *orthonormal basis* for \mathcal{H} if for every $f \in \mathcal{H}$ there is a unique sequence $\{c_n\} \in \mathbb{C}$ such that

$$f = \sum c_n \phi_n.$$

Note that the assumption that the sequence of functions is orthonormal to begin with is critical since there are many nonorthogonal sequences of

functions with unique expansion coefficients as in the preceding and yet do not form ONBs (viz. [AU93], [Ald96], and Section 3.3).

For an orthonormal basis of an infinite dimensional space \mathcal{H} , for example, $L^2(\mathbb{R})$, the expansion is explicitly given as

$$f(t) = \sum_{n=-\infty}^{\infty} c_n \phi_n(t),$$

for almost all t . In this case the equality is understood to mean

$$\|f - \lim_{N \rightarrow \infty} \sum_{n=-N}^N c_n \phi_n\|_{\mathcal{H}} = 0.$$

Moreover, the unique sequence is given by $c_n = \langle f, \phi_n \rangle$; that is, $c = Lf$.

3.2.1 Parseval and Plancherel

The Parseval and Plancherel relations are valid for general orthonormal bases. Let $f, g \in \mathcal{H}$ and $\{\phi_n\}$ be an orthonormal basis. $L(\mathcal{H})$ denotes the range of the L defined as

$$L(\mathcal{H}) \triangleq \{Lf : f \in \mathcal{H}\}.$$

Plancherel for an ONB of \mathcal{H}

$$\|f\|_{\mathcal{H}}^2 = \sum |\langle f, \phi_n \rangle|^2 = \|Lf\|_{L(\mathcal{H})}^2. \quad (3.3)$$

Parseval for an ONB of \mathcal{H}

$$\langle f, g \rangle = \sum \langle f, \phi_n \rangle \overline{\langle g, \phi_n \rangle} = \langle Lf, Lg \rangle_{L(\mathcal{H})}. \quad (3.4)$$

3.2.2 Reconstruction

Given an orthonormal basis representation Lf of a signal $f \in \mathcal{H}$, the recovery of f from Lf is conceptually a simple matter. If $\{\phi_n\}$ is an orthonormal basis for \mathcal{H} then for every $f \in \mathcal{H}$

$$f = L^* Lf = \sum \langle f, \phi_n \rangle \phi_n. \quad (3.5)$$

Consequently, the representation L associated with an orthonormal basis is a unitary operator; that is,

$$\{\phi_n\} \text{ an ONB for } \mathcal{H} \implies L^* L = I_{\mathcal{H}},$$

where $I_{\mathcal{H}}$ is the identity operator on \mathcal{H} . In particular, L is invertible and $L^{-1} = L^*$. In other words f may be recovered from its orthonormal representation Lf through the application of the operator L^* . This fact is formally stated as Theorem 3.1.

Theorem 3.1 (Reconstruction from ONB Representation) *If L is the frame representation associated with an orthonormal basis $\{\phi_n\}$ of \mathcal{H} then*

$$\forall f \in \mathcal{H}, \quad f = L^*(Lf).$$

Some well-known equivalent conditions for an orthonormal sequence to form an ONB are given in Theorem 3.2. Recall that an orthonormal sequence $\{\phi_n\}$ is one for which $\forall n, m, \langle \phi_m, \phi_n \rangle = \delta_{m,n}$.

Theorem 3.2 (Orthonormal Bases) *Let $\{\phi_n\}$ be an orthonormal sequence in \mathcal{H} . The following are equivalent.*

- a. $\{\phi_n\}$ is an orthonormal basis for \mathcal{H} .
- b. $\forall n, \langle f, \phi_n \rangle = 0 \implies f = 0$.
- c. $\overline{\text{span}\{\phi_n\}} = \mathcal{H}$.
- d. $\|f\|^2 = \sum |\langle f, \phi_n \rangle|^2$.
- d'. L is unitary.

It is interesting to note that items *b* and *c* of Theorem 3.2 may hold true for more general sequences $\{\phi_n\}$ that need not be orthonormal. An overcomplete frame is one possible example. Overcomplete frames are discussed in detail in Section 3.4.

Note that an orthonormal basis $\{\phi_n\}$ for a space \mathcal{H} is *minimal* in the sense that the removal of a single basis function ϕ_m will cause the reduced set $\{\phi_n\}_{n \neq m}$, for a fixed but arbitrary choice of $m \in \mathbb{Z}$, to fail to be a basis for \mathcal{H} . For the reduced set there will be some functions $f \in \mathcal{H}$ that cannot be constructed without ϕ_m (e.g., ϕ_m itself). Thus, an ONB for a space \mathcal{H} just spans \mathcal{H} *exactly*. The idea of exactness is discussed further in Section 3.4. This section concludes with some examples of orthonormal bases.

3.2.3 Examples

Example 3.3 (Critical Sampling in PW_Ω) *Consider the $\Omega > 0$ bandlimited space PW_Ω and let $\Delta = 1/(2\Omega)$. By Theorem 2.4 (via the Fourier transform) the sequence of uniformly translated (by multiples of Δ) sinc functions*

$$\left\{ \sqrt{\Delta} \tau_{n\Delta} d_{2\pi\Omega} \right\}$$

is an orthonormal basis for PW_Ω , where d is the Dirichlet kernel of Equation (2.12).

Note that this example is an equivalent statement of the Classical Sampling Theorem 2.6.

Example 3.4 (Standard Basis) An orthonormal basis for $\ell^2(\mathbb{Z})$ is the “natural basis” $\{\delta_{,n}\}_{n \in \mathbb{Z}}$. In this case

$$L = I_{\ell^2(\mathbb{Z})} = \begin{pmatrix} \ddots & 0 & 0 & 0 \\ & 1 & 0 & 0 \\ \dots & 0 & 1 & 0 & \dots \\ & 0 & 0 & 1 \\ 0 & 0 & 0 & \ddots \end{pmatrix}.$$

Example 3.5 (Unitary Equivalence) Let $\{\phi_n\}_{n=0}^{N-1}$ be an orthonormal basis for the space $\ell^2(\{0, 1, \dots, N-1\})$. If $U : \ell^2(\{0, 1, \dots, N-1\}) \mapsto \ell^2(\{0, 1, \dots, N-1\})$ is any unitary operator then the set $\{U\phi_n\}_{n=0}^{N-1}$ is also an orthonormal basis for $\ell^2(\{0, 1, \dots, N-1\})$.

As a simple corollary to this example it is easy to see that the columns (or rows) of an N -dimensional unitary operator U constitute an orthonormal basis for $\ell^2(\{0, 1, \dots, N-1\})$ since $U = UI_{\ell^2(\{0, 1, \dots, N-1\})}$ and the columns of $I_{\ell^2(\{0, 1, \dots, N-1\})}$ form the standard basis for $\ell^2(\{0, 1, \dots, N-1\})$.

Example 3.6 (DFT) The set of harmonic complex exponentials

$$\left\{ \left\{ \frac{1}{\sqrt{N}} e_{k/N}(n) \right\}_{n=0}^{N-1} \right\}_{k=0}^{N-1}$$

is an orthonormal basis for the space $\ell^2(\{0, 1, \dots, N-1\})$.

Example 3.7 (Planar Rotations) Let $\mathcal{H} = \ell^2(\{1, 2\}) \subseteq \mathbb{R}^2$ and $\theta \in [0, 2\pi)$. The elements

$$\left\{ \begin{pmatrix} \cos \theta \\ \sin \theta \end{pmatrix}, \begin{pmatrix} -\sin \theta \\ \cos \theta \end{pmatrix} \right\}$$

form an orthonormal basis for \mathbb{R}^2 .

3.3 Riesz Bases

A Riesz basis is a slight generalization of an orthonormal basis in the sense that the conditions that the basis elements be of unit norm and orthogonal to each other are relaxed; however, the exactness of the elements is not compromised. That is to say that, as is the case for an orthonormal basis, the removal of any one atom of a Riesz basis leaves a reduced set that fails to span the original space. In fact, the defining quality of a Riesz basis is that it be related to an orthonormal basis by a well-behaved linear mapping, namely, a topological isomorphism (invertible, in particular).

To state the definition formally, a basis $\{\phi_n\}$ is a *Riesz basis* for \mathcal{H} if there is a topological isomorphism $T : \mathcal{H} \mapsto \mathcal{H}$ such that $\phi_n = Tu_n$ for all n , where $\{u_n\}$ is an orthonormal basis for \mathcal{H} .

Because T is invertible there exists another basis $\{\psi_n\}$ for \mathcal{H} to which $\{\phi_n\}$ is *biorthogonal*; that is, there is a dual sequence $\{\psi_n\}$ such that

$$\overline{\text{span}\{\psi_n\}} = \mathcal{H} \quad \text{and} \quad \langle \phi_n, \psi_m \rangle = \delta_{m,n}.$$

In this case $\{\psi_n\}$ is the *dual* basis to $\{\phi_n\}$. Let T and $\{u_n\}$ be the mapping and ONB in the definition of the Riesz basis $\{\phi_n\}$. It is easy to show that there is a unique biorthogonal basis $\{\psi_n\}$ for the Riesz basis since

$$\begin{aligned} \delta_{m,n} = \langle u_n, u_m \rangle &= \langle T^{-1}Tu_n, u_m \rangle \\ &= \langle Tu_n, (T^{-1})^* u_m \rangle = \langle \phi_n, \psi_m \rangle \end{aligned}$$

with $\psi_m = (T^{-1})^* u_m$. It is obvious that a basis dual to a Riesz basis, being biorthogonal, is itself a Riesz basis.

Riesz bases are seen to be characterized as *exact* frames in the next section which discusses frames in general. This is followed with some specific examples of Riesz bases.

3.3.1 Reconstruction

Suppose $\{\phi_n\}$ and $\{\psi_n\}$ are biorthogonal Riesz bases for a Hilbert space \mathcal{H} . Let L_ϕ and L_ψ denote the representation operators with respect to the bases $\{\phi_n\}$ and $\{\psi_n\}$, respectively.

The main question of interest is given the representation $L_\phi f$ of a signal $f \in \mathcal{H}$, how can the original signal f be recovered. Since $\{\phi_n\}$ and $\{\psi_n\}$ are biorthogonal Riesz bases for \mathcal{H} there is a topological isomorphism T and an orthonormal basis $\{u_n\}$ for \mathcal{H} such that

- (i) $\phi_n = Tu_n$, and
- (ii) $\psi_n = (T^{-1})^* u_n$.

From these two facts, the relationship between the biorthogonal bases is computed as

$$\phi_n = Tu_n = TT^*\psi_n. \tag{3.6}$$

Because $\{u_n\}$ is an ONB for \mathcal{H} and T is a topological isomorphism on \mathcal{H} any signal $f \in \mathcal{H}$ may be written as

$$\begin{aligned} f = T(T^{-1}f) &= T \sum \langle T^{-1}f, u_n \rangle u_n \\ &= \sum \langle f, (T^{-1})^* u_n \rangle Tu_n \\ &= \sum \langle f, \psi_n \rangle \phi_n \\ &= L_\phi^* L_\psi f. \end{aligned}$$

Similarly,

$$\begin{aligned} f = (T^{-1})^* T^* f &= (T^{-1})^* \sum \langle T^* f, u_n \rangle u_n \\ &= \sum \langle f, Tu_n \rangle (T^{-1})^* u_n \\ &= \sum \langle f, \phi_n \rangle \psi_n \\ &= L_\psi^* L_\phi f. \end{aligned}$$

Combining these results and Equation (3.6) leads to the conclusion that

$$L_\psi^* L_\phi = L_\phi^* L_\psi = (TT^*)^{-1} L_\phi^* L_\phi = I_{\mathcal{H}}.$$

Thus, f may be recovered from its representation $L_\phi f$ with respect to $\{\phi_n\}$ by application of the operator $L_\psi^* = (TT^*)^{-1} L_\phi^*$.

The results of the preceding discussion are stated formally as Theorem 3.8.

Theorem 3.8 (Reconstruction from Riesz Representation) *If L is the frame representation and T is the topological isomorphism associated with a Riesz basis $\{\phi_n\}$ of \mathcal{H} then*

$$\forall f \in \mathcal{H}, \quad f = (TT^*)^{-1} L^* (Lf).$$

3.3.2 Examples

Example 3.9 (A Riesz Basis from an ONB) *Starting with an ONB $\{u_n\}$ for a Hilbert space \mathcal{H} , a Riesz basis for \mathcal{H} may be generated by filtering the ONB through an invertible linear filter with impulse response g . More specifically, the sequence*

$$\{u_n * g\}$$

*is a Riesz basis for \mathcal{H} provided there is an inverse filter h such that $u_n = (u_n * g) * h$. Such functions g are easily designed in the frequency domain by requiring $\widehat{g} > 0$ over all frequencies $\text{supp } \widehat{\mathcal{H}}$ that support \mathcal{H} . Then the inverse filter h is given simply as the multiplicative inverse $\widehat{h} = 1/\widehat{g}$ on $\text{supp } \widehat{\mathcal{H}}$ and zero elsewhere.*

In the case of Example 3.9 the dual (biorthogonal) basis to $\{u_n * g\}$ is readily computed as

$$\left\{ h * \tilde{h} * (u_n * g) \right\}.$$

To see this, write

$$\begin{aligned} \delta_{m,n} &= \langle u_m, u_n \rangle = \langle (u_m * g) * h, (u_n * g) * h \rangle \\ &= \langle \widehat{u}_m \widehat{g} \widehat{h}, \widehat{u}_n \widehat{g} \widehat{h} \rangle = \langle \widehat{u}_m \widehat{g}, \widehat{u}_n \widehat{g} | \widehat{h} |^2 \rangle \\ &= \langle (u_m * g), (u_n * g) * h * \tilde{h} \rangle. \end{aligned}$$

A more extensive discussion of the specification of Riesz bases from ONBs and reconstructions may be found in [AU94].

Example 3.10 (Kadec–Levinson Riesz Basis for PW_Ω) Let $\{t_n\}$ be an increasing sequence of real numbers such that

$$\sup_n |t_n - \frac{n}{2\Omega}| < \frac{1}{4} \left(\frac{1}{2\Omega} \right).$$

Then $\{\tau_{t_n} d_{2\pi\Omega}\}$ is a Riesz basis for PW_Ω .

Just as the properties of sets of harmonic complex exponentials are fundamental in regular sampling theory, the properties of sets of *nonharmonic* complex exponentials are fundamental in the theory of irregular sampling. Paley and Wiener first dealt with the question of when a set of nonharmonic complex exponentials $\{e_{-t_n}\}$ forms a basis for the space $L^2[-\Omega, \Omega]$. Levinson ([Lev40]) showed that the constant 0.25 is the smallest such constant that ensures the completeness of $\{e_{-t_n}\}$ in $L^2[-\Omega, \Omega]$; and later Kadec ([Kad64]) gave a direct proof that $\{e_{-t_n}\}$ is, in fact, a Riesz basis (viz. [Ben92, Theorem 34]). Duffin and Schaeffer ([DS52]) dealt with the question in even more generality via the development of frames. This is the subject of the next section.

3.4 General Frames

Frames were originally introduced in problems related to nonharmonic Fourier series and have become a staple in wavelet and irregular sampling theory. In this section basic theory regarding the decomposition and reconstruction of signals with respect to general frames is presented along with some newer results ([TB95]).

The theory of (Hilbert space) frames, first introduced by Duffin and Schaeffer ([DS52]), plays a major role in the general approach to atomic decomposition outlined here. As shown subsequently, the key elements of frame theory pivot around the fact that the associated “frame operator” S has many important properties; these include:

- (i) S has the factorization $S = L^*L$ (e.g., [DGM86], [Dau90]), where L is a “discretization” operator that associates with any $f \in \mathcal{H}$ the discrete representation Lf (L^* is its adjoint); and
- (ii) S is invertible (in fact, a topological isomorphism).

These two properties are key elements for atomic decomposition. To begin with, consider

$$\forall f \in \mathcal{H}, \quad f = S^{-1}L^*Lf. \tag{3.7}$$

In this equation it is seen that any signal f can be fully represented by the discrete vector Lf ; that is, the original signal f can be recovered from the discrete representation Lf . In other words f and Lf contain the same information.

3.4.1 Basic Frame Theory

The theory of frames is due to Duffin and Schaeffer ([DS52]; c.f. [Dau92], [DGM86], [HW89], [You80]). Definition 3.11 gives a formal definition of a Hilbert space frame and Theorem 3.13 states some of the fundamental properties of frames.

As a concept, frames provide an intermediate ground between the two related notions of *completeness* in a space and an orthonormal *basis* for a space. Recall that a set of functions $\{\phi_n\}$ is complete in a Hilbert space \mathcal{H} if the closure of their span is the whole space \mathcal{H} ; that is, $\overline{\text{span}\{\phi_n\}} = \mathcal{H}$. It is shown that the statements that a set $\{\phi_n\}$ is

- (a) complete in \mathcal{H} ,
- (b) a frame for \mathcal{H} ,
- (c) a Riesz basis for \mathcal{H} , and
- (d) an orthonormal basis for \mathcal{H}

are progressively stronger. In other words $d \implies c \implies b \implies a$.

With any set of functions $\{\phi_n\}$ one may associate the operator $S = S_\phi$ defined as

$$Sf \stackrel{\Delta}{=} \sum \langle f, \phi_n \rangle \phi_n.$$

The frame property can be equivalently characterized in terms of this operator S and, consequently, S has been called the *frame operator*. Specifically the set $\{\phi_n\}$ is a frame for \mathcal{H} if and only if there are constants A, B ,

$$\forall f \in \mathcal{H}, \quad A\|f\|^2 \leq \langle f, Sf \rangle \leq B\|f\|^2$$

such that $0 < A \leq B < \infty$. Note that the frame operator has many “nice” properties including linearity, continuity, and invertibility. Definition 3.11 gives the formal definition of a Hilbert space frame, its associated frame operator, and the notions of tightness and exactness.

Definition 3.11 (a) A sequence $\{\phi_n\} \subseteq \mathcal{H}$ is a frame for \mathcal{H} if there exist frame bounds $A, B > 0$ such that

$$\forall f \in \mathcal{H}, \quad A\|f\|^2 \leq \sum |\langle f, \phi_n \rangle|^2 \leq B\|f\|^2, \quad (3.8)$$

where summation is over \mathbb{Z} .

(b) The frame operator of the frame $\{\phi_n\}$ is the function $S : \mathcal{H} \mapsto \mathcal{H}$ defined as $Sf = \sum \langle f, \phi_n \rangle \phi_n$.

(c) A frame is said to be tight if $A = B$.

(d) A frame for \mathcal{H} is said to be exact if the set determined by the removal of any one of its elements fails to be a frame for \mathcal{H} .

Some examples of frames are presented in the following. From these examples it is seen that tightness and exactness are independent properties.

Tight and Exact Frame

Clearly any orthonormal basis $\{u_n\} \subseteq \mathcal{H}$ is a tight exact frame with bounds $A = B = 1$ since by Parseval's equality for orthonormal bases $\sum |\langle f, u_n \rangle|^2 = \|f\|^2$. It is exact since the elements are orthogonal and the removal of any one element will cause the reduced set to fail to be dense in \mathcal{H} .

Tight and Nonexact Frame

The union of any finite number $N > 2$ of orthonormal bases in \mathcal{H} yields a tight nonexact frame for \mathcal{H} with frame bounds $A = B = N$. To see this, for each $m = 1, 2, \dots, N$ let $\{u_{m,n}\}_n$ be an orthonormal basis for \mathcal{H} . Then by Parseval's equality for orthonormal bases

$$\sum_{m=1}^N \sum_n |\langle f, u_{m,n} \rangle|^2 = \sum_{m=1}^N \|f\|^2 = N\|f\|^2.$$

Since each basis $\{u_{m,n}\}_n$ is dense in \mathcal{H} the removal of any one element from

$$\bigcup_{m=1}^N \{u_{m,n}\}$$

will result in a collection of vectors that contains at least one orthonormal basis for \mathcal{H} . Therefore it is dense in \mathcal{H} .

For example $\{u_1, u_1, u_2, u_2, u_3, u_3, \dots\}$ is a tight nonexact frame with $A = B = 2$.

Nontight and Exact Frame

A nontight and exact frame may be generated from an ONB $\{u_n\}$ as

$$\{a_n u_n\},$$

where $\{a_n\}$ is a sequence of scalars satisfying $0 < A \leq a_n^2 \leq B < \infty$ and there is a pair (m, n) such that $a_m \neq a_n$. Clearly then the set $\{a_n u_n\}$ is exact and

$$\sum_n |\langle f, a_n u_n \rangle|^2 = \sum_n a_n^2 |\langle f, u_n \rangle|^2$$

$$\begin{aligned} &\leq \sum_n \left(\sup_k a_k^2 \right) |\langle f, u_n \rangle|^2 \\ &\leq B \sum_n |\langle f, u_n \rangle|^2 = B \|f\|^2. \end{aligned}$$

The lower bound is similar.

For example $\{\sqrt{2}u_1, u_2, u_3, u_4, \dots\}$ is a nontight exact frame with $A = 1$ and $B = 2$.

Nontight and Nonexact Frame

There are many ways to generate nontight nonexact frames, for example, the union of a orthonormal basis with a scaled orthonormal basis. One example is

$$\{u_1, 2u_1, u_2, 2u_2, u_3, 2u_3, \dots\}.$$

The theorem that follows exhibits the close relationship between exact frames and orthonormal bases. More precisely it states that exact frames are Riesz bases. Riesz bases are by definition related to orthonormal bases by a topological isomorphism. See [You80, p. 188] for a proof of the equivalence of parts (a) and (b).

Theorem 3.12 *Let \mathcal{H} be an arbitrary Hilbert space and $\{\phi_n\}$ be a sequence of elements in \mathcal{H} . The following are equivalent.*

- (a) $\{\phi_n\}$ is an exact frame for \mathcal{H} .
- (b) $\{\phi_n\}$ is a Riesz basis for \mathcal{H} .
- (c) $\{\phi_n\}$ is a bounded unconditional basis for \mathcal{H} .

The frame representation operator is the same as the representation operator L in the case where the underlying atomic functions form a frame for \mathcal{H} . Properties associated with such an L are discussed in this section.

The following theorem states some of the fundamental properties of frames. Its proof is left as an exercise for the reader.

Theorem 3.13 (a) *If $\{\phi_n\} \subseteq \mathcal{H}$ is a frame with frame bounds A, B , then S is a topological isomorphism with inverse S^{-1} , $\{S^{-1}\phi_n\}$ is a frame with frame bounds B^{-1} and A^{-1} , and*

$$\forall f \in \mathcal{H}, \quad f = \sum \langle f, S^{-1}\phi_n \rangle \phi_n = \sum \langle f, \phi_n \rangle S^{-1}\phi_n \quad (3.9)$$

in \mathcal{H} .

(b) *If $\{\phi_n\} \subseteq \mathcal{H}$, let $L : \mathcal{H} \mapsto \ell^2(\mathbb{Z})$ be defined as $Lf = \{\langle f, \phi_n \rangle\}$ (cf. (3.2)). If $\{\phi_n\}$ is a frame then $S = L^*L$, where L^* is the adjoint of L .*

Since the frame operator S may be factored ([DGM86], [Dau90]) as L^*L an immediate consequence is that

$$\langle f, Sf \rangle_{\mathcal{H}} = \langle f, L^*Lf \rangle_{\mathcal{H}} = \langle Lf, Lf \rangle_{L(\mathcal{H})} = \|Lf\|_{L(\mathcal{H})}^2.$$

Since $\{\phi_n\}$ is a frame with frame bounds A and B this implies that

$$A\|f\|_{\mathcal{H}}^2 \leq \|Lf\|_{L(\mathcal{H})}^2 \leq B\|f\|_{\mathcal{H}}^2.$$

Thus,

$$\|L\| \leq B^{1/2} \quad \text{and} \quad \|L^{-1}\| \leq A^{-1/2},$$

where L^{-1} is defined on the range $L(\mathcal{H})$, (viz. Theorem 3.14).

It is clear that if A and B are frame bounds for a frame $\{\phi_n\}$ then any other pair A_1 and B_1 such that $0 < A_1 < A$ and $B < B_1 < \infty$ are also valid frame bounds for $\{\phi_n\}$. It is of interest to know the smallest upper bound and the largest lower bound that serve as frame bounds for a frame. This motivates the notion of the *best* frame bounds. Given a frame $\{\phi_n\}$ for a Hilbert space \mathcal{H} with frame operator S , the *best bounds* A and B are

$$A = \inf_{f \in \mathcal{H}} \frac{\langle f, Sf \rangle}{\|f\|^2},$$

$$B = \sup_{f \in \mathcal{H}} \frac{\langle f, Sf \rangle}{\|f\|^2}.$$

Since $\|Lf\|_{L(\mathcal{H})}^2 = \langle f, Sf \rangle_{\mathcal{H}}$ it follows that the best bounds A, B are also $A = \|L^{-1}\|^{-2}$ and $B = \|L\|^2$.

3.4.2 Frame Representation

Consider the operator L of Theorem 3.13 and its adjoint. The theorem asserts that L and its adjoint L^* are factors of the frame operator S . Explicitly, the operators $L : \mathcal{H} \mapsto \ell^2(\mathbb{Z})$ and $L^* : \ell^2(\mathbb{Z}) \mapsto \mathcal{H}$ are

$$Lf = \{\langle f, \phi_n \rangle\} \tag{3.10}$$

and

$$L^*c = \sum c_n \phi_n, \tag{3.11}$$

where $f \in \mathcal{H}$ and $c \in \ell^2(\mathbb{Z})$. It is easily verified that $S = L^*L$ since for all $f \in \mathcal{H}$,

$$Sf = \sum \langle f, \phi_n \rangle \phi_n = L^*Lf.$$

Figure 3.1 depicts the mappings L and its adjoint L^* . If $\{\phi_n\}$ is a frame for \mathcal{H} then the mapping defined in (3.2) is called the *frame representation* or *frame discretization* operator. The frame representation operator L plays a

central role in Theorem 3.13. Part (a) of the theorem describes one method to recover a signal $f \in \mathcal{H}$ from its frame representation $Lf \in \ell^2(\mathbb{Z})$. In part (b), the theorem indicates that the frame operator S has factors L and L^* . In addition, Theorem 3.14 states that the frame representation operator L has an inverse when considered on the range $L(\mathcal{H})$. These facts form the basis for the iterative reconstruction scheme given in Algorithm 3.24 and, in turn, the notion of the *frame correlation operator* discussed in Section 3.4.7.

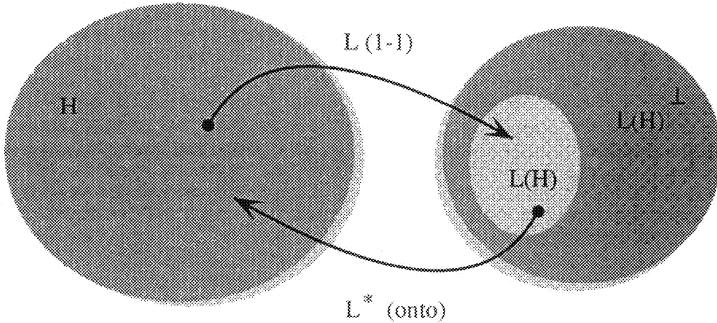


FIGURE 3.1. The mappings L and L^* .

A characterization of frame representation operators is given in the following theorem.

Theorem 3.14 ([Ben93, Theorem 3.6]) *The sequence $\{\phi_n\}$ is a frame for \mathcal{H} if and only if the mapping L given in Equation (3.2) is a well-defined topological isomorphism onto a closed subspace of $\ell^2(\mathbb{Z})$.*

Theorem 3.14 has certain significant consequences worthy of mention. Namely, if L is a frame representation operator then

- (a) L is injective (one-to-one),
- (b) $L(\mathcal{H})$ is closed, and
- (c) L^* is surjective (onto).

To see the injectivity of the map L suppose $\{\phi_n\}$ is a frame for \mathcal{H} with bounds A and B and that $Lf_1 = Lf_2$. Then

$$0 = \|Lf_1 - Lf_2\|^2 = \|L(f_1 - f_2)\|^2 \geq A\|f_1 - f_2\|^2,$$

which implies that $f_1 = f_2$. Thus, L is injective. Because L is an injective bounded linear operator whose range $L(\mathcal{H})$ is closed, then L^* is onto.

3.4.3 Frame Correlation and Pseudo-Inverse

The *frame correlation* operator R is introduced in this section. It is shown that the frame correlation R and its pseudo-inverse R^\dagger play a crucial role in the process of digital reconstruction from frame representation. That relationship is fully developed in this section.

Definition 3.15 Let $\{\phi_n\}$ be a frame for \mathcal{H} with frame representation operator L . The **frame correlation** operator is defined as $R \triangleq LL^*$.

The frame correlation matrix R shares many properties with the frame operator S . For instance, they are both nonnegative self-adjoint operators that map bijectively onto their range. The crucial differences, however, are that

- (a) the range and domain of R are contained in $\ell^2(\mathbb{Z})$, whereas the range of S is \mathcal{H} , and
- (b) the range of R need not be all of $\ell^2(\mathbb{Z})$.

These two differences directly relate to the issues of

- (a) digital implementability and
- (b) representation noise robustness,

respectively. As discussed previously the fact that R operates on countable sequences (i.e., digital signals) immediately suggests that it is possible to implement R on a digital machine. This is not directly possible for S . Robustness to noise in the representation is directly related to the size of the kernel of R . This issue is addressed in Section 7.1.8.

The remainder of this section presents several propositions that help illuminate the properties of the frame correlation R . The first, Proposition 3.16, exposes the matrix representation of the frame correlation R (of a frame $\{\phi_n\}$ for a Hilbert space \mathcal{H}) as the Gram matrix associated with the sequence of elements $\{\phi_n\} \subseteq \mathcal{H}$. The second, Proposition 3.17, compiles a list of useful general properties of a frame correlation R . Finally, Theorem 3.18 establishes the ramifications for R if the underlying frame is exact.

Proposition 3.16 Given a frame $\{\phi_n\}$ for the Hilbert space \mathcal{H} , the frame correlation matrix R has the matrix representation

$$R \triangleq (R_{m,n}) = (\langle \phi_m, \phi_n \rangle).$$

Proof: Take any $c \in \ell^2(\mathbb{Z})$. Then one can write

$$Rc = LL^*c = \{\langle L^*c, \phi_n \rangle\} = \left\{ \sum_m c_m \langle \phi_m, \phi_n \rangle \right\}$$

so that

$$(Rc)_n = (LL^*c)_n = \sum_m c_m \langle \phi_m, \phi_n \rangle, \quad (3.12)$$

and

$$\begin{pmatrix} \vdots \\ (Rc)_{-1} \\ (Rc)_0 \\ (Rc)_1 \\ \vdots \end{pmatrix} = \begin{pmatrix} & & & \vdots \\ \langle \phi_{-1}, \phi_{-1} \rangle & \langle \phi_0, \phi_{-1} \rangle & \langle \phi_1, \phi_{-1} \rangle & \\ \langle \phi_{-1}, \phi_0 \rangle & \langle \phi_0, \phi_0 \rangle & \langle \phi_1, \phi_0 \rangle & \\ \langle \phi_{-1}, \phi_0 \rangle & \langle \phi_0, \phi_1 \rangle & \langle \phi_1, \phi_1 \rangle & \\ & & \vdots & \end{pmatrix} \begin{pmatrix} \vdots \\ c_{-1} \\ c_0 \\ c_1 \\ \vdots \end{pmatrix}. \quad (3.13)$$

■

Proposition 3.17 Suppose $\{\phi_n\}$ is a frame for the Hilbert space \mathcal{H} with frame representation operator L , correlation R , and bounds A and B .

(a) If \mathcal{H} is infinite-dimensional then L is not compact, and thus, R is not compact.

(b) For each row m of R , $\lim_{n \rightarrow \infty} |R_{m,n}| = 0$.

For each column n of R , $\lim_{m \rightarrow \infty} |R_{m,n}| = 0$.

(c) If the set $\{\phi_n\}$ is an orthonormal basis for \mathcal{H} then the frame correlation operator is the identity.

(d) $\ker R = L(\mathcal{H})^\perp$.

(e) R maps $L(\mathcal{H})$ bijectively to itself.

(f) $R = P_{L(\mathcal{H})}R = RP_{L(\mathcal{H})}$

(g) R is self-adjoint.

(h) $R \geq 0$.

Proof: (a) To prove the noncompactness of L a normalized sequence $\{f_n\} \subseteq L^2(\mathbb{R})$ shall be chosen such that $\{Lf_n\}$ has no converging subsequence. Let $f_n = 1_{[n, n+1]}$ so that $\|f_n\| = 1$. For such a sequence one has

$$\|Lf_n - Lf_m\| = \|L(f_n - f_m)\| \geq A^{1/2} \|f_n - f_m\|,$$

where the equality holds by the linearity of L and the inequality holds by the frame condition applied to $f_n - f_m$. For our choice of $\{f_n\}$ it is easy to see that $\|f_n - f_m\| = 2(1 - \delta_{m,n})$. Thus one has that

$$\forall n \neq m, \quad \|Lf_n - Lf_m\| \geq 2A^{1/2}.$$

Since $\ell^2(\mathbb{Z})$ is complete one may conclude that $\{Lf_n\}$ has no converging subsequences. Hence L is not compact.

To show R is not compact note that by part (e), there is a well-defined inverse R^{-1} of R on $L(\mathcal{H})$. With $\{f_n\}$ as in part (a) define

$$c_n \triangleq R^{-1}Lf_n / \|R^{-1}Lf_n\| = R^{-1}L(f_n/a_n)$$

where the normalization is well-defined since

$$a_n \stackrel{\Delta}{=} \|R^{-1}L f_n\| = \|L_\psi f_n\| \geq B^{-1/2} > 0.$$

Also note that $a_n \leq A^{-1/2}$ or $a_n^{-2} \geq A$. As in part (a) compute

$$\|f_n/a_n - f_m/a_m\| = (a_n^{-2} + a_m^{-2})(1 - \delta_{m,n}) \geq 2A(1 - \delta_{m,n}).$$

For the sequence $\{c_n\} \subseteq \ell^2(\mathbb{Z})$ then

$$\forall n \neq m, \quad \|Rc_n - Rc_m\| = \|L(f_n/a_n - f_m/a_m)\| \geq 2A^{3/2},$$

and one concludes that $\{Rc_n\}$ has no converging subsequences so that R is not compact.

(b) Since $\{\phi_n\}$ is a frame for \mathcal{H} then one must have in particular that

$$\forall m \quad \sum_n |\langle \phi_m, \phi_n \rangle|^2 \leq B \|\phi_m\|^2.$$

For the sum to converge it is necessary that $|\langle \phi_m, \phi_n \rangle| \rightarrow 0$ as $m \rightarrow \infty$.

(c) Follows from Proposition 3.16 since $\langle \phi_m, \phi_n \rangle = \delta_{m,n}$.

(d) i/ $\ker R \subseteq L(\mathcal{H})^\perp$

Let $c_0 \in \ker R$ so that $Rc_0 = 0$. It is shown that for all $f \in \mathcal{H}$ that $\langle c_0, Lf \rangle = 0$. Since L^* is onto then for all $f \in \mathcal{H}$ there is a $c \in L(\mathcal{H})$ so that $f = L^*c$. Thus,

$$\langle c_0, Lf \rangle = \langle L^*c_0, f \rangle = \langle L^*c_0, L^*c \rangle = \langle Rc_0, c \rangle = 0.$$

ii/ $\ker R \supseteq L(\mathcal{H})^\perp$

Let $c_\perp \in L(\mathcal{H})^\perp$. For any $c \in \ell^2(\mathbb{Z})$

$$\begin{aligned} \langle Rc_\perp, c \rangle &= \langle c_\perp, Rc \rangle \\ &= \langle c_\perp, L(L^*c) \rangle = 0. \end{aligned}$$

Since $\langle Rc_\perp, c \rangle = 0$ for all $c \in \ell^2(\mathbb{Z})$ then $Rc_\perp = 0$.

(e) Since L is a linear injective map, one need only demonstrate RL is an injective map from $L(\mathcal{H})$ to $L(\mathcal{H})$ to prove that R is injective on $L(\mathcal{H})$. Write $RL = LL^*L = LS$. Since both L and S are injective then $LS = RL$ is also. Thus R is 1-1. Since R is self-adjoint and 1-1 then it must also be onto.

(f) Let $c \in \ell^2(\mathbb{Z})$. Clearly, $Rc = LL^*c \in L(\mathcal{H})$. Thus, $R = P_{L(\mathcal{H})}R$ and by taking the adjoint $R = RP_{L(\mathcal{H})}$.

(g) $R^* = (LL^*)^* = LL^* = R$.

(h) Since L^* is surjective for all $f \in L^2(\mathbb{R})$ there is a $c \in L(\mathcal{H})$ such that $f = L^*c$. Thus, $0 \leq \|f\|^2 = \langle L^*c, L^*c \rangle = \langle c, LL^*c \rangle = \langle c, Rc \rangle$. ■

It has been seen in Proposition 3.17(e) that R maps $L(\mathcal{H})$ bijectively to itself. If $L(\mathcal{H})$ is all of $\ell^2(\mathbb{Z})$ (L is onto) then R is a topological isomorphism

on $\ell^2(\mathbb{Z})$. This can only happen if the underlying frame is a Riesz basis, that is, an exact frame. This and other equivalences are the content of Theorem 3.18.

Theorem 3.18 (Riesz Equivalences) *Let $\{\phi_n\}$ be a frame for \mathcal{H} with frame representation operator L , frame correlation R , and bounds A and B . The following are equivalent.*

- (a) L is onto $\ell^2(\mathbb{Z})$.
- (b) L^* is one-to-one.
- (c) R is a topological isomorphism on $\ell^2(\mathbb{Z})$.
- (d) $R > 0$.
- (e) $\{\phi_n\}$ is a Riesz–Fischer sequence.
- (f) $\{\phi_n\}$ is a Riesz basis.

Proof:

$$((a) \implies (b))$$

From elementary operator theory,

$$\ker L^* = L(\mathcal{H})^\perp = (\ell^2(\mathbb{Z}))^\perp = \{0\}$$

so that L^* (a linear operator) is one-to-one.

$$((b) \implies (c))$$

Clearly $R \in \mathcal{B}(\mathcal{H}, \mathcal{H})$ since $R = LL^*$ and $L \in \mathcal{B}(\mathcal{H}, \mathcal{H})$. R is one-to-one because L is one-to-one (always) and by assumption L^* is one-to-one so that the composition LL^* is also one-to-one. $R^* = R$ is onto since $R(\mathcal{H}) = L(\mathcal{H})$ is closed (Theorem 3.14(b)). Thus R is bijective on $\ell^2(\mathbb{Z})$ and there exists an inverse R^{-1} on all of $\ell^2(\mathbb{Z})$. Moreover, $\|R^{-1}\| \leq A^{-1}$ by dual frame arguments. Thus, R is a topological isomorphism on $\ell^2(\mathbb{Z})$.

$$((c) \implies (a))$$

Suppose R is a topological isomorphism and L is not onto $\ell^2(\mathbb{Z})$. But $R(\mathcal{H}) = L(\mathcal{H}) \neq \ell^2(\mathbb{Z})$ which means that R cannot be onto. This contradicts the assumption that R is a topological isomorphism.

$$((b) \implies (d))$$

Write

$$\langle c, Rc \rangle = \langle c, LL^*c \rangle = \|L^*c\|^2 \geq 0.$$

Since L^* is one-to-one then $\langle c, Rc \rangle = 0$ if and only if $c = 0$. Thus, $R > 0$.

$$((d) \implies (b))$$

$R > 0$ means that for all nonzero $c \in \ell^2(\mathbb{Z})$

$$0 < \langle c, Rc \rangle = \|L^*c\|^2.$$

If $L^*c_1 = L^*c_2$ then

$$\langle c_1 - c_2, R(c_1 - c_2) \rangle = \|L^*c_1 - L^*c_2\| = 0$$

and it is concluded that $c_1 = c_2$. Thus, L^* is one-to-one.

((a) \iff (e))

By definition $\{\phi_n\}$ is a Riesz–Fischer sequence means that for all $c \in \ell^2(\mathbb{Z})$ there is an $f \in \mathcal{H}$ such that $c = \{\langle f, \phi_n \rangle\}$. With L the frame representation $Lf = \{\langle f, \phi_n \rangle\}$, this translates equivalently to the statement that L is onto $\ell^2(\mathbb{Z})$.

((c) \implies (f))

Suppose R is a topological isomorphism on $\ell^2(\mathbb{Z})$. Recall that $Lf = \{\langle f, \phi_n \rangle\}$. Fix an integer N and define a new operator L_N as $L_Nf = \{\langle f, \phi_n \rangle\}_{n \neq N}$. It is shown that the removal of one element from the set $\{\phi_n\}$ will fail to be a frame by constructing a nonzero $f \in \mathcal{H}$ so that $L_Nf = 0$. Let $\{u_n\}$ be the standard orthonormal basis for $\ell^2(\mathbb{Z})$; that is, $u_n = \{\delta_{m,n}\}$. Since R is a topological isomorphism on $\ell^2(\mathbb{Z})$ then R has an inverse R^{-1} on $\ell^2(\mathbb{Z})$. One may then pick

$$f = L^*R^{-1}e_N$$

so that for this choice of f

$$Lf = L(L^*R^{-1}e_N) = e_N$$

and hence $L_Nf = 0$.

((f) \implies (c))

By definition $\{\phi_n\}$ is an exact frame means there exists a topological isomorphism T and an orthonormal basis $\{u_n\}$ for \mathcal{H} so that for all n , $\phi_n = Tu_n$. Let L_e be the frame representation associated with the orthonormal basis $\{u_n\}$. Since $\{u_n\}$ is an orthonormal basis then L_e is a topological isomorphism from \mathcal{H} onto $\ell^2(\mathbb{Z})$. Similarly L_e^* is a topological isomorphism from $\ell^2(\mathbb{Z})$ onto \mathcal{H} . Noting that

$$L_e T^* f = \{\langle T^* f, u_n \rangle\} = \{\langle f, Tu_n \rangle\} = \{\langle f, \phi_n \rangle\} = Lf,$$

it is concluded that $L = L_e T^*$ and $R = LL^* = L_e T^* T L_e^*$. Thus, since each factor of R is a topological isomorphism, R is a topological isomorphism on $\ell^2(\mathbb{Z})$. ■

3.4.4 Pseudo-Inverse

Proposition 3.17(e) implies that R has an inverse on $L(\mathcal{H})$. This inverse is denoted R^{-1} and

$$\forall c \in L(\mathcal{H}) \quad c = R^{-1}Rc = RR^{-1}c.$$

To extend the inverse to all of $\ell^2(\mathbb{Z})$ a *pseudo-inverse* of R may be defined as

$$R^\dagger \triangleq R^{-1}P_{L(\mathcal{H})}, \quad (3.14)$$

where $P_{L(\mathcal{H})}$ is the orthogonal projection operator onto the image of L . Definition 3.19 gives a definition of pseudo-inverse. There are several equivalent definitions of the pseudo-inverse with interesting interpretations ([Gro77]).

Proposition 3.17(f) can be used to demonstrate that as defined in Equation (3.14) R^\dagger is in fact a bona fide pseudo-inverse. Using Proposition 3.17(f) it is seen that for all $c \in \ell^2(\mathbb{Z})$

$$R^\dagger R c = R^{-1} P_{L(\mathcal{H})} R c = R^{-1} R P_{L(\mathcal{H})} c = P_{L(\mathcal{H})} c$$

and similarly

$$R R^\dagger c = R R^{-1} P_{L(\mathcal{H})} c = P_{L(\mathcal{H})} c.$$

It may be concluded that

$$R^\dagger R = R R^\dagger = P_{L(\mathcal{H})}. \quad (3.15)$$

This observation makes the verification of the defining conditions of a pseudo-inverse in Definition 3.19 trivial. Thus, Equation (3.14) does indeed give the pseudo-inverse of R . In Section 7.1.7 an iterative method for the construction of the pseudo-inverse is presented.

Definition 3.19 ([Gro77, Definition (P)]) *If $T \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$ has a closed range, then T^\dagger is the unique operator in $\mathcal{B}(\mathcal{H}_2, \mathcal{H}_1)$ satisfying*

- (1) TT^\dagger is self-adjoint,
- (2) $T^\dagger T$ is self-adjoint,
- (3) $TT^\dagger T = T$, and
- (4) $T^\dagger TT^\dagger = T^\dagger$.

One of the basic properties of the pseudo-inverse is that it provides the minimal norm solution to an (overcomplete) system of linear equations. This fact is stated formally in the following.

Fact 3.20 *Let \mathcal{H}_1 and \mathcal{H}_2 be two Hilbert spaces and suppose that the operator $A \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$ has a closed range. The mapping $A^\dagger : \mathcal{H}_2 \mapsto \mathcal{H}_1$ defined by*

$$A^\dagger x = \underset{y \in \mathcal{H}_2}{\operatorname{argmin}} \|y - Ax\|$$

is the pseudo-inverse (or generalized inverse) of A .

Pseudo-inverses and frames have been connected in a fundamental way in [Str91], [Teo93], [TB93], and [TB95]. The next subsection establishes the role that the pseudo-inverse plays in determining the best frame bounds.

3.4.5 Best Frame Bounds

There are various numerical methods for obtaining the best frame bounds for a given frame. The different methods are all related in that they seek to determine aspects of the eigenstructure of the associated frame correlation operator. One such method which is amenable to digital implementation is discussed in this section.

The following theorem shows that the best frame bounds of a frame are directly related to the operator norms of the frame correlation R and its pseudo-inverse R^\dagger .

Theorem 3.21 (Best Bounds) *Let $\{\phi_n\}$ be a frame for a Hilbert Space \mathcal{H} with best frame bounds A and B and frame correlation R . Then the frame correlation R is related to the frame bounds A and B as*

$$(a) A = \|R^\dagger\|^{-1}$$

$$(b) B = \|R\|.$$

Proof: (a) Since $\{\phi_n\}$ is a frame for \mathcal{H} then L^* is surjective. Hence, for all $f \in \mathcal{H}$ there is a $c \in L(\mathcal{H})$ so that $f = L^*c$ and one may write for f not zero a.e.

$$\frac{\langle f, Sf \rangle}{\|f\|^2} = \frac{\langle L^*c, SL^*c \rangle}{\langle L^*c, L^*c \rangle} = \frac{\langle L^*c, L^*LL^*c \rangle}{\langle c, LL^*c \rangle} = \frac{\langle c, R^2c \rangle}{\langle c, Rc \rangle} \quad (3.16)$$

for c off the $\ker L^* = L(\mathcal{H})^\perp$; that is, $c \in L(\mathcal{H})$. The surjectivity of L^* then implies the best lower frame bound A is

$$A \triangleq \inf_{f \in \mathcal{H}} \frac{\langle f, Sf \rangle}{\|f\|^2} = \inf_{c \in L(\mathcal{H})} \frac{\langle c, R^2c \rangle}{\langle c, Rc \rangle}.$$

Since R^\dagger is onto $L(\mathcal{H})$ for any $c \in L(\mathcal{H})$ there is a $c_0 \in L(\mathcal{H})$ so that $c = R^\dagger c_0$. With this substitution one has

$$\frac{\langle c, R^2c \rangle}{\langle c, Rc \rangle} = \frac{\langle R^\dagger c_0, R^2 R^\dagger c_0 \rangle}{\langle R^\dagger c_0, R R^\dagger c_0 \rangle} = \frac{\langle c_0, c_0 \rangle}{\langle R^\dagger c_0, c_0 \rangle}.$$

Thus,

$$\begin{aligned} A &= \inf_{c_0 \in L(\mathcal{H})} \frac{\langle c_0, c_0 \rangle}{\langle R^\dagger c_0, c_0 \rangle} &= \left(\sup_{c_0 \in L(\mathcal{H})} \frac{\langle R^\dagger c_0, c_0 \rangle}{\langle c_0, c_0 \rangle} \right)^{-1} \\ &= \left(\sup_{c_0 \in L(\mathcal{H})} \frac{|\langle R^\dagger c_0, c_0 \rangle|}{\langle c_0, c_0 \rangle} \right)^{-1}, \end{aligned}$$

where the last equality follows since R^\dagger is a nonnegative operator. Furthermore,

$$\begin{aligned} A &= \left(\sup_{c_0 \in L(\mathcal{H}) \oplus \ker R^\dagger} \frac{|\langle R^\dagger c_0, c_0 \rangle|}{\langle c_0, c_0 \rangle} \right)^{-1} \\ &= \left(\sup_{c_0 \in \ell^2(\mathbb{Z})} \frac{|\langle R^\dagger c_0, c_0 \rangle|}{\langle c_0, c_0 \rangle} \right)^{-1} = \|R^\dagger\|^{-1}, \end{aligned}$$

where one has used the fact that $\ker R^\dagger = L(\mathcal{H})^\perp$. The last equality is a consequence of Fact 2.2. ■

(b) The upper bound can be proven by the dual frame $\{S^{-1}\phi_n\}$ with best bounds A', B' , and frame correlation R' . Since $R' = R^\dagger$ and $A' = B^{-1}$ application of part (a) to the dual frame yields $B^{-1} = \|(R')^\dagger\|^{-1}$ or $B = \|R\|$. ■

3.4.6 Duality

From Theorem 3.13(a) a frame $\{\phi_n\}$ has an associated dual frame $\{\psi_n\}$, where $\psi_n \stackrel{\triangle}{=} S^{-1}\phi_n$ and S is the frame operator. As a frame, $\{\psi_n\}$ also has a frame representation operator L_ψ , where $L_\psi f \stackrel{\triangle}{=} \{\langle f, \psi_n \rangle\} = \{\langle f, S^{-1}\phi_n \rangle\}$. As a matter of notation both L and L_ϕ are written to indicate the frame representation with respect to the frame $\{\phi_n\}$. Since S is a topological isomorphism, clearly $L_\phi(\mathcal{H}) = L_\psi(\mathcal{H})$. With this notation, Equation (3.9) may be written as

$$\forall f \in \mathcal{H}, \quad f = L_\phi^* L_\psi f = L_\psi^* L_\phi f. \quad (3.17)$$

From this observation it may be concluded that

$$L_\phi^* L_\psi = L_\psi^* L_\phi = I_{\mathcal{H}},$$

where $I_{\mathcal{H}}$ is the identity operator on \mathcal{H} . Furthermore, if attention is restricted to the range $L(\mathcal{H})$ one may write

$$L_\phi^{-1} = L_\psi^*$$

and

$$L_\psi^{-1} = L_\phi.$$

Moreover, the relation between the frame representation L_ϕ and its dual L_ψ is

$$L_\psi = R^{-1} R L_\psi = R^{-1} L_\phi \underbrace{L_\phi^* L_\psi}_{I_{\mathcal{H}}} = R^{-1} L_\phi,$$

where R is the frame correlation associated with the frame $\{\phi_n\}$. Table 3.1 lists the relationships among the frame bounds A and B , frame operator S , frame representation L , and frame correlation R of a frame $\{\phi_n\}$ and its dual frame $\{S^{-1}\phi_n\}$. Dual quantities are denoted with $'$; for example, S' is the dual frame operator.

Elements	Bounds	Operator	Representation	Correlation
$\{\phi_n\}$	(A, B)	S	L	R
$\{S^{-1}\phi_n\}$	(B^{-1}, A^{-1})	S^{-1}	$R^{-1}L$	R^\dagger

TABLE 3.1. Relation of frame objects and their duals.

3.4.7 Iterative Reconstruction

Let \mathcal{H} be a Hilbert space and take $f_* \in \mathcal{H}$ arbitrarily. Assume also that $\{\phi_n\}$ is a frame for \mathcal{H} with frame bounds A and B , frame representation L , and frame correlation R . This section details an iterative procedure for the recovery of a signal f_* from its frame representation Lf_* . The iterative procedure generates a sequence $\{c_n\} \subseteq L(\mathcal{H})$ that converges to a $c_* \in L(\mathcal{H})$ such that $f_* = L^*c_*$. Moreover, the sequence converges at an exponential rate. The algorithm for the computation of the sequence $\{c_n\}$ may be implemented digitally and identified with the computation of the inverse R^{-1} of the frame correlation R .

Frame Operator

It may be easily shown (viz. [DS52, p. 360, Theorem III] the original formulation, [Ben92, Algorithm 50], [Ben93, Section 6.6], and its genesis in [DS52]) that since $\{\phi_n\}$ is a frame for \mathcal{H} with frame bounds A, B ,

$$\|I - \frac{2}{A+B}S\| \leq \frac{B-A}{A+B} < 1,$$

so that by the Neumann expansion,

$$S^{-1} = \frac{2}{A+B} \sum_{j=0}^{\infty} (I - \frac{2}{A+B}S)^j, \quad (3.18)$$

where I is the identity operator in \mathcal{H} . For any $f_* \in \mathcal{H}$ applying (3.18) to Sf_* yields

$$f_* = \sum_{j=0}^{\infty} (I - \lambda S)^j (\lambda S) f_*, \quad (3.19)$$

where $\lambda = 2/(A+B)$.

Frame Correlation

With a view toward digital implementation it is desirable to construct an iterative algorithm for the recovery of f_* from Lf_* . To do this it is first shown that $I - \lambda R$ is a contraction on $L(\mathcal{H})$.

Lemma 3.22 *Let $\{\phi_n\}$ be a frame for \mathcal{H} with frame representation operator L , correlation R , and bounds A, B . If $0 < \lambda < 2/B$ then $\|I - \lambda R\|_{L(\mathcal{H})} < 1$ and $\|I - \lambda R\|_{\ell^2(\mathbb{Z})} = 1$ if $\{\phi_n\}$ is not exact. In particular one may take $\lambda = 2/(A + B)$.*

Proof: Let L' and R' be the dual frame representation and correlation, respectively. Since $(L')^*$ is surjective then for any $f \in \mathcal{H}$ there is a $c \in L'(\mathcal{H})$ so that $f = (L')^*c$. This together with the fact that $\{S^{-1}\phi_n\}$ is a frame for \mathcal{H} yields

$$B^{-1} \langle c, R'c \rangle \leq \langle c, (R')^2 c \rangle \leq A^{-1} \langle c, R'c \rangle. \text{ (c.f. Equation (3.16))}.$$

Letting $c = (R')^\dagger c_0$ one has

$$B^{-1} \langle Rc_0, c_0 \rangle \leq \langle c_0, c_0 \rangle \leq A^{-1} \langle Rc_0, c_0 \rangle.$$

For all nonzero $c_0 \in L'(\mathcal{H})$ this means

$$A \leq \frac{\langle Rc_0, c_0 \rangle}{\langle c_0, c_0 \rangle} \leq B.$$

Thus one has for $\lambda > 0$

$$1 - \lambda B \leq \frac{\langle (I - \lambda R)c_0, c_0 \rangle}{\langle c_0, c_0 \rangle} \leq 1 - \lambda A$$

and combining the facts that $I - \lambda R$ is self-adjoint (cf. Proposition 3.17(g)), and Fact 2.2,

$$\|I - \lambda R\|_{L(\mathcal{H})} = \sup_{c \in L(\mathcal{H})} \frac{|\langle (I - \lambda R)c, c \rangle|}{\langle c, c \rangle} \leq \max \{|1 - \lambda A|, |1 - \lambda B|\}. \quad (3.20)$$

It is desirable to find a value of λ such that $\|I - \lambda R\|_{L(\mathcal{H})} < 1$. This condition is satisfied for all $\lambda \in (0, 2/B)$. In particular if $\lambda = 2/(A + B)$ then

$$|1 - \lambda A| = |1 - \lambda B| = (B - A)/(A + B) < 1.$$

For this choice of λ it has been proven that $\|I - \lambda R\|_{L(\mathcal{H})} < 1$. Clearly, for all nonzero c taken from $\ker R = L(\mathcal{H})^\perp$ one has $|\langle (I - \lambda R)c, c \rangle| / \langle c, c \rangle = 1$. Since $\ell^2(\mathbb{Z}) = L(\mathcal{H}) \oplus L(\mathcal{H})^\perp$ then $\|I - \lambda R\|_{\ell^2(\mathbb{Z})} = 1$ if $L(\mathcal{H})^\perp = \ker L^* \neq \{0\}$; that is, by Theorem 3.18 $\{\phi_n\}$ is not exact. ■

Proposition 3.23 *The signal f_* may be recovered from its frame representation Lf_* as*

$$f_* = \lambda \sum_{j=0}^{\infty} L^*(I - \lambda R)^j L f_*, \quad (3.21)$$

where $L^*c = \sum c_n \phi_n$ for $c = \{c_n\}$.

Proof: Since $\langle Lf, c \rangle \triangleq \langle f, L^*c \rangle$ and

$$\langle Lf, c \rangle = \sum \bar{c}_n \langle f, \phi_n \rangle = \left\langle f, \sum c_n \phi_n \right\rangle,$$

one obtains the formula for L^*c .

Because of (3.19) and the fact that $S = L^*L$, it is sufficient to prove

$$\lambda \sum_{j=0}^{\infty} L^*(I - \lambda R)^j L f_* = \sum_{j=0}^{\infty} (I - \lambda L^*L)^j (\lambda L^*L) f_*. \quad (3.22)$$

The $j = 0$ terms are clearly the same in (3.22). Assume

$$\lambda L^*(I - \lambda R)^j L f_* = (I - \lambda L^*L)^j (\lambda L^*L) f_*. \quad (3.23)$$

Then, using (3.23), compute

$$\begin{aligned} \lambda L^*(I - \lambda R)^{j+1} L f_* &= \lambda L^*(I - \lambda R)^j L f_* - \lambda L^*(I - \lambda R)^j \lambda R L f_* \\ &= \lambda(I - \lambda L^*L)^j (I - \lambda L^*L) L^* L f_* \\ &= \lambda(I - \lambda L^*L)^{j+1} L^* L f_*, \end{aligned}$$

and the result follows by induction. ■

Proposition 3.23 leads directly to Algorithm 3.24 which details an iterative reconstruction procedure for the recovery of the signal f_* from its frame representation Lf_* . Moreover, this iterative procedure will converge at an exponential rate. This algorithm is known as the frame algorithm. It should be mentioned that Gröchenig ([Grö93a]) has developed a weighted version of the frame algorithm that provides faster convergence than the frame algorithm.

Algorithm 3.24 (Frame Algorithm) *Let $\{\phi_n\}$ be a frame for a Hilbert space \mathcal{H} with frame representation L , correlation R , and bounds A, B . Suppose $c_0 \triangleq Lf_*$ is the frame representation of a signal $f_* \in \mathcal{H}$. Set $d_0 = 0$. If $\lambda = 2/(A + B)$ and d_n and f_n are defined as*

$$\begin{aligned} d_{n+1} &\triangleq d_n + (I - \lambda R)^n c_0, \\ f_n &\triangleq \lambda L^* c_n, \end{aligned}$$

then

- (a) $\lim f_n = f_*$, and
- (b) $\|f_n - f_*\|/\|f_*\| < B/A \cdot \alpha^n$, where $\alpha \triangleq \|I - \lambda R\|_{L(\mathcal{H})} < 1$.

Proof: (a) An elementary induction argument shows that

$$\forall n, \quad f_{n+1} = \lambda L^* \left(\sum_{j=0}^n (I - \lambda R)^j \right) c_0.$$

Consequently, by Proposition 3.23, one has

$$\lim f_n = f_*.$$

(b) Write

$$\begin{aligned} \|f_n - f_*\| &= \|(f_{n+1} - f_n) + (f_{n+2} - f_{n+1}) + (f_{n+3} - f_{n+2}) \dots\| \\ &\leq \sum_{k \geq n} \|f_{k+1} - f_k\| \\ &= \sum_{k \geq n} \|\lambda L^*(I - \lambda R)^k L f_*\| \\ &\leq \sum_{k \geq n} \lambda \|L^*\| \|(I - \lambda R)^k\|_{L(\mathcal{H})} \|L\| \|f_*\| \\ &\leq \lambda B \left(\sum_{k \geq n} \alpha^k \right) \|f_*\| \\ &= \left(\frac{\alpha^n}{1 - \alpha} \right) \lambda B \|f_*\| \\ &\leq \frac{B}{A} \alpha^n \|f_*\| \quad (\text{by Equation (3.20).}) \end{aligned}$$

■

Algorithm 3.24 underscores the importance of the correlation frame operator R in the reconstruction process. Formally one may rewrite (3.21) as

$$f_* = \lambda L^* \underbrace{R^{-1}(L f_*)}_{d_*}. \quad (3.24)$$

This algorithm is digitally oriented in the sense that all the processing is done in the coefficient domain. Starting with knowledge of the representation $L f_*$, the algorithm works iteratively to compute the inverse R^{-1} . At each iteration n a new coefficient d_n is generated such that $\lambda L^* d_n$ is closer to f_* . In fact, the application of L^* to d_n need not be performed until n is deemed large enough. Working in the coefficient domain in this fashion,

the algorithm has a hidden numerical pitfall. Namely, the sequence d_n is not guaranteed to converge unless the initial representation is entirely in $L(\mathcal{H})$.

A crucial element in determining convergence of the sequence d_n in Algorithm 3.24 is the fact that $c_0 = Lf_*$, and in particular that $c_0 \in L(\mathcal{H})$. In fact, if c_0 is not entirely in $L(\mathcal{H})$ then the sequence d_n will *not* converge (cf. Lemma 3.22) since $I - \lambda R$ is not a contraction on all of $\ell^2(\mathbb{Z})$. To see this, consider an initial representation entirely outside the range of L ; that is, $c_0 = c_\perp \in L(\mathcal{H})^\perp$. By Proposition 3.17(d) $c_\perp \in \ker R$ so that $(I - \lambda R)c_\perp = c_\perp$ and hence

$$\forall n, \quad \|d_{n+1} - d_n\| = \|(I - \lambda R)^n c_\perp\| = \|c_\perp\|.$$

Thus, unless $c_\perp = 0$ the sequence d_n will diverge. This issue is addressed in Section 7.1.8 where a second algorithm is presented that converges on all of $\ell^2(\mathbb{Z})$. There, the idea is to compute R^{-2} of Rc_0 so as to ensure that the modified initial coefficient sequence Rc_0 is in $L(\mathcal{H})$.

Problems

3.1 Signal Representation. Given the signal

$$f(t) = \sum_{n=1}^{\infty} \frac{1}{n} 1_{[n-1,n)}(t)$$

the pair

$$c_n = 1/n, \phi_n = 1_{[n-1,n)}$$

is an obvious representation. With the atomic set

$$\psi_n = (-1)^n \cdot 1_{[0,n)}$$

it is true that $\phi_n = (-1)^n(\psi_n + \psi_{n-1})$ and

$$\begin{aligned} f(t) &= \sum_{n=1}^{\infty} \frac{1}{n} (\psi_n + \psi_{n-1})(t) \\ &= \sum_{n=1}^{\infty} \left(\frac{1}{n} + \frac{1}{n+1} \right) \psi_n(t). \end{aligned}$$

Show that the right-hand side summation of this last equation is divergent for every choice of t . What's gone wrong?

3.2 Frames in \mathbb{R}^2 . Consider the Hilbert space $\mathcal{H} = \ell^2(\{1, 2\}) \subseteq \mathbb{R}^2$. Give two specific example atomic sets $\{\phi_n\}$ that form

- (a) an orthonormal basis for \mathcal{H} ;
- (b) an exact frame (Riesz basis) for \mathcal{H} ;
- (c) a tight frame for \mathcal{H} ;
- (d) a nonexact frame for \mathcal{H} ; and
- (e) a nontight and nonexact frame for \mathcal{H} .

Compute the frame bounds for each of the preceding frames.

3.3 For each of the bases of Exercise 3.2 write a reconstruction formula to recover the original vector $f \in \mathbb{R}^2$ from its representation $Lf \triangleq \{\langle f, \phi_n \rangle\}$.

3.4 Let $\phi_n \in \ell^2(\{0, 1, \dots, N-1\})$ be the n th harmonic of the complex exponential; that is, $\phi_n(k) = e_{n/N}(k) = e^{j2\pi nk/N}$. Show that

$$\langle \phi_m, \phi_n \rangle_{\ell^2(\{0, 1, \dots, N-1\})} = N \cdot \delta_{m,n}$$

for all $m, n \in \{0, 1, \dots, N-1\}$. Conclude that $\{\phi_n/\sqrt{N}\}_{n=0}^{N-1}$ is an orthonormal basis for $\ell^2(\{0, 1, \dots, N-1\})$.

3.5 Suppose that $\Omega = 1/(2\Delta)$ for some $\Delta > 0$.

- (a) Show that for all $f \in PW_\Omega$ and all $t \in \mathbb{R}$,

$$\langle f, \tau_t d_{2\pi\Omega} \rangle = f(t).$$

- (b) Using Theorem 2.4 show that $\{\phi_n\}$, where

$$\forall n, \quad \phi_n = \sqrt{\Delta} \tau_{n\Delta} d_{2\pi\Omega},$$

is an orthonormal basis for PW_Ω .

- (c) Identify the representation operator L associated with $\{\phi_n\}$ and its adjoint L^* and explicitly write the reconstruction formula $L^* L f$.

3.6 Write an explicit formula for the underlying topological isomorphism T associated with the Riesz bases of Example 3.9.

3.7 Suppose $\{\phi_n\}$ is a Riesz basis for \mathcal{H} with frame operator S and T is the topological isomorphism that relates ϕ_n to an orthonormal basis. Show that $S = TT^*$.

3.8 Prove Theorem 3.13.

3.9 *Overcomplete Frame in \mathbb{R}^3* . Consider the representation of elements in $\mathcal{H} = \mathbb{R}^2$ with respect to the atoms

$$L^* = \begin{pmatrix} | & | & | \\ \phi_1 & \phi_2 & \phi_3 \\ | & | & | \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}.$$

- (a) Compute R .

- (b) Verify that $R = U^* \begin{pmatrix} 3 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} U$, where

$$U = \frac{1}{\sqrt{6}} \begin{pmatrix} 1 & 1 & 2 \\ -\sqrt{3} & \sqrt{3} & 0 \\ \sqrt{2} & \sqrt{2} & -\sqrt{2} \end{pmatrix}.$$

- (c) Use Problem 3.9(b) to compute R^\dagger .

- (d) Compute $P_{L(\mathcal{H})}$.

- (e) Let $c_1 = \begin{pmatrix} 1 \\ 1 \\ 2 \end{pmatrix}$ and $c_2 = \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix}$. Compute $P_{L(\mathcal{H})} c_1$ and $P_{L(\mathcal{H})} c_2$.

- (f) Compute L^*c_1 and $L^*(c_1 + c_2)$.
- 3.10 Suppose $\{\phi_n\}$ is a frame for \mathcal{H} with representation L and correlation $R = LL^*$. In this exercise show that the coefficient sequence $c_* = R^\dagger Lf$ is unique in the following sense. Of all the coefficient sequences $c \in \ell^2(\mathbb{Z})$ that recover f (via $f = L^*c$), c_* is the smallest one, that is, the one with the smallest norm.

- (a) Show that if $f = L^*(R^\dagger Lf + c_\perp)$ then $c_\perp \in L(\mathcal{H})^\perp$.
 (b) Show that

$$\|R^\dagger Lf + c_\perp\|_{\ell^2(\mathbb{Z})}^2 = \|R^\dagger Lf\|_{\ell^2(\mathbb{Z})}^2 + \|c_\perp\|_{\ell^2(\mathbb{Z})}^2.$$

Thus, the norm is minimized for $c_\perp = 0$.

- 3.11 Suppose $\{\phi_n\}$ is a frame for \mathcal{H} with representation L . Show that

$$\forall f \in \mathcal{H}, \quad \|f\| = \|(R^\dagger)^{1/2}Lf\|.$$

4

Continuous Wavelet and Gabor Transforms

The continuous wavelet transform (CWT) as well as the continuous Gabor transform (CGT) (also known as the short-time Fourier transform) and their inverses are presented in this chapter. Both the CGT and the CWT take a one-dimensional time signal to a two-dimensional function of time and frequency. As such, they both seek to extract the time–frequency characteristics of the one-dimensional signal.

Both transforms may be described in terms of (an inner-product) representation of a signal with respect to a specific family of (atomic) functions that are generated by a single *analyzing* function. In the case of the wavelet transform, signals are decomposed in terms of families of atoms consisting of all the translated and *dilated* versions of the analyzing function. In the case of the Gabor transform, however, functions are decomposed in terms of families of atoms consisting of all the translated and *modulated* versions of the analyzing function. Thus, the common element of wavelet and Gabor families is translation and the distinguishing element is dilation versus modulation. Accordingly, the key difference between the two transforms is that the Gabor atoms have a fixed bandwidth and the wavelet atoms have bandwidths that range continuously from arbitrarily small to arbitrarily large. The fact that the wavelet transform represents signals with atoms of various bandwidths naturally endows it with an inherent “zooming” ability that more naturally facilitates the simultaneous analysis of short-duration (transient) and long-duration signals.

The common feature of the transforms is that atoms are generated from a single analyzing function through the application of an operator which depends continuously on values of time t and frequency γ (or scale s). In the case of the wavelet transform, this operator consists of translation by t and dilation by s ; in the case of the Gabor transform, the operator consists of translation by t and modulation by γ . This operator-theoretic view of the two transforms may be exploited to yield a greatly generalized and unified theoretical framework for the simultaneous study of a large class of time–frequency transforms in terms of mathematical groups: the affine and Weyl–Heisenberg groups are of particular interest for the wavelet and

Gabor transforms, respectively. This framework is developed at the end of the chapter in Section 4.6.

4.1 What Is a Wavelet?

A wavelet is a function with some special properties. Literally, the term “wavelet” means *little wave*. In the most general context, a wavelet is a function that satisfies the main (time domain) conditions:

1. it has a small concentrated burst of finite energy in the time domain; and
2. it exhibits some oscillation in time.

The first condition makes the wavelet “little” in the sense that it is well localized in time, whereas the second condition makes it “wavy” and hence a *wavelet*.

A single wavelet function generates a family of *wavelets* by dilating (stretching or contracting) and translating (moving along the time axis) itself over a continuum of dilation and translation values. If g is a wavelet analyzing function then the set,

$$\{\tau_t D_s g\},$$

of all the dilated (by $s \neq 0$) and translated (by t) versions of g is the wavelet family generated by g . Here, s represents “scale” and t represents “translation.” For this family of functions, g is said to be the *mother* or *analyzing* wavelet. Figure 4.1 illustrates several example functions from a family of wavelets. The left plot shows the time domain functions and the right plot shows their corresponding magnitude Fourier transforms. Dilation in time by contracting values of scale ($s > 1$) corresponds to stretching dilation in the frequency domain; that is, $(D_s g)^\wedge = D_{s^{-1}} \widehat{g}$. The primary effects of such dilation on the frequency support of g are (i) translation¹ along the frequency axis toward higher frequencies, and (ii) increased support (bandwidth widening). These effects are clearly seen in Figure 4.1.

Of course, the main interest in developing families of wavelets is for their utility in representing and, in fact, characterizing functions. As shown in Section 4.3.1, the continuous wavelet transform of a function f is defined as the inner product representation

$$\langle f, \tau_t D_s g \rangle$$

¹Strictly speaking, this effect occurs only for analyzing wavelets that have a frequency support which does not contain 0, that is, wavelets with zero mean. Later, this is seen to be a requirement for admissible wavelets though there are recent variations that allow for wavelets with zero mean (viz. [Ald97]).

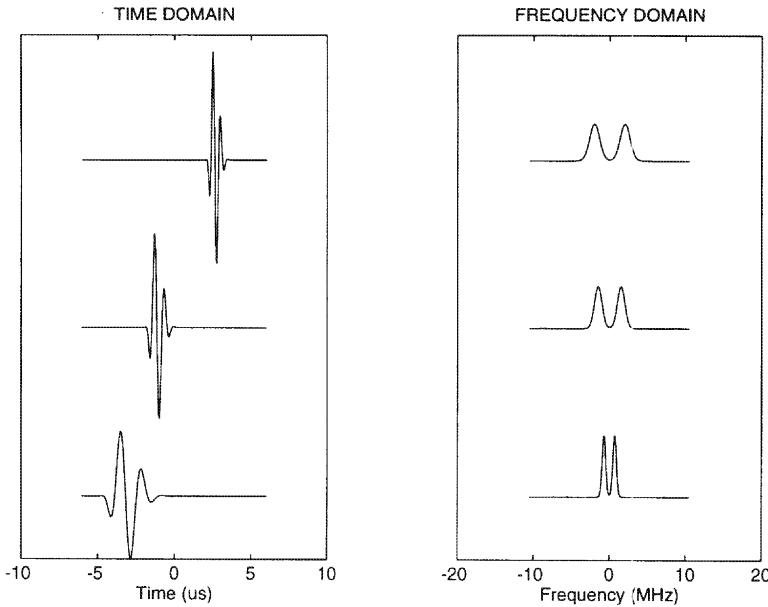


FIGURE 4.1. Members of a family of wavelets shown in both the time (left) and frequency (right) domains. All members are generated by the dilation and translation in time of a single function called the analyzing wavelet.

of f with respect to the wavelet family generated by g . For each point (t, s) in the time-scale plane, the magnitude of the wavelet transform of a signal f describes how much f is like a translated by t and scaled by s version of g .

For almost all reasonable choices of the analyzing wavelet g , signals are recoverable from knowledge of their wavelet transforms; that is, the CWT is invertible. Yet, it is clear that, in order to recover the signal f from its wavelet representation, the analyzing wavelet g cannot be chosen arbitrarily in $L^2(\mathbb{R})$. For one thing, it is necessary that the analyzing wavelet g be such that its wavelet family $\{\tau_t D_s g\}$ span a large enough space of interest. That is to say that every function f of interest should be representable as a linear combination of dilated and translated versions of g . Moreover, each function should be recoverable from knowledge of all the inner products $\{\langle f, \tau_t D_s g \rangle\}$. In this light, it is reasonable to impose a third condition on the wavelet family. As shown in Section 4.4, it turns out that requiring $g \in L^1(\mathbb{R})$ (g not identically zero) to have zero mean is a sufficient condition to ensure the invertibility of the CWT. The zero mean condition is equivalently and concisely stated as $\hat{g}(0) = 0$. Since a nonzero function with zero mean necessarily has some oscillation, the oscillation requirement of the wavelet is naturally satisfied by the zero-mean condition.

Accordingly, there is a diverse collection of functions available as candidates for an analyzing wavelet. This freedom is an important aspect in the design of generally overcomplete wavelet transforms and is exploited in later chapters.

4.2 Example Wavelets

In the following subsections some examples of valid analyzing wavelets that meet the requirements of the previous section are described. The example wavelets presented are the Haar, Shannon (or sinc), frequency B-spline, and Morlet wavelets. For the cases of the Shannon and the frequency B-spline wavelets, the wavelets are constructed directly in the frequency domain such that they have compact support on a frequency interval (not containing zero) described by a center frequency γ_c and a bandwidth γ_b given by $(\gamma_c - \gamma_b/2, \gamma_c + \gamma_b/2]$.

4.2.1 Haar Wavelet

The Haar wavelet has become a widely used example of a wavelet function that leads to an orthonormal basis when its wavelet family is restricted to integer translations and dilations by powers of 2 (the so-called *dyadic lattice*). As such, the Haar wavelet is a member of an exclusive group of functions that by being required to generate orthonormal bases on the dyadic lattice has relatively few members.

The Haar wavelet g_{Haar} is defined as

$$g_{\text{Haar}} = 1_{(-1/2, 0]} - 1_{(0, 1/2]}.$$

In the frequency domain the Haar wavelet behaves as

$$\begin{aligned} \widehat{g}_{\text{Haar}} &= (\tau_{-1/4}1_{(-1/4, 1/4]} - \tau_{1/4}1_{(-1/4, 1/4]})^{\wedge} \\ &= (e_{1/4} - e_{-1/4}) d_{\pi/2}, \end{aligned}$$

so that explicitly

$$\widehat{g}_{\text{Haar}}(\gamma) = 2j \cdot \frac{\sin^2\left(\frac{\pi\gamma}{2}\right)}{\pi\gamma}.$$

Because the frequency envelope of the Haar wavelet decays only as fast as $1/\gamma$, frequency localization associated with the Haar wavelet is poor. The Haar wavelet is shown in Figure 4.2.

4.2.2 Shannon Wavelet

The essence of the Shannon (or sinc) wavelet g_{Shannon} is that its Fourier transform is constant over some interval of frequencies excluding the origin

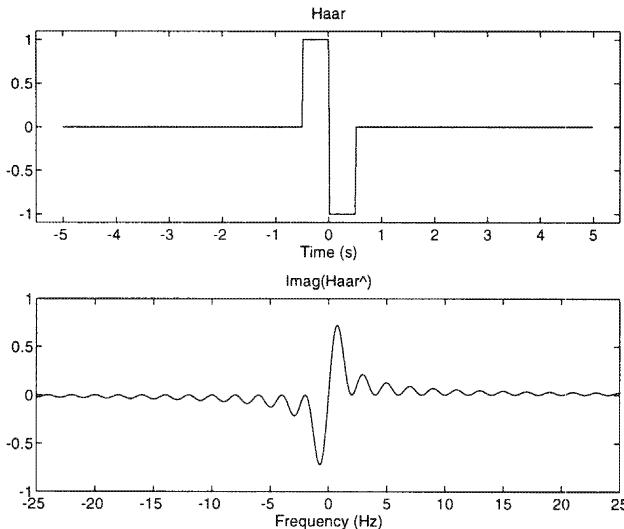


FIGURE 4.2. The Haar wavelet in the time and frequency domains.

and zero elsewhere. As mentioned earlier, the frequency interval of support is described in terms of a desired center frequency γ_c and a desired bandwidth γ_b . Clearly, the condition $\gamma_c > \gamma_b/2$ is sufficient to ensure that zero is not in the frequency support interval. More concretely, the Shannon wavelet may be defined as the inverse Fourier transform of the (normalized with respect to the L^2 norm) indicator function of the interval $(\gamma_c - \gamma_b/2, \gamma_c + \gamma_b/2]$; that is,

$$g_{\text{Shannon}} \triangleq \left(\tau_{\gamma_c} D_{\gamma_b^{-1}} \mathbb{1}_{(-1/2, 1/2]} \right)^{\vee} = e_{\gamma_c} D_{\gamma_b} d_{\pi},$$

where d_{π} is the (sinc) Dirichlet kernel (2.12). The Shannon wavelet is given explicitly by

$$g_{\text{Shannon}}(t) = \gamma_b^{-1/2} e^{j2\pi\gamma_c t} \frac{\sin(\pi\gamma_b t)}{\pi t}.$$

Because the envelope of the Shannon wavelet is a Dirichlet (sinc) function, the Shannon wavelet has poor (as $1/t$) time decay.

4.2.3 Frequency B-spline Wavelets

As in the Shannon case, the frequency B-spline wavelets are defined directly in the frequency domain on a compact frequency interval of support described in terms of a desired center frequency γ_c and a desired band-

width γ_b . The frequency B-spline wavelets² are, in fact, an entire family of valid analyzing wavelets indexed by an integer order parameter m . These wavelets are a generalization of the Shannon wavelet in the sense that the frequency B-spline wavelet with $m = 1$ is the Shannon wavelet.

With g_m denoting the m th order frequency B-spline wavelet, g_m is defined with respect to an auxiliary set of functions θ_m whose Fourier transforms are the m th order B-splines. More specifically, $\theta_m \stackrel{\triangle}{=} d_{\pi}^m$ or

$$\theta_m(t) \stackrel{\triangle}{=} d_{\pi}^m(t) = \left(\frac{\sin \pi t}{\pi t} \right)^m, \quad m = 1, 2, 3, \dots.$$

In the frequency domain this corresponds to

$$\widehat{\theta}_m = \underbrace{1_{[-1/2, 1/2]} * 1_{[-1/2, 1/2]} * \dots * 1_{[-1/2, 1/2]}}_{m \text{ times}}.$$

The frequency B-spline wavelet is defined on the frequency domain interval $(\gamma_c - \gamma_b/2, \gamma_c + \gamma_b/2]$ through the L^2 norm-preserving operations of translation and dilation of $\widehat{\theta}_m$ as

$$\widehat{g}_m \stackrel{\triangle}{=} \tau_{\gamma_c} D_{m\gamma_b^{-1}} \widehat{\theta}_m.$$

Taking the inverse Fourier transform gives $g_m = e_{\gamma_c} D_{m^{-1}\gamma_b} \theta_m$ or explicitly

$$g_m(t) = (\gamma_b/m)^{(1/2-m)} e^{j2\pi\gamma_c t} \left(\frac{\sin(\frac{\pi\gamma_b t}{m})}{\pi t} \right)^m.$$

As mentioned earlier, setting $m = 1$ in the preceding equation gives the Shannon wavelet; that is, $g_1 = g_{\text{Shannon}}$. Real wavelets may be defined from these necessarily analytic wavelets by simply taking the real part. An example of a frequency B-spline wavelet is shown in Figure 4.3. Here, values of bandwidth, center frequency, and order are $\gamma_b = 10$, $\gamma_c = 5$, and $m = 2$.

As defined, the function $\widehat{\theta}_m$ is the so-called m th order *B-spline* function and has the properties listed in the following fact.

Fact 4.1 ([Chu92]) *The m th order B-spline $\widehat{\theta}_m$ has the following properties.*

1. $\sum_n \tau_n \widehat{\theta}_m = 1$ a.e.,
2. $\text{supp } \widehat{\theta}_m = [-\frac{m}{2}, \frac{m}{2}]$,

²Note that the *frequency* B-spline wavelets described here are different from the B-spline wavelets common in the literature.

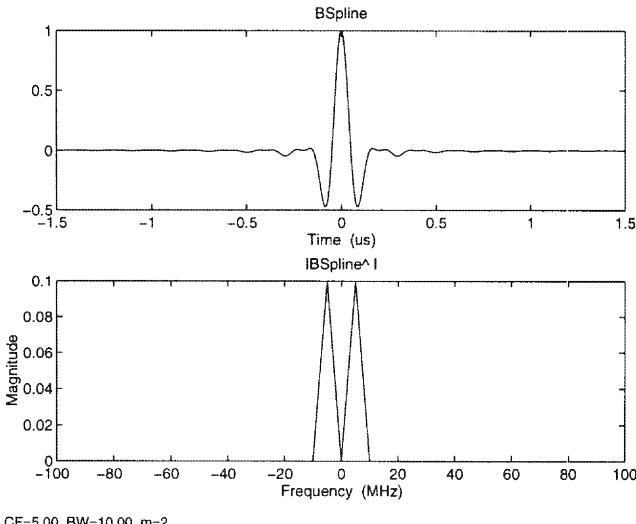


FIGURE 4.3. A second-order real B-spline wavelet in the time and frequency domains.

3. $0 < \hat{\theta}_m \leq 1$ on $(-\frac{m}{2}, \frac{m}{2})$,
4. $\partial \hat{\theta}_m = \tau_{1/2} \hat{\theta}_{m-1} - \tau_{-1/2} \hat{\theta}_{m-1}$,
5. $\partial^{(k)} \hat{\theta}_m < 0$ on $(0, \frac{m}{2})$, $k = 0, 1, \dots, m-1$, and
6. $\hat{\theta}_m$ is real and even.

Proof: (1) Clearly, $\sum_n \tau_n \hat{\theta}_1 = 1$. Now, write

$$\begin{aligned}
 \sum_n \tau_n (\hat{\theta}_m * \hat{\theta}_1) &= \sum_n \hat{\theta}_m * \tau_n \hat{\theta}_1 \\
 &= \hat{\theta}_m * \underbrace{\sum_n \tau_n \hat{\theta}_1}_1 \\
 &= \int \hat{\theta}_m = \theta_m(0) = 1^m = 1.
 \end{aligned}$$

(2–6) Exercise.

4.2.4 Morlet Wavelet

In the same vein as the Shannon and frequency B-spline wavelets, the Morlet wavelet may be specified in the frequency domain again using the

parameters:

1. center frequency γ_c , and
2. variance (bandwidth) γ_b .

However, the frequency support of the Morlet wavelet is not a compact interval but the entire frequency axis. In fact, the Morlet wavelet is a modulated Gaussian function. In this case, the bandwidth is defined as the variance of the Fourier transform of the Morlet wavelet

$$\gamma_b = \int \gamma^2 \hat{g}_{\text{Morlet}}(\gamma) d\gamma.$$

Precisely, the Morlet wavelet g_{Morlet} is given as

$$g_{\text{Morlet}}(t) = \frac{1}{\sqrt{\pi\gamma_b}} \cdot e^{j2\pi\gamma_c t - (t^2/\gamma_b)}. \quad (4.1)$$

Note that the Morlet function is involutive ($g_{\text{Morlet}} = \tilde{g}_{\text{Morlet}}$) and (L^1) normalized in the sense that

$$\|g\|_1 \triangleq \int |g_{\text{Morlet}}| = 1.$$

Because of this fact its Fourier transform \hat{g}_{Morlet} has a maximum value of 1 (viz. Exercise 2.2). In addition, this maximum occurs at γ_c so that $\hat{g}_{\text{Morlet}}(\gamma_c) = 1$. Figure 4.4 displays the Morlet wavelet with $\gamma_c = 1$ (MHz) and $\gamma_b = 1$. In fact, the Fourier transform may be computed analytically as (Exercise 2.5)

$$\hat{g}_{\text{Morlet}}(\gamma) = e^{-\pi^2\gamma_b(\gamma-\gamma_c)^2}.$$

Strictly speaking, the Morlet wavelet does not satisfy the zero-mean requirement; however, the mean can be made arbitrarily small by picking the product $\gamma_b \cdot \gamma_c^2$ large enough. Even so, there are small modifications that can be made to the Morlet wavelet to make it have zero mean. More practically, the Morlet wavelet may be used without modification in the case where the signals to be analyzed are in $L^1(\mathbb{R})$ and have zero mean themselves, for example, narrow band frequency modulated signals as found in sonar or radar applications. This issue is addressed in Exercise 4.4.

4.2.5 Time-Frequency Tradeoffs

With respect to time and frequency localization, the Haar and Shannon wavelets take opposite extremes. Having compact support in time, the Haar wavelet has poor decay in frequency (as $1/\gamma$), whereas the Shannon wavelet has compact support in frequency with poor decay in time (as $1/t$). Other wavelets typically fall in the middle of these two extremes. In fact, having exponential decay in both the time and frequency domain, the Morlet wavelet has optimal joint time-frequency concentration with respect to the L^2 centroid product à la the uncertainty principle (viz. Section 6.3.1).

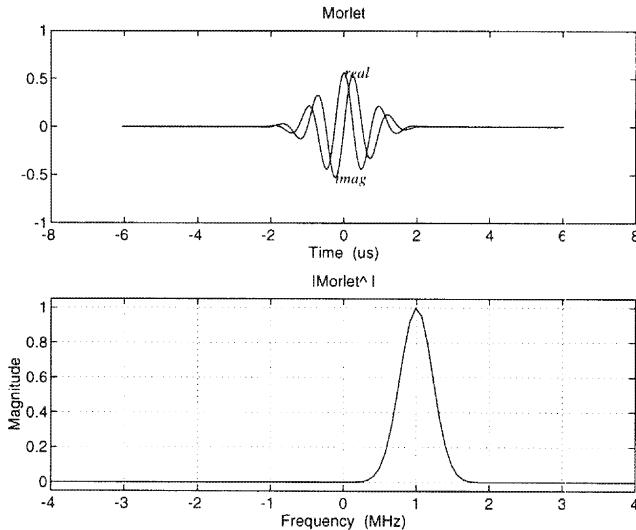


FIGURE 4.4. Morlet analyzing wavelet in the time and frequency domains (center frequency at 1MHz and unit variance).

4.3 Continuous Wavelet Transform

In this section, the continuous wavelet transform (CWT) of a function of one-dimension is rigorously defined and some of its fundamental properties are discussed. The CWT may be defined as a mapping W_g that is dependent on the specification of an auxiliary function g called the analyzing wavelet. Given a specific analyzing wavelet g , the CWT may be thought of in terms of the representation of a signal with respect to the wavelet family generated by g , that is, all its translated and dilated versions. As discussed earlier, there is great flexibility in the choice of analyzing wavelet g . Despite the fact that it must satisfy an admissibility condition, the analyzing wavelet is otherwise open to design. In fact, any nontrivial function in $L^1(\mathbb{R})$ with zero mean is a valid candidate for the analyzing wavelet with respect to the CWT. Admissibility is discussed in Section 4.4.4.

4.3.1 Definition

For a general Hilbert space \mathcal{H} , the CWT may be described as a mapping $W_g : \mathcal{H} \mapsto W_g(\mathcal{H})$ parameterized by a function g . More specifically, the CWT of a one-dimensional function $f \in L^2(\mathbb{R})$ is given by

$$\begin{aligned} W_g &: L^2(\mathbb{R}) &\rightarrow W_g(L^2(\mathbb{R})) \\ f &&\mapsto \langle f, \tau_t D_s g \rangle_{L^2(\mathbb{R})}, \end{aligned}$$

where $\tau_t D_s g$ is a dilated³ (by s) and translated (by t) version of g given explicitly as

$$(\tau_t D_s g)(\sigma) = |s|^{1/2} g(s(t - \sigma)).$$

Thus, the CWT of a one-dimensional signal is a two-dimensional function of the real variables time t and scale $s \neq 0$ and may be written as

$$(W_g f)(t, s) \stackrel{\triangle}{=} \langle f, \tau_t D_s g \rangle = (f * D_s \tilde{g})(t). \quad (4.2)$$

For particular values of scale s and time t the wavelet transform assigns a (complex) numerical value to the function f which quantitatively describes the degree of similarity between the function f and a t -translated and s -dilated version of f . Written out explicitly the CWT of a signal f is

$$(W_g f)(t, s) = |s|^{1/2} \int_{\mathbb{R}} f(\sigma) \bar{g}(s(t - \sigma)) d\sigma.$$

Provided that the analyzing function is chosen with enough time-frequency localization, the CWT gives a picture of the time-frequency characteristics of the function f over the whole time-scale plane $\mathbb{R} \times (\mathbb{R} \setminus \{0\})$.



FIGURE 4.5. The CWT as the output of a continuous bank of linear filters.

From Equation (4.2), the wavelet transform $W_g f$ of a function f may be interpreted as the output of an infinite bank of linear filters described by the impulse responses $D_s \tilde{g}$ over a continuous range of scales $s \neq 0$ (see Figure 4.5). Consider such an infinite filter bank indexed by scale and stacked vertically such that higher filters correspond to larger values of scale s . For an analyzing function with a well-defined bandwidth, as the magnitude of s increases, the bandwidth both increases and travels toward higher and higher frequencies. Thus, lower filters respond to a lower and smaller range of frequencies and higher filters in the bank respond to a higher and larger range of frequencies.

³It is important to point out that the way in which the dilation operator D_s is defined leads to a *directly* proportional relationship between scale and frequency. This is in contrast to the more standard definition in which the relationship is inversely proportional. Although it is not standard, the direct definition leads to cleaner inverse reconstruction formulae and implementations. With this understanding, the terms “time-scale” and “time-frequency” are used somewhat interchangeably.

As an example of a CWT, Figure 4.6 displays the wavelet transform of a synthetic chirp signal. Here, the analyzing wavelet is the Morlet wavelet depicted in Figure 4.4. At the very top of the figure is plotted the signal and the corresponding CWT is presented as an image in (a subset of) the time–frequency plane. In this case the vertical (scale) axis is labeled in terms of the strongest response frequencies associated with the dilated by s versions of the analyzing wavelet.

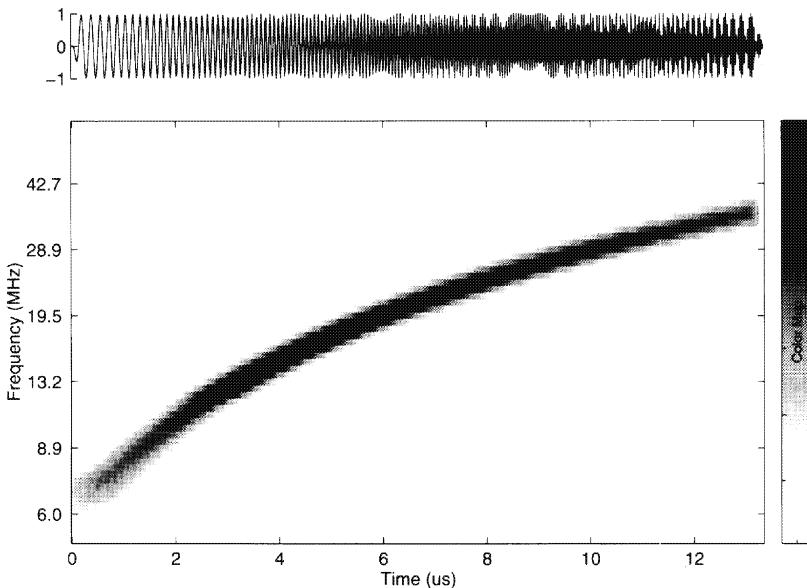


FIGURE 4.6. The “continuous” wavelet transform magnitude of a chirp signal.

4.3.2 Properties

Let a and b be arbitrary real numbers and f, f_1 , and f_2 be arbitrary functions from $L^2(\mathbb{R})$. The CWT, W_g , with respect to an admissible analyzing wavelet g satisfies the following properties.

1. (Linearity) $(W_g(af_1 + bf_2))(t, s) = a(W_gf_1)(t, s) + b(W_gf_2)(t, s)$,
2. (Time Invariance) $(W_g(\tau_b f))(t, s) = (W_g f)(t - b, s)$,
3. (Dilation) $(W_g(D_a f))(t, s) = (W_g f)(at, a^{-1}s)$, $a \neq 0$, and
4. (Negative Scales) $(W_g f)(t, -s) = (W_{\mathcal{R}g} f)(t, s) = \mathcal{W}_g \mathcal{R}f(-t, s)$.

Linearity and Time Invariance

Linearity is inherited directly from the inner product whereas time invariance may be seen from the fact that the CWT can be interpreted in terms of a bank of linear time-invariant filters. Time invariance is an important property in terms of pattern recognition since it dictates that the wavelet transform of a time-delayed version of a signal is a time-delayed version of its wavelet transform. This property is lost for fixed sampling sets (discretizations) in the time-scale plane and is especially problematic for the discrete wavelet transform.

Dilation

To see the dilation property write

$$\begin{aligned} (W_g(D_a f))(t, s) &\stackrel{\Delta}{=} \langle D_a f, \tau_t D_s g \rangle = \langle f, D_{a^{-1}} \tau_t D_s g \rangle \\ &= \langle f, \tau_{at} D_{a^{-1}} s g \rangle = (W_g f)(at, a^{-1}s). \end{aligned}$$

Negative Scales

Noting that the reflection operator and the dilation operator commute (i.e., $\mathcal{R}D_s = D_s\mathcal{R}$) and also that $\mathcal{R} = \tau_t \mathcal{R} \tau_t$ for all $t \in \mathbb{R}$, the negative scale properties are evident from

$$\begin{aligned} (W_g f)(t, -s) &\stackrel{\Delta}{=} \langle f, \tau_t D_{-s} g \rangle = \langle f, \tau_t D_s \mathcal{R} g \rangle = (W_{\mathcal{R}g} f)(t, s) \\ &= \langle f, \tau_t \mathcal{R} D_s g \rangle = \langle f, \mathcal{R} \tau_{-t} D_s g \rangle \\ &= \langle \mathcal{R}f, \tau_{-t} D_s g \rangle = (W_g \mathcal{R}f)(-t, s). \end{aligned}$$

4.4 Inverse Wavelet Transform

The invertibility of the CWT is an important feature of wavelet theory. Analytic formulae to reconstruct a function from its CWT and the restrictions placed on analyzing wavelets for such an inverse to exist are presented in this section. Although this restriction, the so-called *admissibility condition*, limits the functions available for use as an analyzing wavelet, it is not too restrictive and, in fact, is satisfied by any zero-mean function that is absolutely integrable.

4.4.1 *The Idea Behind the Inverse*

It is appealing to develop the inverse wavelet transform with respect to the continuous filter bank interpretation of the forward wavelet transform given in Section 4.3.1. In that interpretation the CWT is thought of as

$$(W_g f)(t, s) = (f * D_s \tilde{g})(t),$$

the output of an infinite bank of linear filters described by the impulse responses $\{D_s \tilde{g}\}$. Alternatively, the bank may be described by its frequency responses $\{D_{s-1} \tilde{\bar{g}}\}$.

To determine an inverse for the entire filter bank, first consider the case of just a single filter described by the sole impulse response g . Assume, for the time being, that the functions f of interest have their frequency support entirely within the frequency support of g and that g is well behaved in the following senses.

- \hat{g} is continuous, and
- $|\hat{g}| > \epsilon > 0$ over $\text{supp } \hat{f}$.

In this case, all such f are recoverable from their filtering $f * g$ by g through

$$f = (f * g) * h = \left[\hat{f} \cdot \hat{g} \cdot \left(\bar{\hat{g}} \cdot \frac{1}{|\hat{g}|^2} \cdot 1_{\text{supp } \hat{f}} \right) \right]^\vee, \quad (4.3)$$

where $\hat{h} \triangleq \bar{\hat{g}} \cdot (1/|\hat{g}|^2) \cdot 1_{\text{supp } \hat{f}}$ is the filter inverse to g over the frequencies in the support of \hat{f} . Because of the previously imposed conditions the inverse filter h is well-defined. Figure 4.7 shows the schematic implementation of the single filter inverse suggested by Equation (4.3).

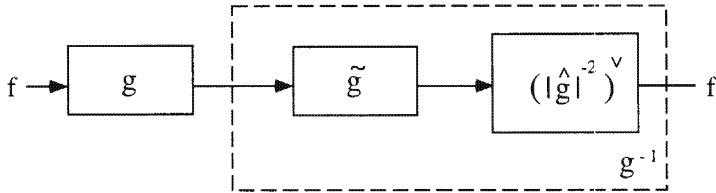


FIGURE 4.7. Filter and its “inverse” implementation.

Now consider a continuously indexed bank of filters $\{g_s\}$ that satisfies

$$0 < \int |\hat{g}_s|^2 ds < \infty \quad (4.4)$$

over the union of the frequency supports of all functions of interest. Let W_g be the operator which takes a one-dimensional signal f to the two-dimensional function $W_g f$ given by the output of the filter bank

$$(W_g f)(t, s) \triangleq (f * g_s)(t).$$

This is an obvious generalization of the wavelet filter bank implementations already discussed; here, however, in contrast to the wavelet case the filter

impulse responses are not constrained to be dilated versions of each other. For the time being, no structure will be imposed on the filter bank. Within this framework, an inverse to the continuous filter bank may be developed using the same ideas as used in the single filter inverse case.

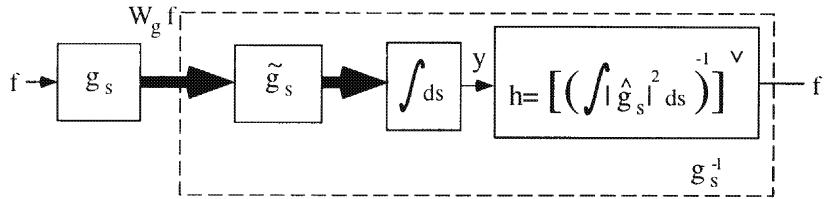


FIGURE 4.8. Continuous filter bank and its “inverse” implementation.

In the full filter bank case, processing includes not only filtering the output of each filter g_s by its involuted counterpart \tilde{g}_s , but also integrating these outputs over all s and filtering the result by a composite inverse h . Figure 4.8 illustrates the implementation of the continuous inverse filter bank. The inverse processing is most easily seen in the frequency domain: the output of the continuous filter bank $\hat{f}\tilde{g}_s$ is passed through the involuted bank $\tilde{\hat{g}}_s$ and then integrated over all s to yield an intermediate output

$$\hat{y} = \int \hat{f}\tilde{g}_s \cdot \tilde{\hat{g}}_s \, ds = \hat{f} \left(\int |\hat{g}_s|^2 \, ds \right).$$

From this equation, \hat{f} may be readily recovered by dividing \hat{y} by $\int |\hat{g}_s|^2$. This operation is well-defined because of the conditions imposed in Equation (4.4) and may be interpreted as a linear filter with impulse response given by h . In this case, the composite inverse filter h is given by

$$h = \left(\frac{1}{\int |\hat{g}_s|^2 \, ds} \right)^v.$$

The preceding development is very general in that it applies to a large class of continuous filter banks. Specialization to wavelet filter banks, however, makes the inverse implementation especially simple. Although not immediately apparent, the imposition of the wavelet structure $g_s = D_{sg}$ on the filter bank leads to an inverse filter h which is just a simple multiplier; that is,

$$\hat{h} = \left(\int |D_{s^{-1}}\hat{g}|^2 \, ds \right)^{-1} = \text{constant}.$$

This fact is shown in Sections 4.4.2 (viz. Equation (4.9)). This is an important property of the wavelet transform which indicates the fact that the

value of the wavelet transform at a particular point (t, s) in the time-scale plane describes directly the relative contribution that a t -translated and s -dilated version of g makes to the overall composition of f .

4.4.2 Derivation for $L^2(\mathbb{R})$

As with the filter bank approach previously discussed and illustrated in Figure 4.8, the inverse wavelet transform

$$W_g^{-1} : W_g(L^2(\mathbb{R})) \mapsto L^2(\mathbb{R}),$$

is derived in this section for functions in $L^2(\mathbb{R})$. Consider a function $f \in L^2(\mathbb{R})$ and the Fourier transform of its CWT:

$$(W_g f) \widehat{\gamma}(\gamma, s) = (\widehat{f} \cdot D_{s^{-1}} \widehat{g})(\gamma). \quad (4.5)$$

Multiplying both sides of this equation by $D_{s^{-1}} \widehat{g}$ and integrating over all nonzero scales $s \neq 0$ yields

$$\int_{s \neq 0} (W_g f) \widehat{\gamma}(\gamma, s) (D_{s^{-1}} \widehat{g})(\gamma) ds = \int_{\mathbb{R}} (\widehat{f} \cdot |D_{s^{-1}} \widehat{g}|^2)(\gamma) ds \quad (4.6)$$

$$= \widehat{f}(\gamma) \int_{\mathbb{R}} |s|^{-1} |\widehat{g}(s^{-1} \gamma)|^2 ds \quad (4.7)$$

$$= \widehat{f}(\gamma) \int_{\mathbb{R}} |u|^{-1} |\widehat{g}(u)|^2 du. \quad (4.8)$$

Here the last equality follows from making the substitution $u = s^{-1} \gamma$ in the integral and noting that $du = -|s^{-1} u| ds$. The fact that the integral

$$C \triangleq \int_{\mathbb{R}} |s|^{-1} |\widehat{g}(s^{-1} \gamma)|^2 ds = \int_{\mathbb{R}} |u|^{-1} |\widehat{g}(u)|^2 \quad (4.9)$$

is independent of s and, in particular, is constant, is an important and remarkable fact of wavelet theory. This fact allows the direct quantitative interpretation of the wavelet coefficients associated with a particular signal as the amount that each dilated and translated version of the analyzing wavelet contributes to the overall composition of the signal. Clearly, the integral in (4.9) must be convergent (i.e., $C < \infty$) for the wavelet inverse to exist. This is exactly the admissibility condition that must be placed on the analyzing wavelet g in order for the inverse wavelet transform with respect to that g to be well-defined. Under these conditions the Fourier transform of the signal f is expressible as

$$\widehat{f}(\gamma) = C^{-1} \int_{\mathbb{R}} (W_g f) \widehat{\gamma}(\gamma, s) (D_{s^{-1}} \widehat{g})(\gamma) ds.$$

Combining (4.8) and (4.9) and transforming to the time domain gives

$$\begin{aligned} f(t) &= C^{-1} \int_{\mathbb{R}} ((W_g f)(\cdot, s) * (D_s g))(t) ds \\ &= C^{-1} \int_{\mathbb{R}} \int_{\mathbb{R}} (W_g f)(\sigma, s) (D_s g)(t - \sigma) ds d\sigma \\ &= C^{-1} \int_{\mathbb{R}} \int_{\mathbb{R}} (W_g f)(\sigma, s) (\tau_\sigma D_s g)(t) ds d\sigma. \end{aligned}$$

Of these three equalities, the top one gives the clearest interpretation of the wavelet inverse in terms of continuous filter banks:

a C -scaled version of the signal f may be recovered from its continuous wavelet transform (f passed through the continuous wavelet filter bank) by passing the filter bank output (at each scale) through the corresponding involutive filter bank and then integrating the results across scale.

The culmination of the previous discussion is succinctly stated in Theorem 4.2.

Theorem 4.2 (Inverse Wavelet Transform) *Let $F \in W_g(L^2(\mathbb{R}))$, $g \in L^2(\mathbb{R}) \setminus \{0\}$ such that*

$$C \triangleq \int_{\mathbb{R}} |\gamma|^{-1} |\hat{g}(\gamma)|^2 d\gamma < \infty,$$

and $W_g^{-1} : W_g(L^2(\mathbb{R})) \mapsto L^2(\mathbb{R})$ be defined as

$$\begin{aligned} W_g^{-1} F &\triangleq C^{-1} \int_{\mathbb{R}} F(\cdot, s) * (D_s g) ds \\ &= C^{-1} \int_{\mathbb{R}} \int_{\mathbb{R}} F(\sigma, s) (\tau_\sigma D_s g) ds d\sigma. \end{aligned}$$

If $f \in L^2(\mathbb{R})$ and $F = W_g f$ then $f = W_g^{-1} F$.

4.4.3 Analytic Signals

There is special interest in functions that are analytic, that is, functions whose Fourier transforms vanish off of the positive (or negative) frequency axis. One reason such functions are important is that they associate with real functions (via the Hilbert transform) a unique (minimum) phase. This is especially useful in communication and radar applications.

Analytic functions are described by the Hardy spaces $H_+^2(\mathbb{R})$ and $H_-^2(\mathbb{R})$. Recall (Section 2.3.8) that $H_+^2(\mathbb{R})$ and $H_-^2(\mathbb{R})$ orthogonally decompose $L^2(\mathbb{R})$ as

$$L^2(\mathbb{R}) = H_-^2(\mathbb{R}) \oplus H_+^2(\mathbb{R}).$$

With $f \in L^2(\mathbb{R})$ then

$$f = f_+ + f_-,$$

where $f_+ \triangleq P_{H_+^2(\mathbb{R})}f$ and $f_- \triangleq P_{H_-^2(\mathbb{R})}f$ are the orthogonal projections of f onto the spaces $H_+^2(\mathbb{R})$ and $H_-^2(\mathbb{R})$, respectively.

Consider the two restricted operators

$$\begin{aligned}(W_g^+)^{-1} &: W_g(H_+^2(\mathbb{R})) \mapsto H_+^2(\mathbb{R}) \\ (W_g^-)^{-1} &: W_g(H_-^2(\mathbb{R})) \mapsto H_-^2(\mathbb{R}).\end{aligned}$$

Because $H_+^2(\mathbb{R})$ and $H_-^2(\mathbb{R})$ orthogonally decompose $L^2(\mathbb{R})$ the following consequences result.

1. (Forward Decomposition)

$$W_g f = W_g(f_+ + f_-) = W_g^+ f_+ + W_g^- f_-, \text{ and}$$

2. (Inverse Decomposition)

$$f = W_g^{-1} W_g f = (W_g^+)^{-1} W_g^+ f_+ + (W_g^-)^{-1} W_g^- f_-.$$

These equations describe how the inverse wavelet transform on all of $L^2(\mathbb{R})$ is related to the wavelet inverses restricted to $H_+^2(\mathbb{R})$ and $H_-^2(\mathbb{R})$. In the remainder of this section the restricted wavelet inverses are explicitly developed. The results are formally stated as corollaries to Theorem 4.2. Although the development of these restricted wavelet inverses parallels that of the development on $L^2(\mathbb{R})$ quite closely, they are included here for completeness.

Restriction to $H_+^2(\mathbb{R})$

Consider a function $f \in H_+^2(\mathbb{R})$. Integrating Equation (4.5) only over positive scales $s > 0$ yields

$$\begin{aligned}\int_{\mathbf{R}^+} (W_g f) \widehat{(\gamma, s)} (D_{s^{-1}} \widehat{g})(\gamma) ds &= \widehat{f}(\gamma) \int_{\mathbf{R}^+} s^{-1} |\widehat{g}(s^{-1} \gamma)|^2 ds \\ &= \widehat{f}(\gamma) \int_{\mathbf{R}^+} u^{-1} |\widehat{g}(u)|^2 du,\end{aligned}$$

and leads to the fact that

$$C_+ \triangleq \int_{\mathbf{R}^+} s^{-1} |\widehat{g}(s^{-1} \gamma)|^2 ds = \int_{\mathbf{R}^+} u^{-1} |\widehat{g}(u)|^2 du. \quad (4.10)$$

The Fourier transform of the signal f is expressible as

$$\widehat{f}(\gamma) = C_+^{-1} \int_{\mathbf{R}^+} (W_g f) \widehat{(\gamma, s)} (D_{s^{-1}} \widehat{g})(\gamma) ds.$$

Combining the preceding and transforming to the time domain gives

$$f(t) = C_+^{-1} \int_{\mathbf{R}^+} ((W_g f)(\cdot, s) * (D_s g))(t) ds.$$

Corollary 4.3 (Inverse Wavelet Transform on $H_+^2(\mathbb{R})$)
Let $F \in W_g(H_+^2(\mathbb{R}))$ and $g \in L^2(\mathbb{R}) \setminus \{0\}$ such that

$$C_+ \triangleq \int_{\mathbf{R}^+} |\gamma|^{-1} |\widehat{g}(\gamma)|^2 d\gamma$$

and $W_g^{-1} : W_g(H_+^2(\mathbb{R})) \mapsto H_+^2(\mathbb{R})$ be defined as

$$\begin{aligned} (W_g^+)^{-1} F &\triangleq C_+^{-1} \int_{\mathbf{R}^+} F(\cdot, s) * (D_s g) ds \\ &= C_+^{-1} \int_{\mathbf{R}} \int_{\mathbf{R}^+} F(\sigma, s) (\tau_\sigma D_s g) ds d\sigma. \end{aligned}$$

If $f \in H_+^2(\mathbb{R})$ and $F = W_g f$ then $f = (W_g^+)^{-1} F$.

Restriction to $H_-^2(\mathbb{R})$

Now consider a function $f \in H_-^2(\mathbb{R})$. The counterpart to Equation (4.8) is now

$$\int_{\mathbf{R}^-} (W_g f) \widehat{(\gamma, s)} (D_{s^{-1}} \widehat{g})(\gamma) ds = \widehat{f}(\gamma) \int_{\mathbf{R}^-} |u|^{-1} |\widehat{g}(u)|^2 du,$$

and

$$C_- \triangleq \int_{\mathbf{R}^-} |u|^{-1} |\widehat{g}(u)|^2. \quad (4.11)$$

Here, too, it is required that $C_- < \infty$, for the wavelet inverse to exist and

$$\widehat{f}(\gamma) = C_-^{-1} \int_{\mathbf{R}^-} (W_g f) \widehat{(\gamma, s)} (D_{s^{-1}} \widehat{g})(\gamma) ds.$$

Combining the preceding and transforming to the time domain gives

$$f(t) = C_-^{-1} \int_{\mathbf{R}^-} ((W_g f)(\cdot, s) * (D_s g))(t) ds.$$

Corollary 4.4 (Inverse Wavelet Transform on $H_-^2(\mathbb{R})$)

Let $F \in W_g(H_-^2(\mathbb{R}))$ and $g \in L^2(\mathbb{R}) \setminus \{0\}$ such that

$$C_- \triangleq \int_{\mathbf{R}^-} |\gamma|^{-1} |\widehat{g}(\gamma)|^2 d\gamma$$

and $W_g^{-1} : W_g(H_-^2(\mathbb{R})) \mapsto H_-^2(\mathbb{R})$ be defined as

$$\begin{aligned} (W_g^-)^{-1} F &\triangleq C_-^{-1} \int_{\mathbf{R}^-} F(\cdot, s) * (D_s g) ds \\ &= C_-^{-1} \int_{\mathbf{R}} \int_{\mathbf{R}^-} F(\sigma, s) (\tau_\sigma D_s g) ds d\sigma. \end{aligned}$$

If $f \in H_-^2(\mathbb{R})$ and $F = W_g f$ then $f = (W_g^-)^{-1} F$.

4.4.4 Admissibility

A main thread of this chapter has been the idea that in order to ensure the invertibility of the CWT, the analyzing wavelet g cannot be chosen arbitrarily. In the previous sections, the condition that functions must satisfy in order to be a viable candidate for an analyzing wavelet (i.e., one that leads to a well-defined wavelet inverse) emerges naturally from the development. From Equations (4.9) through (4.11) this admissibility condition is

$$0 < C = C_+ + C_- = \int_{\mathbb{R}} |\gamma|^{-1} |\hat{g}(\gamma)|^2 < \infty \quad (4.12)$$

or

$$\gamma^{-1/2} \hat{g}(\gamma) \in L^2(\mathbb{R}).$$

The lower bound is satisfied by any nontrivial function (nonzero on a set of nonzero measure). The upper bound says that the analyzing wavelet must be chosen so that in the frequency domain it has sufficient decay around zero. Any zero-mean function g that is continuous around 0 in the frequency domain trivially satisfies this condition. In particular, any function $g \in L^1(\mathbb{R}) \subset L^2(\mathbb{R})$ (absolutely integrable) with zero mean is admissible since $L^1(\mathbb{R})$ functions necessarily have continuous Fourier transforms (viz. Exercise 2.2). Hence, the following theorem formally states this fact.

Theorem 4.5 *Any nontrivial function $g \in L^1(\mathbb{R})$ with zero mean is an admissible wavelet.*

In light of the previous modest admissibility restrictions, there is a vastly rich collection of functions available for use as analyzing wavelets. This great degree of freedom may be exploited to some benefit in applications. This situation is due mainly to the fact that the family of wavelets is highly linearly *dependent* (infinitely so in the continuous case presently under consideration where (t, s) range continuously over the time-scale plane). Later, it is of interest to limit the values of (t, s) to discrete sets in the time-scale plane for computational reasons. Fortunately, there exist many wavelet families that may be restricted to a countable (discrete) set of time-scale values without losing any information about a signal from its representation with respect to the restricted family. In other words, it is possible to exactly reconstruct an analog signal from knowledge of its wavelet transform at only a discrete number of points in the time-scale plane.

The fact that there exist such discrete sets that do not significantly reduce the flexibility in the choice of wavelet function is a key and remarkable factor which has major consequences for the discrete representation of analog signals. This is an important feature of the wavelet (and other) families that facilitate the discrete representation of analog signals through the sampling of a continuous transform. In particular, useful representations of analog signals may be developed in terms of the sampling of the continuous

wavelet transform in the time-scale plane. The resulting representation can be formulated generally as a *overcomplete* (discrete) wavelet transform and this is the main topic of Chapter 6.

4.5 Continuous Gabor Transform

Although the subject matter focus is placed primarily on the wavelet transform, a brief look at the Gabor transform may provide some additional insights into the broader realm of general time-frequency transformations. Such transforms are linked in spirit in that they seek to provide a joint time-frequency description of a signal's composition. Considered together the wavelet and Gabor transforms hint at the deeper underlying (group representation) structure that they share. This connection is explored in the next section.

In this section, the continuous Gabor transform (CGT) of a function of one-dimension and its inverse are presented. Like the CWT, the CGT may be defined as a mapping G_g that is dependent on the specification of an auxiliary function g . Given a specific function $g \in L^2(\mathbb{R})$, the CGT may be described in terms of the representation of a signal with respect to all the translations and *modulations* of the function g . Unlike the wavelet case, there are no inherent restrictions on the choice of the auxiliary function g except that it be of finite (and nonzero) energy.

4.5.1 Definition

With $g \in L^2(\mathbb{R})$, the continuous Gabor Transform (CGT) $G_g f$ of a signal $f \in L^2(\mathbb{R})$ is defined as

$$(G_g f)(t, \gamma) \triangleq \int f(x) e^{-2\pi i \gamma x} \bar{g}(x-t) dx.$$

The Gabor transform may be interpreted as a time-varying Fourier transform where the function g acts as a sliding window in time over f . Despite the fact that in Gabor's original formulation ([Gab46]) the window g was fixed as a Gaussian function, the name "Gabor" nevertheless refers to the general formulation previously shown. It is also called the "short-time Fourier transform." At a particular time instant t the Gabor transform of a function f is the Fourier transform of f modulated by a t -translated version of g ; that is

$$(G_g f)(t, \gamma) = (f \tau_t \bar{g})^\wedge.$$

In this way the Gabor transform attempts to expose the time-frequency content of the underlying signal f .

A rigorous definition of the Gabor transform is presented in the following. For a general Hilbert space \mathcal{H} , the CGT may be described as a mapping

$G_g : \mathcal{H} \mapsto G_g(\mathcal{H})$ parameterized by a function g . More specifically, the CGT of a one-dimensional function $f \in L^2(\mathbb{R})$ is given by

$$\begin{aligned} G_g : L^2(\mathbb{R}) &\rightarrow G_g(L^2(\mathbb{R})) \\ f &\mapsto \langle f, e_\gamma \tau_t g \rangle_{L^2(\mathbb{R})}. \end{aligned}$$

Note that this inner product form of the definition exposes clearly the Gabor atoms as $\{e_\gamma \tau_t g\}$, that is, all the modulations and translations of the analyzing function g .

4.5.2 Inverse Gabor Transform

For the Gabor transform inversion is conceptually a relatively simple matter. In fact, it is shown that any nontrivial analyzing function $g \in L^2(\mathbb{R})$ leads to a Gabor inverse G_g^{-1} .

Since G_g is the Fourier transform of the product of the signal f and the sliding window $\tau_t \bar{g}$,

$$(G_g f)(t, \gamma) = (f \cdot \tau_t \bar{g})^\wedge(\gamma),$$

the product may be recovered as

$$(G_g f)^\vee(t, x) = (f \cdot \tau_t \bar{g})(x).$$

Multiplying both sides of this equation by $\tau_t g$ and integrating over all t yields

$$\begin{aligned} \int_{\mathbb{R}} (G_g f)^\vee(t, x) \cdot (\tau_t g)(x) dt &= \int_{\mathbb{R}} f(x) \cdot (\tau_t \bar{g})(x) (\tau_t g)(x) dt \\ &= \int_{\mathbb{R}} f(x) |\tau_t g(x)|^2 dt \\ &= f(x) \int_{\mathbb{R}} |g(x-t)|^2 dt \\ &= f(x) \cdot \|g\|^2. \end{aligned}$$

Thus, if $g \in L^2(\mathbb{R}) \setminus \{0\}$ then f may be recovered from its Gabor transform as

$$\begin{aligned} f(x) &= C^{-1} \int_{\mathbb{R}} (G_g f)^\vee(t, x) \cdot (\tau_t g)(x) dt \\ &= C^{-1} \int_{\mathbb{R}} \int_{\mathbb{R}} (G_g f)(t, \gamma) e^{j2\pi\gamma x} g(x-t) d\gamma dt, \end{aligned}$$

where $C \triangleq \|g\|^2$. This result is stated formally as Theorem 4.6.

Theorem 4.6 (Inverse Gabor Transform) *Let $F \in G_g(L^2(\mathbb{R}))$, $g \in L^2(\mathbb{R}) \setminus \{0\}$ such that*

$$C \triangleq \|g\|^2 < \infty,$$

and $G_g^{-1} : G_g(L^2(\mathbb{R})) \mapsto L^2(\mathbb{R})$ be defined as

$$\begin{aligned} (G_g^{-1}F)(x) &\triangleq C^{-1} \int_{\mathbb{R}} F^\vee(t, x)(\tau_t g)(x) dt \\ &= C^{-1} \int_{\mathbb{R}} \int_{\mathbb{R}} F(t, \gamma) e^{j2\pi\gamma x} g(x-t) d\gamma dt. \end{aligned}$$

If $f \in L^2(\mathbb{R})$ and $F = G_g f$ then $f = G_g^{-1}F$.

4.6 Unified Representation and Groups

Exploiting the common theme in the wavelet and Gabor transforms of representation with respect to families of atoms generated by a given operator, a general and unified theoretical framework for signal representation may be developed. Such a framework has been developed by Feichtinger and Gröchenig ([FG89]). Key to this development is the idea of a generalized transform called the *group representation transform* which is dependent on the notion of a mathematical group. The wavelet and Gabor transforms are special cases of this generalized transform in the sense that specifying the group associated with the group representation transform to be the affine or Weyl–Heisenberg group recovers the wavelet or Gabor transform, respectively.

Sampling in the group representation domain is a very general tool that may be used for the discrete representation of functions. It leads directly to the study of wavelet and Gabor ([BW93]) frames. This approach to signal discretization is explored further in Section 6.5.2 in the case of wavelet frames. Attention in this chapter is confined to the continuous group representation (wavelet and Gabor) transform.

4.6.1 Groups

A group \mathcal{G} is a set on which an associative binary operation, called the group action, is specified along with a corresponding identity element. The group action is denoted as the operation “.” so that with $x, y \in \mathcal{G}$ then $x \cdot y \in \mathcal{G}$ denotes x acting on y . The identity element of the group is denoted as “ e ,” and the inverse of an element $x \in \mathcal{G}$ is x^{-1} . To formally state the definition, a group is a pair (\mathcal{G}, \cdot) , where \mathcal{G} is a set, “.” is the group action, and there is an $e \in \mathcal{G}$ so that

- (a) if $x \in \mathcal{G}$ then $x \cdot e = x$, and

(b) if $x \in \mathcal{G}$ then there is a $x^{-1} \in \mathcal{G}$ so that $x \cdot x^{-1} = e$.

A group is *Abelian* if the group action is commutative. For example, $(\mathbb{R}, +)$ is an Abelian group with identity element $e = 0$. Non-Abelian groups are of use in the general approach to discrete signal representation presented here. Much of the setup that is presented is adopted from [HW89].

The two examples of groups that follow are found in the background of both the Gabor and wavelet theory.

Example 4.7 (Affine Group) *The affine group is the upper half-plane $\mathcal{G}_A = \mathbb{R} \times \mathbb{R}^+$ along with the group action defined as follows. Let $x, y \in \mathcal{G}_A$ and $x = (t_x, s_x)$ and $y = (t_y, s_y)$; then*

$$x \cdot y = (t_x, s_x) \cdot (t_y, s_y) = (t_x + s_x^{-1}t_y, s_xs_y)$$

and

$$x^{-1} = (t_x, s_x)^{-1} = (-s_xt_x, s_x^{-1})$$

so that the identity element is $e = (0, 1)$.

Example 4.8 (Weyl–Heisenberg Group) *The Weyl–Heisenberg Group is $\mathcal{G}_H = \mathbb{T} \times \mathbb{R} \times \mathbb{R}$ with the following group action. Let $x, y \in \mathcal{G}_H$ so with $z_x, z_y \in \mathbb{T}$ and $t_x, \gamma_x, t_y, \gamma_y \in \mathbb{R}$ we have $x = (z_x, t_x, \gamma_x)$ and $y = (z_y, t_y, \gamma_y)$; then*

$$x \cdot y = (z_x, t_x, \gamma_x) \cdot (z_y, t_y, \gamma_y) = (z_x z_y e^{-2\pi i \gamma_y t_x}, \gamma_x + \gamma_y, t_x + t_y)$$

and

$$(z_x, t_x, \gamma_x)^{-1} = (z_x^{-1} e^{-2\pi i \gamma_x t_x}, -t_x, -\gamma_x)$$

so that the identity element is $e = (1, 0, 0)$.

4.6.2 Weighted Spaces

Define the Hilbert space of μ -square integrable functions as

$$L_\mu^2(\mathcal{G}) \triangleq \left\{ F : \int_{\mathcal{G}} |F(x)|^2 d\mu(x) < \infty \right\}.$$

Let $F, G \in L_\mu^2(\mathcal{G})$. The space $L_\mu^2(\mathcal{G})$ has the associated norm

$$\|F\|_{L_\mu^2(\mathcal{G})} = \left(\int_{\mathcal{G}} |F(x)|^2 d\mu(x) \right)^{1/2}$$

and the inner product

$$\langle F, G \rangle_{L_\mu^2(\mathcal{G})} = \int_{\mathcal{G}} F(x) \overline{G}(x) d\mu(x).$$

A measure μ on a group \mathcal{G} is *left-invariant* if for every integrable function F on \mathcal{G} and every $y \in \mathcal{G}$

$$\int_{\mathcal{G}} F(y^{-1}x)d\mu(x) = \int_{\mathcal{G}} F(x)d\mu(x).$$

Such a measure is called a *left Haar measure* and it is well-known from measure theory that such a measure exists and is unique up to multiplication by a constant.

The *convolution* $F * G$ of F and G elements of $L^2_{\mu}(\mathcal{G})$ is defined as

$$F * G(x) = \int_{\mathcal{G}} F(y)G(y^{-1}x)d\mu(y).$$

For $x, y \in \mathcal{G}$, the *translation* operator $T_x : L^2_{\mu}(\mathcal{G}) \mapsto L^2_{\mu}(\mathcal{G})$ is

$$T_x f(y) = f(x^{-1}y).$$

If μ is the left Haar measure then T_x is unitary and $T_x^{-1} = T_x^* = T_{x^{-1}}$. The *involution* \tilde{F} of $F \in L^2_{\mu}(\mathcal{G})$ is

$$\tilde{F}(x) = \overline{F(x^{-1})}.$$

A function F is *involutive* if $F = \tilde{F}$. The convolution $F * G$ may also be written

$$(F * G)(x) = \left\langle F, T_x \tilde{G} \right\rangle_{L^2_{\mu}(\mathcal{G})}.$$

In the case where $\mathcal{G} = (\mathbb{R}, +)$ these formulas reduce to the standard ones and involutive functions are those whose real parts are even and imaginary parts are odd.

4.6.3 Representation

Let \mathcal{H} be a Hilbert space and \mathcal{G} be a group. A *representation* of \mathcal{G} on \mathcal{H} is a mapping $\Pi : \mathcal{G} \mapsto \mathcal{B}(\mathcal{H}, \mathcal{H})$ that satisfies the relation

$$\Pi(x)\Pi(y) = \Pi(x \cdot y),$$

for all $x, y \in \mathcal{G}$. For example, with respect to the group $(\mathbb{R}, +)$, $\Pi(t) = \tau_t$ is a representation of \mathbb{R} on $L^2(\mathbb{R})$ since $\tau_t \tau_s = \tau_{t+s}$, where $t, s \in \mathbb{R}$.

An element $g \in \mathcal{H}$ is *cyclic* if $\text{span } \{\Pi(x)g\}_{x \in \mathcal{G}} = \mathcal{H}$. For example, $d_{2\pi\Omega} \in PW_{\Omega}$ is cyclic with respect to the group $(\mathbb{R}, +)$ with representation τ_x .

1. A representation Π is *irreducible* if every $g \in \mathcal{H} \setminus \{0\}$ is cyclic.
2. An element $g \in \mathcal{H}$ is *admissible* if

$$\int_{\mathcal{G}} |\langle g, \Pi(x)g \rangle|^2 d\mu(x) < \infty.$$

The set of admissible functions is denoted $\mathcal{A}_{\Pi,\mu}(\mathcal{H})$ so that

$$\mathcal{A}_{\Pi,\mu}(\mathcal{H}) \triangleq \left\{ g \in \mathcal{H} \setminus \{0\} : \int_{\mathcal{G}} |\langle g, \Pi(x)g \rangle|^2 d\mu(x) < \infty \right\}.$$

Note that $\mathcal{A}_{\Pi,\mu}(\mathcal{H}) \subseteq \mathcal{H}$.

3. A representation Π is *square-integrable* if it is irreducible and $\mathcal{A}_{\Pi,\mu}(\mathcal{H})$ is not empty.

4.6.4 Reproducing Kernel

For Hilbert spaces of functions, the Riesz representation theorem states that every bounded linear functional may be represented as an inner product with a particular element from the Hilbert space. Let \mathcal{H} denote the Hilbert space. A functional on \mathcal{H} is any mapping $F : \mathcal{H} \mapsto \mathbb{C}$. In particular, consider the collection of point mappings $\{F_x\}_{x \in \mathcal{G}}$ given as

$$\forall x \in \mathcal{G}, \quad F_x f = f(x),$$

where \mathcal{G} is some domain on which functions in \mathcal{H} are defined. Clearly, the point mapping collection $\{F_x\}_{x \in \mathcal{G}}$ is a set of functionals on \mathcal{H} . It is easy to verify that each member of this set is also linear. If all members of the Hilbert space \mathcal{H} satisfy a pointwise bound on \mathcal{G}

$$\forall x \in \mathcal{G}, \quad |f(x)| < M_x \|f\| \tag{4.13}$$

then the collection $\{F_x\}_{x \in \mathcal{G}}$ is a set of bounded linear functionals. By the Riesz representation theorem, for each $x \in \mathcal{G}$ there is an element $K_x \in \mathcal{H}$ such that the functional F_x may be expressed as

$$F_x f = \langle f, K_x \rangle.$$

The *reproducing kernel* for a Hilbert space $\mathcal{H}(\mathcal{G})$ is a mapping $\mathcal{K} : \mathcal{G} \times \mathcal{G} \mapsto \mathbb{C}$ defined as

$$\mathcal{K}(x, y) \triangleq \langle K_y, K_x \rangle_{\mathcal{H}(\mathcal{G})}.$$

If the pointwise bound of (4.13) holds over \mathcal{G} then $K_y(x) = \mathcal{K}(x, y)$ for $x, y \in \mathcal{G}$ and $\mathcal{H}(\mathcal{G})$ is a reproducing kernel Hilbert space (RKHS).

Example 4.9 For $\Omega > 0$ the Paley–Wiener space PW_Ω is a RKHS with reproducing kernel $\mathcal{K}(s, t) = d_{2\pi\Omega}(t - s)$ since for all $f \in PW_\Omega$

$$\begin{aligned} f(t) &= \left(\widehat{f} \mathbf{1}_{[-\Omega, \Omega]} \right)^\vee(t) \\ &= (f * d_{2\pi\Omega})(t) \\ &= \langle f, K_t \rangle, \end{aligned}$$

where $K_t = \tau_t d_{2\pi\Omega}$.

4.6.5 Group Representation Transform

A generalized approach to signal representation in Hilbert spaces involves the specification of a general transform called the *group representation transform*. It is a very general transform that has the wavelet and Gabor transforms as special cases. Feichtinger and Gröchenig ([FG89]) have developed this generalized approach in the setting of Banach spaces.

The *group representation transform* ([FG89]) of $f \in L^2(\mathbb{R})$ with respect to an admissible $g \in \mathcal{A}_{\Pi, \mu}(\mathcal{H})$ is a mapping $V_g : L^2(\mathbb{R}) \mapsto L^2_\mu(\mathcal{G})$ given as

$$(V_g f)(x) = \langle f, \Pi(x)g \rangle. \quad (4.14)$$

Discretizations to consider are of the form $\{(V_g f)(x)\}_{x \in \Gamma}$ where Γ is a countable subset of \mathcal{G} . It is expected that such samplings of the group representation transform will allow full recovery of f under certain density conditions on Γ . This is because the range $V_g(\mathcal{H})$ is a reproducing kernel Hilbert space. To see this, take $f, g \in \mathcal{H}$ and note that by Cauchy–Schwarz

$$|(V_g f)(x)| = |\langle f, \Pi(x)g \rangle| \leq \|f\| \|\Pi(x)g\| < \|f\| \|g\| \|\Pi(x)\| < M_x \|f\|,$$

where $M_x < \infty$ since $\Pi(x)$ is a bounded operator by definition. Thus, there is a pointwise bound of the form given in (4.13). From the discussion in Section 4.6.4 there is a reproducing kernel associated with V_g and the range $V_g(\mathcal{H})$ of V_g is a RKHS. Moreover, with the aid of the following theorem it is possible to write down explicitly the reproducing kernel for $V_g(\mathcal{H})$.

Theorem 4.10 ([GMP85]) Let Π be a representation of the group \mathcal{G} on the Hilbert space \mathcal{H} with left Haar measure μ . If Π is square integrable then there exists a unique self-adjoint positive operator $T : \mathcal{A}_{\Pi, \mu}(\mathcal{H}) \mapsto \mathcal{H}$ such that for all $g_1, g_2 \in \mathcal{A}_{\Pi, \mu}(\mathcal{H})$ and for all $f_1, f_2 \in \mathcal{H}$,

$$\langle V_{g_1} f_1, V_{g_2} f_2 \rangle_{L^2_\mu(\mathcal{G})} = \langle f_1, f_2 \rangle \langle Tg_1, Tg_2 \rangle.$$

As consequences of this theorem two corollaries are derived immediately. These corollaries may be thought of as counterparts of the Parseval and Plancherel relations. With T as the unique self-adjoint positive operator of Theorem 4.10 the positive scalar c_g is defined as $c_g = \|Tg\|$.

Corollary 4.11 (“Parseval”) If $g \in \mathcal{A}_{\Pi,\mu}(\mathcal{H})$ and $f_1, f_2 \in \mathcal{H}$ then

$$\langle V_g f_1, V_g f_2 \rangle_{L^2_\mu(\mathcal{G})} = c_g^2 \langle f_1, f_2 \rangle.$$

Corollary 4.12 (“Plancherel”) If $g \in \mathcal{A}_{\Pi,\mu}(\mathcal{H})$ and $f \in \mathcal{H}$ then

$$\|V_g f\|_{L^2_\mu(\mathcal{G})}^2 = c_g^2 \|f\|^2.$$

The following theorem establishes that the range of the group representation transform is a reproducing kernel and gives explicitly the form of the kernel. Asserted as a corollary is the fact that convolution with this kernel performs the orthogonal projection onto the range of the group representation transform.

Theorem 4.13 Suppose the hypothesis of Theorem 4.10 is satisfied and Π is a unitary representation; then $V_g(\mathcal{H})$ is a RKHS with kernel $\mathcal{K}(x, y) = T_y K(x)$ where

$$K(x) = \frac{1}{c_g^2} (V_g g)(x).$$

Proof: Let $F \in L^2_\mu(\mathcal{G})$.

$$\begin{aligned} (F * K)(x) &\stackrel{\triangle}{=} \int_{\mathcal{G}} F(y) K(y^{-1}x) d\mu(y) \\ &= \int_{\mathcal{G}} F(y) \left(\frac{1}{c_g^2} (V_g g)(y^{-1}x) \right) d\mu(y) \\ &= \frac{1}{c_g^2} \int_{\mathcal{G}} F(y) \langle g, \Pi(y^{-1}x)g \rangle d\mu(y) \\ &= \frac{1}{c_g^2} \int_{\mathcal{G}} F(y) \langle \Pi(y)g, \Pi(x)g \rangle d\mu(y) \\ &= \frac{1}{c_g^2} \int_{\mathcal{G}} F(y) \overline{V_g(\Pi(x)g)} d\mu(y) \\ &= \frac{1}{c_g^2} \langle F, V_g(\Pi(x)g) \rangle_{L^2_\mu(\mathcal{G})}. \end{aligned}$$

Orthogonally decomposing F with respect to $V_g(\mathcal{H})$ as $F = V_g f + H$, where $H \in V_g(\mathcal{H})^\perp$ for some $f \in \mathcal{H}$, yields

$$\begin{aligned} (F * K)(x) &= \frac{1}{c_g^2} \langle V_g f, V_g(\Pi(x)g) \rangle_{L^2_\mu(\mathcal{G})} \quad (H \text{ is nullified}) \\ &= \langle f, \Pi(x)g \rangle \quad (\text{by Theorem 4.10}) \\ &= V_g f(x). \end{aligned}$$

■

Corollary 4.14 *Convolution with the reproducing kernel K performs the orthogonal projection onto the range $V_g(\mathcal{H})$; that is,*

$$\forall F \in L^2(\mathcal{G}), \quad F * K = P_{V_g(\mathcal{H})}F.$$

Note that the kernel K is involutive; that is, $K = \tilde{K}$. To see this, without loss of generality let $c_g = 1$. Write

$$\begin{aligned}\tilde{K}(x) &= (V_g g)^{\sim}(x) &=& \overline{V_g g}(x^{-1}) \\ &=& \overline{\langle g, \Pi(x^{-1})g \rangle} \\ &=& \overline{\langle \Pi(x)g, g \rangle} \\ &=& \langle g, \Pi(x)g \rangle \\ &=& (V_g g)(x) = K(x).\end{aligned}$$

Thus, $(F * K)(x) = \langle F, T_x K \rangle$.

Gabor Transform

The Gabor transform may be written as

$$\begin{aligned}(G_g f)(t, \gamma) &=& \langle f, e_\gamma \tau_t g \rangle \\ &=& \langle f, \Pi(1, t, \gamma)g \rangle \\ &=& \langle f, \Pi(x)g \rangle,\end{aligned}$$

where $x = (1, t, \gamma) \in \mathcal{G}_H$, the Weyl–Heisenberg group, and $\Pi(z, t, \gamma) = z\tau_t e_\gamma g$, where $|z| = 1$ is a representation of \mathcal{G}_H on $L^2(\mathbb{R})$. This representation is denoted $\Pi = \Pi_H$. The Weyl–Heisenberg group action is given in Example 4.8. For the Gabor transform the associated left Haar measure is the product measure

$$d\mu(z, t, \gamma) = dz dt d\gamma.$$

Although the toral component z associated with the Weyl–Heisenberg group \mathcal{G}_H is necessary to properly define the representation Π_H it can effectively be ignored. To see this, first note that $\Pi_H(z, t, \gamma) = z\Pi_H(1, t, \gamma)$. Hence, for any $g \in L^2(\mathbb{R})$

$$\begin{aligned}\int_G \Pi(x)g d\mu(x) &=& \int_{\mathbf{T}} \int_{\mathbf{R}} \int_{\mathbf{R}} \Pi_H(z, t, \gamma)g dz dt d\gamma \\ &=& \int_{\mathbf{R}} \int_{\mathbf{R}} \int_{\mathbf{T}} zdz \Pi_H(1, t, \gamma)g dt d\gamma \\ &=& \int_{\mathbf{R}} \int_{\mathbf{R}} \Pi_H(1, t, \gamma)g dt d\gamma.\end{aligned}$$

Theorem 4.15 ([HW89, Proposition 3.2.4], [DGM86])

If $f, g \in L^2(\mathbb{R})$ then

$$\|G_g f\|_{L^2(\mathcal{G}_H)} = \|g\| \|f\|.$$

This theorem may be used to show that Π_H is a square integrable representation of \mathcal{G}_H on $L^2(\mathbb{R})$. Thus, Theorem 4.10 is applicable where $T = T_H = I$. All $g \in L^2(\mathbb{R})$ are admissible and G_g is a multiple of an isometry.

Wavelet Transform

For $s > 0$ the continuous wavelet transform $W_g f$ of a signal $f \in L^2(\mathbb{R})$ is

$$(W_g f)(t, s) = |s|^{1/2} \int f(x) \bar{g}(s(x-t)) dx.$$

Alternatively, the wavelet transform may be written as

$$\begin{aligned} W_g f(t, s) &= \langle f, \tau_t D_s g \rangle \\ &= \langle f, \Pi(t, s) g \rangle \\ &= \langle f, \Pi(x) g \rangle, \end{aligned}$$

where $x = (t, s) \in \mathcal{G}_A$, the affine group, and $\Pi(t, s) = \Pi_A(t, s) \stackrel{\triangle}{=} \tau_t D_s$ is a representation of $\mathcal{G}_A = \mathbb{R} \times \mathbb{R}^+$ on $L^2(\mathbb{R})$. The affine group action is given in Example 4.7. For the wavelet transform the associated left Haar measure is

$$d\mu(t, s) = dt ds.$$

It should be noted that if the dilation operator D_s were defined such that scale was inverse to frequency (as is more common), that is, $(D_s g)(t) = |s|^{-1/2} g(s^{-1}t)$, then the left Haar measure would be $d\mu(t, s) = s^{-2} dt ds$ instead. One benefit of choosing the former definition is exactly that it leads to a simpler left Haar measure and, therefore, simpler reconstruction formulae. Exercise 4.7 deals with these issues.

Theorem 4.16 ([HW89, Theorem 3.3.5],[GM84]) *If $f, g \in L^2(\mathbb{R})$ then*

$$\|W_g f\|_{L^2_\mu(\mathcal{G}_A)}^2 = \|f\|_{H_+^2}^2 \|T_A g\|_{H_+^2}^2 + \|f\|_{H_-^2}^2 \|T_A g\|_{H_-^2}^2,$$

where $T_A g = (|\gamma|^{-1/2} \widehat{g}(\gamma))^\vee$.

This theorem may be used to show that Π_A is a square integrable representation of \mathcal{G}_A on H_+^2 and H_-^2 . Thus, Theorem 4.10 is applicable. A similar result can be developed for real signals (c.f. Remark 2.1). By Theorem 4.16 if f is a real signal then

$$\|W_g f\|_{L^2_\mu(\mathcal{G}_A)} = \|f\| \|\frac{1}{2} T_A g\|.$$

If g is admissible then $c_g = \|T_A g\| < \infty$ and W_g is a multiple of an isometry.

Problems

- 4.1 Compute the Fourier transform of the Morlet wavelet and evaluate its second derivative at $\gamma = 0$. How is this value related to the Morlet wavelet's bandwidth γ_b ?
- 4.2 Prove that the CWT W_g with respect to a function $g \in L^2(\mathbb{R})$ is linear and time-invariant.
- 4.3 Show the following operator identities.
- $\mathcal{R}D_s = D_s\mathcal{R}$, and
 - $\mathcal{R} = \tau_t\mathcal{R}\tau_t$ for all $t \in \mathbb{R}$.
- 4.4 *Admissibility over $L^1(\mathbb{R})$.* Suppose that the function $f \in L^1(\mathbb{R})$ and has zero mean. Show that for all functions $g \in L^2(\mathbb{R})$ there is a function $h \in L^1(\mathbb{R})$ such that $W_g f = W_h f$. Conclude that all functions $g \in L^2(\mathbb{R})$ are admissible if the wavelet transform is restricted to operate only on zero-mean functions in $L^1(\mathbb{R})$.
- 4.5 Let W_g be the CWT with respect to the admissible wavelet g . Show the following.
- If $f \in H_+^2(\mathbb{R})$ and $g \in H_+^2(\mathbb{R})$ then $|W_g f(t, s)| = 0$ for all $s < 0$.
 - If $f \in H_+^2(\mathbb{R})$ and $g \in L^2(\mathbb{R})$ is purely real and even then
- $$W_g f(t, -s) = W_g f(t, s).$$
- (c) If $f \in H_+^2(\mathbb{R})$ and $g \in L^2(\mathbb{R})$ is purely imaginary and odd then
- $$W_g f(t, -s) = -W_g f(t, s).$$
- 4.6 Verify that the integral
- $$\int_{\mathbb{R}} |s|^{-1} |\widehat{g}(s^{-1}\gamma)|^2 ds$$
- is independent of s (viz. Equation (4.9)).
- 4.7 *Haar Measure.* With $x = (t, s)$, $y = (t_y, s_y)$, and $\Pi(x) = \tau_t D_s$ show that
- $$\int_{G_A} F(y^{-1}x) d\mu(x) = \int_{G_A} F(x) d\mu(x),$$
- where $d\mu(x) = dt ds$. More specifically show
- $$\int_{\mathbb{R}} \int_{\mathbb{R} \setminus \{0\}} F((t_y, s_y)^{-1} \cdot (t, s)) dt ds = \int_{\mathbb{R}} \int_{\mathbb{R} \setminus \{0\}} F(t, s) dt ds.$$
- Repeat for the case that $\Pi(x) = \tau_t D_{s^{-1}}$ and $d\mu(x) = s^{-2} dt ds$.

5

Discrete Wavelet Transform

In general, discrete wavelet transforms are generated by samplings (in the time-scale plane) of a corresponding continuous wavelet transform. Such a discrete wavelet transform is specified by the choice of items:

1. a time-scale sampling set (a countable set of points), and
2. an analyzing wavelet.

These choices cannot be made arbitrarily if the resulting transform is to satisfy some basic properties, namely, invertibility. Although an infinity of discrete wavelet transforms may be specified in this way, the term *discrete wavelet transform* (DWT) is commonly reserved to mean the one associated with a particular choice of sampling and a particular class of analyzing wavelets. For the DWT, time-scale sampling is restricted to be the standard dyadic lattice (see Figure 5.1) and the class of wavelets is restricted to be those that generate orthonormal (or biorthogonal) wavelet bases. Less restrictive discretizations lead to, in general, overcomplete wavelet transforms and are discussed in Chapter 6.

A main thread in the wavelet literature is dedicated to the idea of generating wavelet families that form orthonormal bases (ONBs); ([Mal89a], [Mal89b]), and, especially, those that generate compactly supported ONBs ([Dau92], [Dau88]). Because they allow for DWT implementation with finite impulse response (FIR) filters, wavelet orthonormal bases of compact support have been a focus of intense interest in signal processing. Underlying the DWT implementation (with compactly supported wavelets) is an efficient algorithm¹ called the fast wavelet transform (FWT). The FWT has a complexity that is $O(N)$ where N is the number of points in the signal to be analyzed.

This chapter reviews the theory of multiresolution analysis used for generating wavelet orthonormal bases and provides some examples of such

¹Although the main interest in this chapter is in orthogonal wavelets it should be noted that efficient $O(N)$ algorithms may still be formulated for biorthogonal wavelets ([CDF92]), and for some nonorthogonal bases as well ([Ald96], [AU93]).

bases. The fast wavelet transform (FWT) (pyramid algorithm) associated with the wavelet bases of compact support is also described and illustrated with several examples.

It should be noted that the discussion of the discrete wavelet theory presented here touches only on the key ideas underlying the discrete theory of orthonormal wavelets and that there are much more comprehensive treatments available, for example, [Dau92], [Wal94], [VK95], and [SN96].

5.1 Discretization of the CWT

The relationship between the CWT and DWT is made explicit in this section and related to orthonormal bases and sampling in the wavelet domain.

Recall from Section 4.3.1 that the continuous wavelet transform (CWT) of a one-dimensional function f is

$$W_g f(t, s) \triangleq \langle f, \tau_t D_s g \rangle.$$

Because the CWT is a two-dimensional function defined *continuously*² over the time-scale plane, the CWT cannot be computed using finite precision discrete machines. Approximation to the CWT, however, can be made to almost arbitrary precision through dense samplings of the time-scale plane. In this case, the sampling set is a countable set of discrete points contained in the time-scale plane. Each discrete set $\Gamma \triangleq \{(t_{m,n}, s_n)\} \subset \mathcal{G}_A$, specifies a countable set of wavelets $\{\tau_{t_{m,n}} D_{s_n} g\}$ that specifies, in turn, a particular discrete wavelet transform. In order for these transforms to have some important properties, (e.g. invertibility), it is clear that the sampling in the time-scale plane cannot be made arbitrarily. In fact, it is required that the underlying family of wavelets $\{\tau_{t_{m,n}} D_{s_n} g\}$ form a frame for a large enough space of interest.

Discretizations of this sort lead to, in general, overcomplete wavelet transforms and are discussed in all their generality in Chapter 6. In this chapter interest is focused on one discretization in particular which has a wealth of computational structure associated with it. Despite the fact that there is an infinity of possible discretizations of the CWT, the term *discrete wavelet transform* (DWT) is commonly used to mean the one associated with the dyadic sampling lattice

$$\Gamma_D \triangleq \{(2^{-n}m, 2^n)\}_{m,n \in \mathbb{Z}} \quad (5.1)$$

for certain analyzing wavelets that give rise to wavelet ONBs. The dyadic lattice Γ_D is depicted in Figure 5.1. This sampling set is special in the

²By this we mean that the domain of the CWT is simply connected.

sense that there exists a very fast algorithm for the computation of the CWT with respect to orthonormal wavelet families of compact support when restricted to the lattice Γ_D .

In summary, the term *discrete wavelet transform* (DWT) is used to indicate a special type of sampling of the CWT that satisfies the following (rather strict) conditions.

1. The time-scale discrete sampling set must be the dyadic grid Γ_D ;
2. the family of wavelets $\{\tau_t D_s g\}_{(t,s) \in \Gamma_D}$ must form an orthonormal basis for the space of interest; and
3. the analyzing wavelet must be compactly supported (hence, the entire family is compactly supported).

Under these conditions there is a fast (pyramid) algorithm for computing the DWT that requires filtering only with finite impulse response filters. This algorithm, the so-called fast wavelet transform, is discussed in Section 5.6.

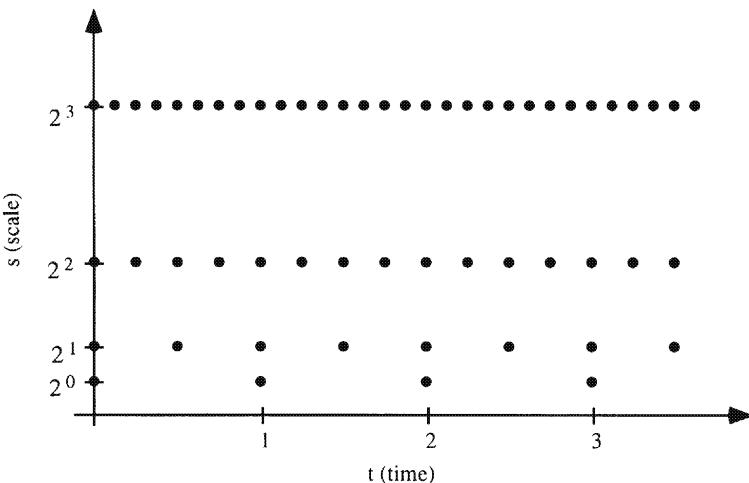


FIGURE 5.1. The dyadic lattice in the time-scale plane.

Note that the previously listed conditions ensure that there is a fast algorithm for the DWT computation and that there, in fact, exist algorithms of the same (low) complexity in cases where these strict conditions are relaxed, for example, [AA95].

5.2 Multiresolution Analysis

In this section, a mathematical framework for the construction of orthonormal wavelet bases in Hilbert spaces is presented.

Up until now, a very broad class of wavelet families, consisting of, in general, overcomplete elements, has been considered. An interesting and key factor leading to the proliferation of wavelet-based signal processing was the discovery and construction of *orthonormal* wavelet bases ([Mal89a]), and, especially, those that generate compactly supported ONBs ([Dau92], [Dau88]). Developed by Mallat ([Mal89b]), the main mathematical framework and mechanism by which the construction of orthonormal wavelet bases is facilitated is called multiresolution analysis (MRA). For generality, it is assumed that signals of interest come from an arbitrary Hilbert space \mathcal{H} of functions of a single variable that is usually a subset of the one-dimensional finite energy signal space $L^2(\mathbb{R})$. Although not difficult, generalization to functional Hilbert spaces of several variables is not addressed here.

5.2.1 Multiresolution Design

There are several key ingredients in a multiresolution analysis of a Hilbert space \mathcal{H} . First is a sequence of (increasing) nested spaces

$$\{0\} \subset \cdots V_{-1} \subset V_0 \subset V_1 \cdots \subset \mathcal{H}$$

that form the basic resolution structure. As its index k increases, the space V_k becomes more and more like \mathcal{H} . Second is a function

$$\phi \in \mathcal{H},$$

called the *scaling function*, whose integer translates $\{\tau_n \phi\}$ form an orthonormal basis for one of the subspaces V_k of \mathcal{H} . Without loss of generality, this space is chosen to be the subspace V_0 . Because they must be composed of only integer translations of ϕ , functions in V_0 are *coarser* than many other functions in \mathcal{H} in the sense that their variation over time is constrained³ by the variation in time of the function ϕ . Thus, V_0 may be thought of as a coarse or (low resolution) version of \mathcal{H} . Third, the larger subspaces should have higher resolution than the lower subspaces. For an arbitrary function $f \in \mathcal{H}$, the subspace V_k should contain functions that approximate f to some (level k) resolution. Because the V_k are chosen to be increasing subspaces of \mathcal{H} , larger values of k have corresponding subspaces V_k that contain higher resolution approximations of f .

³Formally, this may be expressed as $|\sum a_n \tau_n \phi| \leq \sum |a_n| |\tau_n \phi|$ (assuming that ϕ is differentiable).

5.2.2 Resolution and Dilation Invariance

A main idea behind a MRA is the specification of a sequence of subspaces $\{V_k : V_k \subseteq \mathcal{H}\}$ that are (uniformly) better and better approximations to \mathcal{H} as k increases. To achieve this goal the sequence of subspaces is required to satisfy

1. $V_k \subset V_{k+1}$, and
2. $V_k \rightarrow \mathcal{H}$ as $k \rightarrow \infty$ and $V_k \rightarrow \{0\}$ as $k \rightarrow -\infty$.

Under these assumptions each subspace V_k is a k th level approximation to the space \mathcal{H} . That is to say that the projection of a function onto the subspace V_k yields an approximation that approaches the projected function as k becomes large.

In addition, the increase in resolution should be uniform across subspaces; that is, stepping up to the next space V_{k+1} from V_k should give the same increase in resolution for all k . Although it is just one resolution subspace in the MRA framework, V_0 is special in that it is used as the reference space, albeit arbitrarily so, to which all other resolutions are related. One way to ensure a uniform increase in resolution across levels is to require that each subspace V_k have an ONB which is related to the ONB for V_0 in a consistent way. Precisely, the ONB for the space V_k is related to the ONB for V_0 by dilation by a power of 2. This leads to the additional constraint on the sequence $\{V_k\}$ that it be dilation (by a power of 2) invariant; that is,

$$3. f \in V_k \implies D_2 f \in V_{k+1}.$$

Explicitly, this means that if a function f_k is in V_k then the function $f_{k+1}(t) \stackrel{\triangle}{=} f_k(2t)$ must be in V_{k+1} . Thus, with

$$V_0 = \overline{\text{span}} \{ \tau_n \phi \}$$

and, in general,

$$\begin{aligned} V_k &\stackrel{\triangle}{=} \overline{\text{span}} \{ D_{2^k} \tau_n \phi \} \\ &= \overline{\text{span}} \{ \tau_{2^{-k} n} D_{2^k} \phi \}, \end{aligned}$$

the dilation invariance requirement is satisfied. In fact, the subspaces $\{V_k\}$ as defined previously are standard in MRAs.

It is important to note that the fact that $\{\tau_n \phi\}$ is an ONB for V_0 directly implies that $\{D_{2^k} \tau_n \phi\}$ is an ONB for V_k . This is because $\{D_{2^k} \tau_n \phi\}$ is an ONS in V_k ; that is,

$$\delta_{m,n} = \langle \tau_m \phi, \tau_n \phi \rangle = \langle D_{2^k} \tau_m \phi, D_{2^k} \tau_n \phi \rangle$$

and $V_k = \overline{\text{span}} \{ D_{2^k} \tau_n \phi \}$. Therefore, $\{D_{2^k} \tau_n \phi\}$ is an ONB for V_k by Theorem 3.2. Accordingly, functions f_k in the space V_k have an orthonormal

decomposition in terms of translated (by integer multiples of 2^{-k}) and dilated (by 2^k) versions of ϕ . Thus, for $k > 0$, functions in the space V_k have a higher resolution than those in V_0 . For the case of the Haar MRA, these points are illustrated in Figure 5.2 in Section 5.3.4.

5.2.3 Definition

A multiresolution analysis of a Hilbert space \mathcal{H} is formally defined in this section. It depends on the specification of a sequence of nested resolution subspaces $\{V_k\}$ of \mathcal{H} and a scaling function ϕ that satisfy some basic properties. The definition is stated formally as Definition 5.1.

Definition 5.1 (MRA) *Let $\{V_k : V_k \subset \mathcal{H}\}$ be an increasing sequence of subspaces and $\phi \in V_0$. The pair $(\{V_k\}, \phi)$ is said to be a multiresolution analysis of \mathcal{H} if*

1. *(Orthonormal basis for V_0) there is a function $\phi \in V_0$ such that $\{\tau_n \phi\}_{n \in \mathbb{Z}}$ is an orthonormal basis (ONB) for V_0 ,*
2. *(Dilation invariance) if $f \in V_k$ then $D_2 f \in V_{k+1}$, and*
3. *(Completeness) $\overline{\bigcup V_j} = \mathcal{H}$ and $\bigcap V_j = \{0\}$.*

These three defining conditions are motivated in the previous section. A MRA provides a mathematically well-founded structure that connects discrete time functions to continuous time functions. To see this connection, the underlying multiresolution representation is developed in the next section.

5.3 Multiresolution Representation

Much of the subsequent analysis and development can be succinctly described in terms of the representation operators discussed generally in Chapter 3. In the present case, interest is specialized to representations that are generated by the integer translations and power of two dilations of the scaling function ϕ and its soon-to-be described counterpart, the wavelet⁴ function ψ .

Under the assumption that $(\{V_k\}, \phi)$ is a MRA of \mathcal{H} , let $L_{\phi,k}$ denote the representation associated with the sequence of functions $\{D_{2^k} \tau_n \phi\}$ that spans the k th resolution subspace V_k . For an arbitrary function $f \in \mathcal{H}$, its k th resolution representation is

$$\begin{aligned} L_{\phi,k} f &\stackrel{\triangle}{=} \{\langle f, D_{2^k} \tau_n \phi \rangle\} \\ &= \left\{ (f * D_{2^k} \tilde{\phi})(2^{-k} n) \right\} \end{aligned}$$

⁴Analyzing wavelets that generate ONBs are commonly denoted ψ .

since $D_{2^k}\tau_n = \tau_{2^{-k}n}D_{2^k}$. Thus, the k th level representation of a function $f \in \mathcal{H}$ is a countable (discrete) sequence that is given by the regular samples (at the points $2^{-k}n$) of f passed through a filter having an impulse response which is $D_{2^k}\tilde{\phi}$.

Some useful mathematical observations are developed in the subsections that follow. Throughout the development f is an arbitrary function from \mathcal{H} and $(\{V_k\}, \phi)$ is the associated MRA.

5.3.1 Projection

Let $P_{V_k} : \mathcal{H} \mapsto V_k$ denote the orthogonal projection operator onto the space V_k . Since $\{D_{2^k}\tau_n\phi\}$ is an ONB for V_k the k th level resolution projection operator may be written as $P_{V_k} = L_{\phi,k}^* L_{\phi,k}$ (viz. Section 3.2.2, Theorem 3.2). In particular,

$$\begin{aligned} P_{V_0}f = L_{\phi,0}^* L_{\phi,0}f &= \sum \langle f, \tau_n\phi \rangle \tau_n\phi \\ &= \sum (f * \tilde{\phi})(n) \tau_n\phi \end{aligned}$$

and, more generally, the k th resolution subspace V_k has the projection

$$\begin{aligned} P_{V_k}f = L_{\phi,k}^* L_{\phi,k}f &= \sum_n \langle f, D_{2^k}\tau_n\phi \rangle D_{2^k}\tau_n\phi \\ &= \sum_n (f * D_{2^k}\tilde{\phi})(2^{-k}n) D_{2^k}\tau_n\phi. \end{aligned} \quad (5.2)$$

5.3.2 Fourier Transforms

Suppose $c = \{c_n\} \in \ell^2(\mathbb{Z})$ has the DTFT $C \triangleq \sum c_n e_{-n}$ (a 1-periodic function; viz. Section 2.6.3). Because they are integer translations of each other, the MRA atoms $\{\tau_n\phi\}$ have linear combinations that are determined by the Fourier transform of ϕ and the DTFT of the weights of the linear combination. Precisely,

$$\begin{aligned} (L_{\phi,0}^* c)^\wedge &= \left(\sum c_n \tau_n\phi \right)^\wedge \\ &= \sum c_n e_{-n} \cdot \hat{\phi} = C \cdot \hat{\phi}. \end{aligned}$$

This establishes an important connection between the continuous time domain and the discrete time domain in the MRA structure.

Consider now the Fourier transform of the projection $P_{V_k}f$ of f onto an arbitrary MRA subspace V_k given by Equation (5.3). This is determined by the Fourier transform of ϕ and the DTFT of $L_{\phi,k}f$ as

$$(P_{V_k}f)^\wedge = \sum_n (L_{\phi,k}f)_n D_{2^{-k}}e_{-n} \hat{\phi} = D_{2^{-k}} \left[(L_{\phi,k}f)^\wedge \cdot \hat{\phi} \right]. \quad (5.3)$$

From this equation, the interplay between periodic functions, for example, discrete time Fourier transforms (DTFTs), and continuous time functions, for example, the Fourier transform of ϕ , is seen to be an important component in the MRA structure. In fact, this interplay has a vital role in subsequent computational development. For $T > 0$, the relationship between inner products in $L^2(\mathbb{R})$ and $L^2(-T/2, T/2)$ has special utility and is given in Proposition 5.2.

Proposition 5.2 *If $G \in L^2(-T/2, T/2)$ is a T -periodic function and $f \in \mathcal{P}_T$ then*

$$\langle f, G \rangle_{L^2(\mathbb{R})} = \left\langle \sum_n \tau_{nT} f, G \right\rangle_{L^2(-T/2, T/2)}.$$

Proof:

$$\begin{aligned} \langle f, G \rangle_{L^2(\mathbb{R})} &= \int f \cdot \overline{G} \\ &= \sum_n \int_{nT}^{(n+1)T} f \cdot \overline{G} \\ &= \sum_n \int_0^T (\tau_{nT} f) \cdot \overline{G} \\ &= \int_0^T \left(\sum_n \tau_{nT} f \right) \cdot \overline{G} \\ &= \left\langle \sum_n \tau_{nT} f, G \right\rangle_{L^2(-T/2, T/2)}. \end{aligned}$$

■

Using this proposition it can be shown that the DTFT $(L_{\phi,0}f)^{\wedge}$ of the zeroth level representation of f is

$$(L_{\phi,0}f)^{\wedge} = \sum_n \tau_n \left(\widehat{f} \cdot \overline{\widehat{\phi}} \right). \quad (5.4)$$

To see this write

$$\begin{aligned} L_{\phi,0}f &\stackrel{\Delta}{=} \langle f, \tau_n \phi \rangle_{L^2(\mathbb{R})} \\ &= \left\langle \widehat{f}, e_{-n} \widehat{\phi} \right\rangle_{L^2(\mathbb{R})} \\ &= \left\langle \widehat{f} \cdot \overline{\widehat{\phi}}, e_{-n} \right\rangle_{L^2(\mathbb{R})} \\ &= \left\langle \sum_n \tau_n \left(\widehat{f} \cdot \overline{\widehat{\phi}} \right), e_{-n} \right\rangle_{L^2(-1/2, 1/2)} \end{aligned}$$

where Proposition 5.2 is invoked in the last equality. Since this last equality is the inverse DTFT in inner product form

$$L_{\phi,0}f = \{(f * \tilde{\phi})(n)\} \quad \text{and} \quad \sum_n \tau_n (\hat{f} \cdot \bar{\hat{\phi}})$$

must be DTFT pairs. Hence, Equation (5.4) is valid.

Equation (5.4) may be further used to characterize the fact that $\{\tau_n \phi\}$ is an orthonormal sequence in V_0 by substituting $f = \phi$ to yield

$$\begin{aligned} (L_{\phi,0}\phi)^{\wedge} &= \{\delta_{0,n}\}^{\wedge} \\ &= 1 = \sum_n \tau_n |\hat{\phi}|^2. \end{aligned}$$

Note that this is true for general functions ϕ that are not necessarily restricted to be members of V_0 . Since the DTFT of the Kronecker delta sequence is one almost everywhere the following well-known proposition has been proven.

Proposition 5.3 *Let $\phi \in L^2(\mathbb{R})$. The sequence of functions $\{\tau_n \phi\}$ is an orthonormal sequence if and only if*

$$\sum_n \tau_n |\hat{\phi}|^2 = 1 \quad \text{a.e.} \quad (5.5)$$

If, in addition, $(\{V_k\}, \phi)$ is a MRA of \mathcal{H} then $\{\tau_n \phi\}$ is an ONB for V_0 if and only if (5.5) holds.

Because any orthonormal sequence is complete for its span, (viz. Theorem 3.2 on page 32), and V_0 is the span of $\{\tau_n \phi\}$ in a MRA, imposition of the MRA structure leads to the stronger equivalent condition that $\{\tau_n \phi\}$ is an ONB for V_0 in Proposition 5.3.

5.3.3 Between Scale Relations

The relationship between functions in the k th resolution subspace V_k and the zeroth (reference) level resolution subspace V_0 reveals much of the structure associated with a MRA. Some of that structure is exposed in this section.

Consider the MRA representation $L_{\phi,0}$ for level 0 as it relates to the MRA representation $L_{\phi,k}$ for level k . Since for any $f \in \mathcal{H}$

$$\begin{aligned} L_{\phi,k}f &= \{\langle f, D_{2^k} \tau_n \phi \rangle\} \\ &= \{\langle D_{2^{-k}} f, \tau_n \phi \rangle\} = L_{\phi,0} D_{2^{-k}} f \end{aligned}$$

the MRA representation for level k is related to the MRA representation for level 0 as

$$L_{\phi,k} = L_{\phi,0} D_{2^{-k}}.$$

This, in turn, leads to the intrascale projection operator relationship of

$$P_{V_k} = L_{\phi,k}^* L_{\phi,k} = D_{2^k} L_{\phi,0}^* L_{\phi,0} D_{2^{-k}} = D_{2^k} P_{V_0} D_{2^{-k}}.$$

As a consequence of the nested hierarchy of subspaces associated with a MRA, a key relationship between the Fourier transform of a V_1 function, the Fourier transform of the scaling function, and the DTFT of the first-level representation, that is, the inner product representation with respect to $\{D_2 \tau_n \phi\}$, results. To see this, let $f \in V_1$ so that

$$f = P_{V_1} f.$$

In the Fourier domain this becomes (from Equation (5.3))

$$\widehat{f} = (P_{V_1} f)^\wedge = D_{2^{-1}} \left((L_{\phi,1} f)^\wedge \cdot \widehat{\phi} \right).$$

This result is stated formally as Proposition 5.4.

Proposition 5.4 *If $f \in V_1$ then*

$$\widehat{f} = D_{2^{-1}} \left(F \cdot \widehat{\phi} \right),$$

where F is the DTFT of the sequence $L_{\phi,1} f = \{\langle f, D_2 \tau_n \phi \rangle\}$.

Applied to ϕ itself (this is valid since it is a member of $V_0 \subset V_1$), Proposition 5.4 indicates the recursive structure that $\widehat{\phi}$ must necessarily satisfy in order to generate a valid MRA. Specifically,

$$\widehat{\phi} = D_{2^{-1}} \left(H \cdot \widehat{\phi} \right), \quad (5.6)$$

where $H = (L_{\phi,1} \phi)^\wedge$ is the DTFT of the V_1 resolution representation of ϕ . In the time domain this relationship is called the *dilation equation* ([HC93]) and describes how ϕ is composed of dilated by 2 and unit-translated versions of itself; that is,

$$\phi = P_{V_1} \phi = \sum \langle \phi, D_2 \tau_n \phi \rangle D_2 \tau_n \phi. \quad (5.7)$$

Using Equation (5.6) the familiar infinite product expansion of $\widehat{\phi}$ may be derived. To see the product expansion, let \check{H} be the multiplicatively scaled version of H given by

$$\check{H} \stackrel{\Delta}{=} 2^{-1/2} H \quad (5.8)$$

so that Equation (5.6) becomes

$$\widehat{\phi}(\gamma) = \check{H}(\gamma/2) \cdot \widehat{\phi}(\gamma/2).$$

Substituting this expression for $\widehat{\phi}$ recursively into itself yields

$$\begin{aligned}
 \widehat{\phi}(\gamma) &= \check{H}(2^{-1}\gamma) \widehat{\phi}(2^{-1}\gamma) \\
 &= \check{H}(2^{-1}\gamma) \check{H}(2^{-2}\gamma) \widehat{\phi}(2^{-2}\gamma) \\
 &= \check{H}(2^{-1}\gamma) \check{H}(2^{-2}\gamma) \check{H}(2^{-3}\gamma) \widehat{\phi}(2^{-3}\gamma) \\
 &\vdots \\
 &= \widehat{\phi}(2^{-N}\gamma) \cdot \prod_{k=1}^N \check{H}(2^{-k}\gamma), \quad N = 1, 2, 3, \dots \\
 &= \lim_{N \rightarrow \infty} \widehat{\phi}(2^{-N}\gamma) \cdot \prod_{k=1}^N \check{H}(2^{-k}\gamma).
 \end{aligned} \tag{5.9}$$

In general, this limit need not exist and its convergence properties have been analyzed in various ways (viz. [Dau92],[HC93]). Relevant questions regarding the infinite product include the following. Does the product converge and, if so, in what sense, and to what? In addition, what are the smoothness properties associated with the limit function? A simple sufficient condition ([Mal89a]) ensuring the convergence of the product in $L^2(\mathbb{R})$ is

$$\sup_{\gamma \in [-1/4, 1/4]} |H(\gamma)| > 0,$$

and there are others; see [Dau92, Chapter 6, p. 182] for an in-depth discussion. It is clear, however, that in order for the infinite product to converge in *any* sense to a nontrivial limit all but a finite number of product terms must tend to 1. This further implies that

$$\lim_{k \rightarrow \infty} \check{H}(2^{-k}\gamma) = \check{H}(0) = 1,$$

where \check{H} is assumed continuous around zero. Under proper convergence assumptions and because $\widehat{\phi}(0) = 1$ (demonstrated in Proposition 5.9), the Fourier transform $\widehat{\phi}$ of the scaling function ϕ is

$$\widehat{\phi}(\gamma) = \prod_{k=1}^{\infty} \check{H}(2^{-k}\gamma), \tag{5.10}$$

where $\check{H}(0) = 1$ (or $H(0) = \sqrt{2}$). Moreover, it is shown that the condition $H(0) = \sqrt{2}$ implies that $H(\frac{1}{2}) = 0$ (viz. Equation (5.14)).

5.3.4 Haar MRA

To illustrate many of the ideas behind a MRA it is instructive and customary to consider the special case of the Haar scaling function $\phi_{\text{Haar}} \stackrel{\Delta}{=} 1_{(0,1]}$.

In this case it is easy to verify that the integer translations of ϕ form an orthonormal sequence and therefore an ONB for $V_0 = \overline{\text{span}}\{\tau_n\phi\}$. In the Haar MRA, functions in V_0 are constrained to be constant over the intervals $(n, n+1)$. Nevertheless, functions $f_0 \in V_0$ may be used to approximate arbitrary functions in $f \in \mathcal{H}$ in the obvious way

$$\begin{aligned} f_0 &= \sum_n \langle f, \tau_n\phi \rangle \tau_n\phi \\ &= \sum_n c_n 1_{(n,n+1]}, \end{aligned}$$

where c_n is the average value of the function f over the interval $(n, n+1]$. Because $\{\tau_n\phi\}$ is an orthonormal basis for V_0 , the preceding equation is the orthogonal projection of f onto V_0 ; that is,

$$f_0 = P_{V_0} f.$$

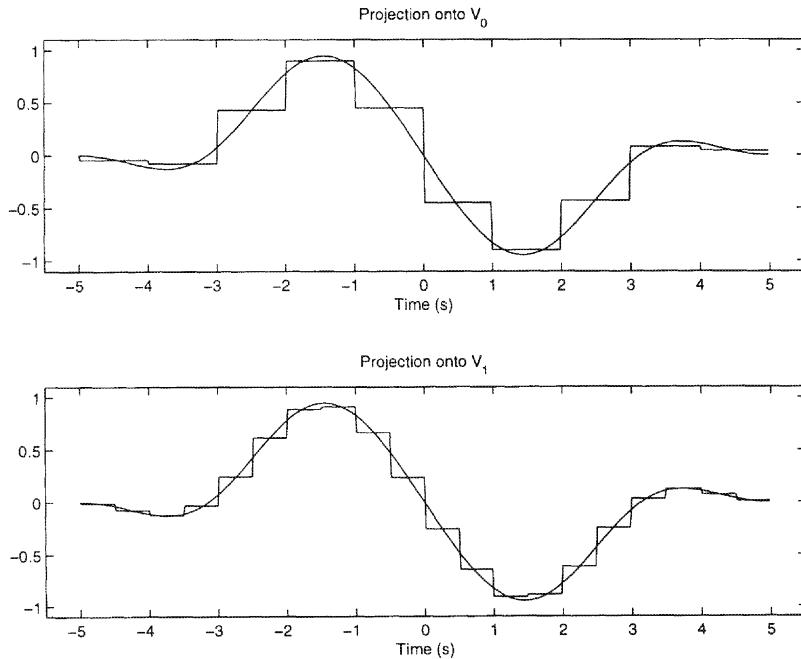


FIGURE 5.2. Two different resolutions in the Haar MRA.

Figure 5.2 shows the approximation of a smooth $L^2(\mathbb{R})$ function by its projection onto the V_0 and V_1 subspaces associated with the Haar MRA. Note that the approximation in V_1 is higher resolution since functions in

V_1 are piecewise constant over the half-length intervals $(n/2, (n+1)/2)$ whereas the functions in V_0 are piecewise constant over the unit length intervals $(n, n+1)$.

For the Haar scaling function the dilation equation (5.7) is readily verified to be

$$\phi = \frac{1}{\sqrt{2}} D_2 \phi + \frac{1}{\sqrt{2}} D_2 \tau_1 \phi$$

so that

$$(L_{\phi,1}\phi)_n = \begin{cases} \frac{1}{\sqrt{2}}, & n = 0, 1 \\ 0, & \text{else.} \end{cases}$$

Figure 5.3 illustrates this dilation equation in picture form.

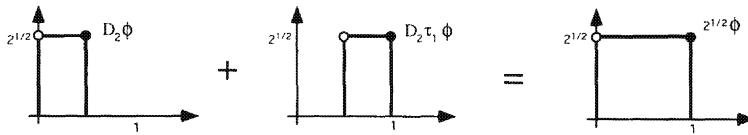


FIGURE 5.3. A graphical depiction of the dilation equation for the Haar MRA.

Computing the DTFT of $L_{\phi,1}\phi$ gives

$$H = (L_{\phi,1}\phi)^\wedge = \frac{1}{\sqrt{2}}(1 + e_{-1})$$

and

$$\check{H} = \frac{1}{2}(1 + e_{-1}).$$

Note, in particular, that $\check{H}(0) = 1$ and all the properties listed at the end of Section 5.4.3 hold.

5.4 Orthonormal Wavelet Bases

Using the MRA framework previously described, a wide variety of families of orthonormal wavelets may be generated for a Hilbert space \mathcal{H} . Already, almost all the elements required of an orthonormal wavelet basis have been realized in the MRA framework. Namely, sets of unit norm functions that are related by the translations and dilations of a single generating function (the scaling function) have been specified which form orthonormal bases for the resolution subspaces V_k . In order to generate a wavelet ONB for all of \mathcal{H} , however, orthogonality between resolutions is also required. Since one is completely contained in the other, the subspaces $\{V_k\}$ clearly are not orthogonal to each other. Thus, the direct combination (union) of the

individual ONBs that span each subspace V_k is not an overall wavelet ONB for the whole space \mathcal{H} .

The main idea used to achieve orthogonality between resolutions in a MRA is to define an auxiliary sequence of subspaces $\{W_k : W_k \subseteq \mathcal{H}\}$ complementary to V_k and orthogonal to V_{k+1} for each k that characterizes the differences between V_k and V_{k+1} . These are the wavelet subspaces. Specifically, to characterize the increase in resolution, the wavelet subspace W_k is defined to be the orthogonal complement of the subspace V_k in the next higher resolution space V_{k+1} ; that is,

$$V_{k+1} = V_k \oplus W_k \quad \text{and} \quad V_k \perp W_k.$$

In this way, the wavelet subspaces W_k describe exactly those wavelet functions that need be added to a set of atoms spanning V_k so that the resulting combination spans V_{k+1} . Without loss of generality, consider only the wavelet subspace W_0 . The goal is to construct a function $\psi \in W_0 \subset V_1$ with the property that $\{\tau_n \psi\}$ forms an ONB for W_0 . If such a ψ were found then $\{D_{2^m} \tau_n \psi\}_{m,n \in \mathbb{Z}}$ would form an ONB for the entire space \mathcal{H} . This is because (i) each subspace W_k is the k th resolution wavelet subspace of \mathcal{H} (i.e., for a fixed value of k , $\{D_{2^k} \tau_n \psi\}$ is an ONB for W_k), and (ii) the wavelet subspaces are orthogonal to each other so that

$$V_{k+1} = V_k \oplus W_k = V_{k-1} \oplus W_{k-1} \oplus W_k \dots = V_J \oplus \bigoplus_{j=J}^k W_j,$$

for any integer $J \leq k$. Because of the completeness property of the MRA spaces $\{V_k\}$, namely, $\lim_{j \rightarrow \infty} V_j = \mathcal{H}$ and $\lim_{j \rightarrow -\infty} V_j = \{0\}$, \mathcal{H} may be written as the direct sum of all the wavelet subspaces as

$$\mathcal{H} = \dots W_{-1} \oplus W_0 \oplus W_1 \dots = \bigoplus_{j=-\infty}^{\infty} W_j.$$

5.4.1 Characterizing W_0

The wavelet subspace W_0 may be characterized in terms of the zero and first-level representation operators $L_{\phi,0}$ and $L_{\phi,1}$. Specifically, a nontrivial function ψ is in the subspace W_0 if and only if

- (i) $L_{\phi,0} \psi = 0$, and
- (ii) $L_{\phi,1} \psi \neq 0$.

These conditions correspond to the defining requirements of W_0 that (i) $\psi \perp V_0$ and (ii) $\psi \in V_1$ and lead to an equivalent characterization of W_0 involving the DTFTs of $L_{\phi,1}\phi$ and $L_{\phi,1}\psi$. This characterization is now developed.

Reiterating the theme of Proposition 5.4, the following expansions for the scaling function ϕ and a possible wavelet ψ are given.

Like the scaling function ϕ , the candidate analyzing wavelet ψ is a member of V_1 so that there are unique sequences $\{h_n\} = L_{\phi,1}\phi$ and $\{g_n\} = L_{\phi,1}\psi$ such that

$$\phi = P_{V_1}\phi = \sum h_n D_2 \tau_n \phi \quad \text{and} \quad \widehat{\phi} = D_{2^{-1}}(H\widehat{\phi}) \quad (5.11)$$

and

$$\psi = P_{V_1}\psi = \sum g_n D_2 \tau_n \phi \quad \text{and} \quad \widehat{\psi} = D_{2^{-1}}(G\widehat{\phi}). \quad (5.12)$$

It is shown that $\psi \in W_0$ if and only if the DTFTs G and H of the level-1 representation coefficient sequences $L_{\phi,1}\psi = \{g_n\}$ and $L_{\phi,1}\phi = \{h_n\}$ satisfy the following properties.

- (i) $|H|^2 + \tau_{1/2}|H|^2 = 2$, and
- (ii) $G\overline{H} + \tau_{1/2}(G\overline{H}) = 0$.

Since these conditions characterize the subspace W_0 , any candidate wavelet ψ (for a given scaling function ϕ) must also satisfy them. Proposition 5.5 provides the mathematical basis by which these conditions are derived.

Proposition 5.5 Suppose that $(\{V_k\}, \phi)$ is a MRA for the Hilbert space \mathcal{H} . If $f \in V_1$, $F = (L_{\phi,1}f)^\wedge$, and $H = (L_{\phi,1}\phi)^\wedge$ then

$$2^{1/2}D_2(L_{\phi,0}f)^\wedge = F\overline{H} + \tau_{1/2}(F\overline{H}). \quad (5.13)$$

Proof: Since $f \in V_1$ by Proposition 5.4

$$\begin{aligned} (L_{\phi,0}f)^\wedge &= \sum_k \tau_k (\widehat{f} \cdot \overline{\widehat{\phi}}) \\ &= \sum_k \tau_k (D_{2^{-1}}(F\widehat{\phi}) \cdot \overline{D_{2^{-1}}(H\widehat{\phi})}) \\ &= 2^{-1/2} \sum_k \tau_k D_{2^{-1}}(F\overline{H} \cdot |\widehat{\phi}|^2) \\ &= 2^{-1/2} D_{2^{-1}} \sum_k \tau_{k/2} (F\overline{H} \cdot |\widehat{\phi}|^2) \\ &= 2^{-1/2} D_{2^{-1}} \sum_k \tau_k (F\overline{H} \cdot |\widehat{\phi}|^2) + \sum_k \tau_{k+1/2} (F\overline{H} \cdot |\widehat{\phi}|^2) \end{aligned}$$

so that

$$\begin{aligned} 2^{1/2}D_2(L_{\phi,0}f)^\wedge &= \sum_k (F\overline{H} \cdot |\tau_k \widehat{\phi}|^2) + \sum_k \tau_{1/2} (F\overline{H} \cdot |\tau_k \widehat{\phi}|^2) \\ &= F\overline{H} + \tau_{1/2}(F\overline{H}). \end{aligned}$$

■

When applied to the scaling function ϕ and the candidate wavelet ψ , Proposition 5.5 gives directly the two conditions listed previously; that is,

$$2 = |H|^2 + \tau_{1/2}|H|^2 \quad (5.14)$$

and

$$0 = G\bar{H} + \tau_{1/2}(G\bar{H}). \quad (5.15)$$

Clearly, $L_{\phi,0}\phi = \{\delta_{n,0}\}$ (since $\{\tau_n\phi\}$ is an ONB for V_0) and $L_{\phi,0}\psi = 0$ (since $\psi \perp V_0$).

5.4.2 Wavelet Construction

Equations (5.14) and (5.15) developed in the previous section provide a characterization of the subspace W_0 . As such, they may be used to help find a candidate wavelet $\psi \in W_0$ whose integer translates form an ONB for W_0 . To find such a ψ it is enough to find the corresponding 1-periodic function G that is related to ψ via Equation (5.12).

A key idea is to recognize ([Dau92]) that a suitable G has the form

$$G = Q\tau_{1/2}\bar{H}, \quad \text{with a corresponding} \quad \check{G} = Q\tau_{1/2}\bar{H}, \quad (5.16)$$

for some 1-periodic function $Q \in L^2[0, 1]$. Additionally, it is required that Q be an all-pass filter (i.e., $|Q| = 1$) so that with this choice for G , Equation (5.14) implies that

$$|H|^2 + |G|^2 = 2. \quad (5.17)$$

Further substituting G into Equation (5.15) yields

$$[Q + \tau_{1/2}Q](\tau_{1/2}\bar{H})\bar{H} = 0.$$

Thus, if Q is chosen such that

$$Q + \tau_{1/2}Q = 0, \quad |Q| = 1, \quad (5.18)$$

then Equation (5.15) will be satisfied. If $\{q_n\}$ is the inverse DTFT sequence associated with Q then this means that

$$q_n + (-1)^n q_n = 0.$$

Consequently, this sequence must be zero at all even indices; that is,

$$q_{2n} = 0, \quad \forall n \in \mathbb{Z}.$$

Picking G as in Equation (5.16) means in the discrete time domain that

$$g_n = \{q_n\} * \{(-1)^n \bar{h}_{-n}\}.$$

The simplest possible choice for $\{q_n\}$ is $q_n = \delta_{n,1}$ which corresponds to choosing $Q = e_{-1}$ so that

$$g_n = (-1)^{(n-1)} \bar{h}_{1-n}, \quad (5.19)$$

and this is the standard choice for the discrete wavelet filter g . As a result, the discrete filters h and g have a 1–1 correspondence so that in the case where h is of finite length N then so is g .

Thus far, a candidate wavelet ψ has been developed that is a member of W_0 (it is orthogonal to V_0 since it satisfies (5.15) and (5.14), and it is a member of $L^2(\mathbb{R})$ because ϕ is). In the Fourier domain the candidate wavelet is

$$\widehat{\psi} = D_{2^{-1}}(G \cdot \widehat{\phi}) = D_{2^{-1}}(e_1 \tau_{1/2} \bar{H} \cdot \widehat{\phi}).$$

It remains to be seen that the integer translates of this ψ form an orthonormal basis for W_0 . That this is true is the content of Theorem 5.6.

Because of the MRA structure, dilating this ONB by integer powers of two gives ONBs for the other resolution subspaces and, moreover, the union of all these ONBs is an ONB for all of \mathcal{H} . This fact is stated formally as Theorem 5.7.

Theorem 5.6 (ONB for W_0) *Let $(\{V_k\}, \phi)$ be a MRA for the Hilbert space \mathcal{H} , $H \triangleq (L_{\phi,1}\phi)^\wedge$, and $G \triangleq (L_{\phi,1}\psi)^\wedge$. If G is chosen as in Equation (5.16) with Q as in (5.18) then*

$$\{\tau_n \psi\} \text{ is an ONB for } W_0,$$

where $\widehat{\psi} = D_{2^{-1}}(G \cdot \widehat{\phi})$.

Proof: First, it is shown that the sequence of functions $\{\tau_n \psi\}$ forms an orthonormal sequence (ONS) and then, second, that it is complete in W_0 .

1. (ONS for W_0) By Proposition 5.3 it is sufficient for the first part to show that $\sum \tau_n |\widehat{\psi}|^2 = 1$. By a straightforward computation

$$\begin{aligned} \sum \tau_n |\widehat{\psi}|^2 &= \sum \tau_n 2^{-1/2} D_{2^{-1}}(|G|^2 |\widehat{\phi}|^2) \\ &= 2^{-1/2} D_{2^{-1}} \sum \tau_{n/2} |G|^2 |\widehat{\phi}|^2 \\ &= 2^{-1/2} D_{2^{-1}} \left[\sum \tau_n |G|^2 |\widehat{\phi}|^2 + \tau_{1/2} \tau_n |G|^2 |\widehat{\phi}|^2 \right] \\ &= 2^{-1/2} D_{2^{-1}} \left[|G|^2 \sum \tau_n |\widehat{\phi}|^2 + \tau_{1/2} |G|^2 \sum \tau_n |\widehat{\phi}|^2 \right] \\ &= 2^{-1/2} D_{2^{-1}} [|G|^2 + |H|^2] = 1/2 \cdot 2 = 1, \end{aligned}$$

since $\tau_{1/2} |G| = |H|$ and (5.14) holds.

2. (Complete in W_0) By Theorem 3.2 the ONS $\{\tau_n\psi\}$ is an ONB for W_0 if and only if $L_{\psi,0}$ is injective over W_0 ; that is,

$$\forall f \in W_0, \quad L_{\psi,0}f = 0 \implies f = 0.$$

Take $f \in W_0$ such that

$$L_{\psi,0}f = 0$$

and let $F = L_{\phi,1}f$. Because $W_0 \perp V_0$ and both $\psi \in W_0$ and $f \in W_0$,

$$L_{\phi,0}f = 0 \quad \text{and} \quad L_{\phi,0}\psi = 0.$$

These three representation equations lead to the three quadrature relations (viz. Proposition 5.5),

$$L_{\phi,0}f = 0 \implies F\bar{H} = -\tau_{1/2}F\bar{H}, \quad (5.20)$$

$$L_{\phi,0}\psi = 0 \implies G\bar{H} = -\tau_{1/2}G\bar{H}, \quad (5.21)$$

and

$$L_{\psi,0}f = 0 \implies F\bar{G} = -\tau_{1/2}F\bar{G}. \quad (5.22)$$

Multiplying (5.20) and the conjugate of (5.21) gives

$$\begin{aligned} F\bar{G}|H|^2 &= \tau_{1/2}(F\bar{G}|H|^2) \\ &= \tau_{1/2}(F\bar{G})|G|^2 \\ &= -F\bar{G}|G|^2, \end{aligned}$$

where the last equality follows from (5.22). This further yields

$$F\bar{G}(|H|^2 + |G|^2) = 0 \implies F\bar{G} = 0$$

since $|H|^2 + |G|^2 = 2$. Similarly, combining (5.21), (5.22), and then (5.20) in the same way yields

$$F\bar{H}(|H|^2 + |G|^2) = 0 \implies F\bar{H} = 0.$$

Hence

$$0 = |F\bar{H}|^2 + |F\bar{G}|^2 = |F|^2(|H|^2 + |G|^2)$$

which implies that $F = 0$ a.e. Since $\widehat{f} = D_{2^{-1}}(F\widehat{\phi}) = 0$ then $f = 0$. This proves that $L_{\psi,0}$ is injective. ■

As a consequence of the imposed MRA structure and Theorem 5.6, the complete wavelet family generated by ψ , that is, the set of integer powers of 2-dilations and integer translates of the wavelet ψ , yields an orthonormal basis for the entire space \mathcal{H} . This fact is stated explicitly as Theorem 5.7.

Theorem 5.7 (Wavelet ONB for \mathcal{H}) *With the assumptions of Theorem 5.6 the set of functions*

$$\{D_{2^m}\tau_n\psi\} \text{ is an ONB for } \mathcal{H}.$$

5.4.3 The Scaling Function

In order for $(\{V_k\}, \phi)$ to be a MRA for \mathcal{H} the scaling function ϕ must be necessarily highly structured and satisfy some specific properties. Although some of this structure has already been exposed, as seen in Section 5.3.3, there is still more that has not yet been discussed. In this section, some further properties of the scaling function and its DTFT $H = \{h_n\}^\wedge$ are presented. These properties are summarized in Table 5.1.

Consider Equation (5.14),

$$|H|^2 + \tau_{1/2}|H|^2 = 2.$$

Integrating both sides of this equation over a unit length interval gives directly that

$$\|H\|_{L^2(-1/2, 1/2)}^2 = 1,$$

so that $\|h\|_{\ell^2(\mathbb{Z})}^2 = 1$ or

$$\sum |h_n|^2 = 1.$$

More generally, taking the inverse DTFT of Equation (5.14) yields

$$(h * \tilde{h})_n + (-1)^n(h * \tilde{h})_n = 2\delta_{0,n}$$

so that $(h * \tilde{h})_{2n} = 0$. Explicitly, this means that the scaling function filter $\{h_n\}$ is orthogonal to its even integer shifts; that is,

$$\sum_k h_k \bar{h}_{k-2n} = \delta_{0,n}, \quad n \in \mathbb{Z}.$$

Proposition 5.8 *If $(\{V_k\}, \phi)$ is a MRA for \mathcal{H} then*

$$\forall n \in \mathbb{Z}, \quad \widehat{\phi}(n) = \delta_{0,n}.$$

Proof: Since ϕ comes from a valid MRA the finite product expansion (5.9) holds for N arbitrarily large. The proof relies on the fact that $H(1/2) = 0$ and that for integers $n \neq 0$ the product expansion has some term $H(2^{-k}n) = H(1/2) = 0$.

1. ($0 < n < \infty$) Expanding the integer n in base 2 gives $n = \sum_{j=0}^J a_j 2^j$ for some finite J with a_j being the binary expansion coefficients (either 0 or 1). If k is the smallest index of all the nonzero coefficients (i.e., $k = \min \{j : a_j = 1\}$) then

$$2^{-(k+1)}n = \sum_{j=0}^J a_j 2^{j-k-1} = \underbrace{\sum_{j=0}^k a_j 2^{j-k-1}}_{1/2} + \underbrace{\sum_{j=k+1}^J a_j 2^{j-k-1}}_{\text{integer} \geq 1}.$$

Since H is 1-periodic this shows that $H(2^{-(k+1)}n) = H(1/2) = 0$. Thus, by (5.10), $\widehat{\phi}(n) = 0$.

2. ($-\infty < n < 0$) This case is analogous to the preceding case since

$$H(-2^{-(k+1)}n) = H\left(-\frac{1}{2}\right) = H\left(\frac{1}{2}\right) = 0.$$

3. ($n = 0$) Since $\phi(n) = 0$ for all $n \neq 0$ the orthonormality of $\{\tau_n\phi\}$ gives, by Proposition 5.4, that

$$1 = \sum_n \tau_n |\widehat{\phi}(0)|^2 = \sum_n |\widehat{\phi}(n)|^2 = |\widehat{\phi}(0)|^2.$$

If ϕ is real then this together with the fact that $\check{H}(0) = 1$ and (5.10) gives $\widehat{\phi}(0) = 1$. ■

ϕ	$\{h_n\}$	$H = \{h_n\}^\wedge$
$\widehat{\phi}(n) = \delta_{n,0}$ $\int \phi ^2 = 1$ $\int \phi = 1$ $\sum \tau_n \widehat{\phi} ^2 = 1$ a.e. $\sum \tau_n \phi = 1$ a.e.	$\sum_k h_k = \sqrt{2}$ $\sum_k h_k ^2 = 1$ $\sum_k h_k \bar{h}_{k-2n} = \delta_n$	$H(0) = \sqrt{2}$ $H(1/2) = 0$ $\int_0^1 H ^2 = 1$ $ H ^2 + \tau_{1/2} H ^2 = 2$

TABLE 5.1. Scaling function and related filter properties for orthonormal wavelet bases.

Proposition 5.9 *If $\forall n \in \mathbb{Z}$, $\widehat{\phi}(n) = \delta_{0,n}$ then $\sum \tau_n \phi = 1$.*

Proof: Apply Proposition 5.2 to $\delta_{0,m} = \widehat{\phi}(m) = \langle \phi, e_m \rangle$ to get $\delta_{0,m} = \langle \sum_n \tau_n \phi, e_m \rangle_{L^2(-1/2, 1/2)}$. Thus by the inverse DTFT, $\sum_n \tau_n \phi = 1$, a.e. ■

In summary, the scaling function ϕ associated with a valid MRA and its corresponding time domain filter h and frequency domain filter H satisfy all the properties listed in Table 5.1.

5.5 Compactly Supported (Daubechies) Wavelets

With the formulation of the MRA theory, an elegant mechanism for generating a wide variety of wavelet ONBs is now available. Once a scaling function ϕ is found that constitutes a valid MRA (viz. Definition 5.1 and

Equation (5.14)) an appropriate analyzing wavelet ψ is readily determined via (5.16). In general, wavelets generated in this fashion need not be compactly supported, that is, be zero off some finite interval. It is natural then to ask if there are wavelet orthonormal bases of compact support. Of course, the answer to this question is decidedly “Yes.”

Compactly supported bases of orthonormal wavelets were first developed by Daubechies in 1988 ([Dau88], [Dau92]). The existence and construction of orthonormal wavelet bases with compact support is a remarkable and important development of wavelet theory. In this section, the construction of such bases is presented.

5.5.1 Main Idea

In order that a wavelet ONB be compactly supported, it is necessary that the generating wavelet ψ be of compact support itself. Since ψ is composed of unit-translated versions of the scaling function ϕ as

$$\psi = \sum g_n D_2 \tau_n \phi,$$

it is clear that the following requirements are sufficient to ensure that ψ have compact support.

1. ϕ has compact support, and
2. the wavelet filter coefficients $\{g_n\}$ are nonzero for only a finite number of indices.

These two conditions are not independent; in fact, Condition 1 actually implies Condition 2 (Exercise 5.6) when the wavelet filter coefficients $\{g_n\}$ are chosen by the 1–1 correspondence with the scaling filter coefficients $\{h_n\}$ given in (5.19).

Generating a compactly supported wavelet ONB requires the construction of a *finite* length coefficient sequence $\{h_n\}$ that satisfies (5.14); that is,

$$|H|^2 + \tau_{1/2} |H|^2 = 2.$$

Filters $H = \{h_n\}^\wedge$ that satisfy these requirements are constructed in the following subsections.

5.5.2 Trigonometric Half-Band Filters

The task at hand is to find a finite length sequence $\{h_n\}$, that is, one that has only a finite number of nonzero members, which satisfies the half-band filter requirement of Equation (5.14). Being the DTFT of a length $N < \infty$ sequence, $H = \{h_n\}^\wedge$ is a trigonometric polynomial of the form

$$H = \sum_{n=0}^{N-1} h_n e^{-jn\omega_0}.$$

It turns out ([Kai94], [Str93]) that such sequences may be derived from the binomial expansion⁵ of $\cos^2 + \sin^2$. More specifically, let $c(\gamma) = \cos(\pi\gamma)$ and $s(\gamma) = \sin(\pi\gamma)$ and note that $\tau_{1/2}c^2 = s^2$ and $\tau_{1/2}s^2 = c^2$. With $M \triangleq 2N - 1$ write

$$\begin{aligned} 1 &= (c^2 + s^2)^M \\ &= \sum_{k=0}^M \binom{M}{k} s^{2k} c^{2(M-k)} \\ &= \sum_{k=0}^{N-1} \binom{M}{k} s^{2k} c^{2(M-k)} + \sum_{k=N}^{2N-1} \binom{M}{k} s^{2k} c^{2(M-k)} \\ &= \sum_{k=0}^{N-1} \binom{M}{k} s^{2k} c^{2(M-k)} + \sum_{k=0}^{N-1} \binom{M}{M-k} s^{2(M-k)} c^{2k} \\ &= \sum_{k=0}^{N-1} \binom{M}{k} s^{2k} c^{2(M-k)} + \tau_{1/2} \sum_{k=0}^{N-1} \binom{M}{k} s^{2k} c^{2(M-k)} \\ &= P_N + \tau_{1/2} P_N, \end{aligned}$$

where

$$P_N \triangleq c^{2N} \cdot Q_N \quad (5.23)$$

and

$$Q_N \triangleq \sum_{k=0}^{N-1} \binom{2N-1}{k} s^{2k} c^{2(N-1-k)}. \quad (5.24)$$

Clearly, $P_N > 0$ and $Q_N > 0$ almost everywhere since all terms in their respective summations are positive almost everywhere.

In light of the preceding, it is clear that if H were chosen such that

$$|H|^2 = 2 \cdot \left| \frac{1 + e_{-1}}{2} \right|^{2N} Q_N$$

then Equation (5.14) would be satisfied. It remains to find a “square root” of $|H|^2$ that has an associated finite length filter $\{h_n\}$. This is a problem of spectral factorization and may be solved in many ways ([SN96], [Pap77]). A direct method involves finding the roots of $|H|^2$ and choosing appropriately half of those to construct the “square root” H .

⁵Recall that the binomial expansion of a sum raised to the integer power M is $(a+b)^M = \sum_k \binom{M}{k} a^k b^{M-k}$, where $\binom{M}{k} \triangleq M!/(k!(M-k)!)$ is the binomial expansion coefficient.

The following MATLAB code segment computes the Daubechies scaling filter coefficients using a minimum phase cepstral ([OS75, Sections 7.2,10.5]) spectral factorization technique.

```

function hN=hdaub(N)

% initialization

M = 2*N-1;
bM = binomial(M); % the binomial expansion coefficients

npoints=16*round2(N); % # points in FFT
% (factor of 16 gives good accuracy)

gg = 0:npoints-1;
gg = gg/npoints; % frequency sample set for DTFT

c2 = cos(pi*gg).^2; % cos^2
s2 = sin(pi*gg).^2; % sin^2

% compute QN on the frequency discrete sample set gg

QN = zeros(size(gg));
for kk=0:N-1
    QN = QN + bM(kk+1)* c2.^ (N-1-kk) .* s2.^kk;
end
QN=2*QN; % want sum(h)=sqrt(2) (not 1)

% compute spectral factorization of QN via cepstral technique

HlogQN = Hilbert(log(QN));
QNroot = exp(conj(HlogQN));
PNroot = ((1+exp(-j*2*pi*gg))/2).^N .* QNroot;

% IDFT to get the filter coefficients

hN = real(ifft(PNroot));

% coefficients with indices > 2N should all be zero
% so retain only first 2N

hN = hN(1:2*N);

```

5.5.3 Examples

For several values of N the Daubechies *scaling* and *wavelet* functions are displayed in various forms in the figures that follow. Figures 5.4 through 5.6 display in the time and frequency domains the scaling and wavelet functions as well as their corresponding discrete filters. The specific values used in these three figures are $N = 2, 4$ and 7 , respectively. Each of these figures consists of two parts: (a) and (b). In part (a) of each figure a two by two grid of four separate plots is displayed. Starting from the upper left and proceeding clockwise these plots show for each value of N

1. the length $2N$ discrete scaling filter $\{h_n\}$,
2. the length $2N$ discrete wavelet filter $\{g_n\}$,
3. the wavelet function ψ in the time domain, and
4. the scaling function ϕ in the time domain.

In part (b) of each figure the wavelet and scaling functions in the frequency domain are illustrated. They plot on the same axes the power spectra $|\widehat{\phi}|^2$ and $|\widehat{\psi}|^2$ for each value of $N = 2, 4$, and 7 .

From the figures, several effects on the wavelet and scaling functions are evident as N increases:

1. the (finite) length of the support of the discrete time filters h and g increases as $2N$,
2. the length of the support of the continuous time wavelet and scaling functions ϕ and ψ increases,
3. their degree of smoothness or *regularity* (continuity and differentiability) of the scaling and wavelet functions ϕ and ψ increases, and
4. the power spectra $|\widehat{\phi}|^2$ and $|\widehat{\psi}|^2$ become more concentrated on the intervals $(-1/2, 1/2)$ and $(-3/2, -1/2) \cup (1/2, 3/2)$.

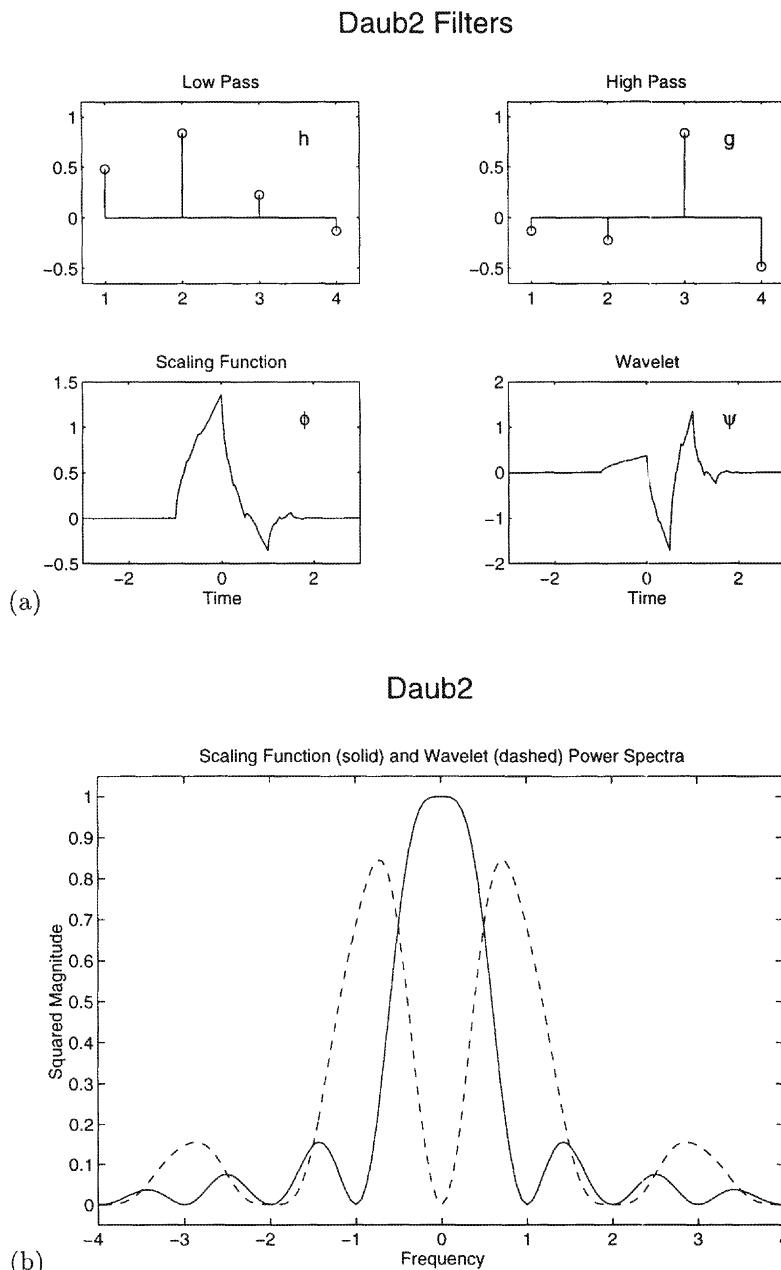
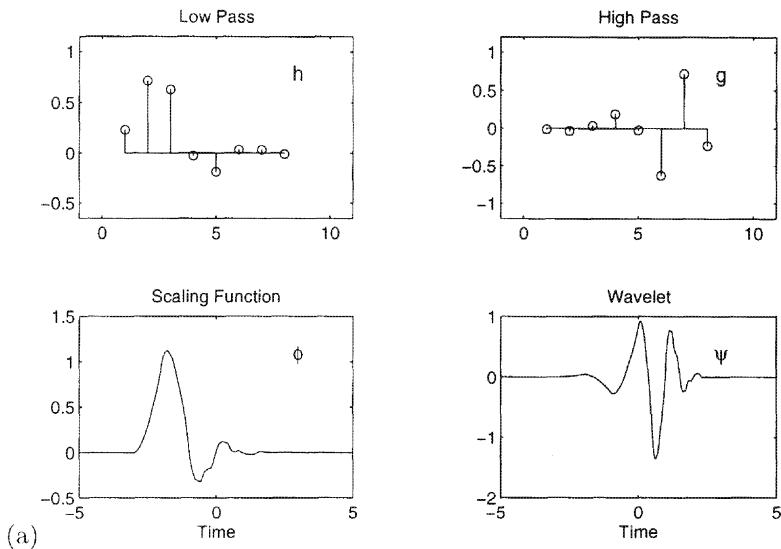


FIGURE 5.4. (a) Daubechies 4 tap filter ($N = 2$) low and high pass filter coefficients and their corresponding scaling and wavelet functions; (b) magnitude squared (power) spectra of Daubechies scaling (solid) and wavelet (dashed) functions for $N = 2$.

Daub4 Filters



Daub4

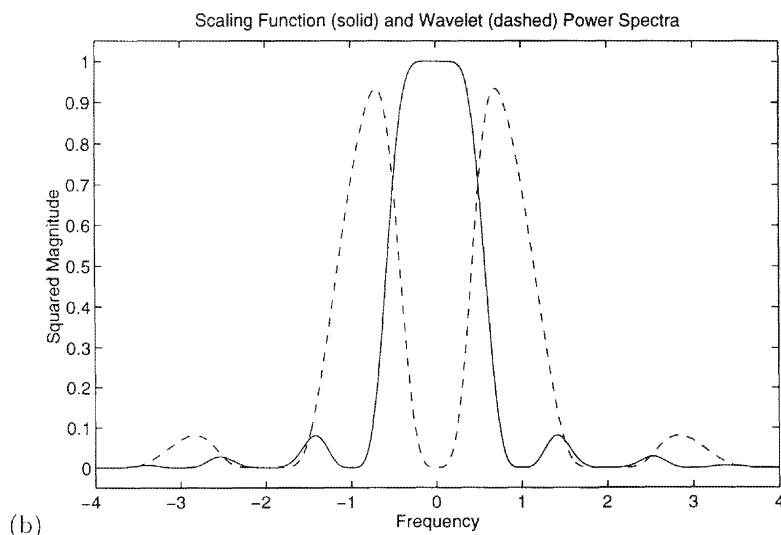
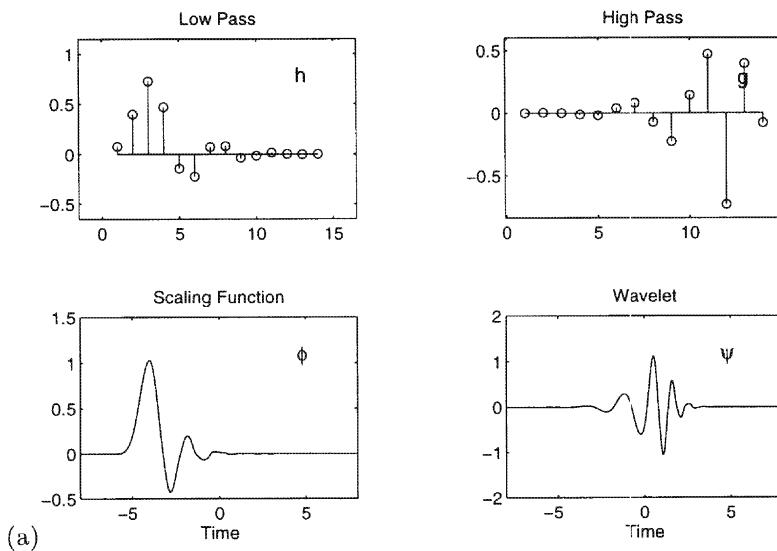


FIGURE 5.5. (a) Daubechies 8 tap filter ($N = 4$) low and high pass filter coefficients and their corresponding scaling and wavelet functions; (b) magnitude squared (power) spectra of Daubechies scaling (solid) and wavelet (dashed) functions for $N = 4$.

Daub7 Filters



Daub7

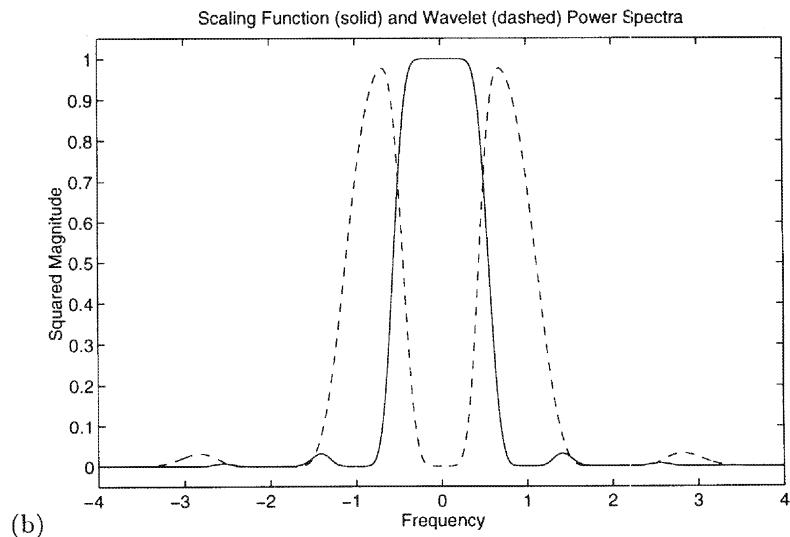


FIGURE 5.6. (a) Daubechies 14 tap filter ($N = 7$) low and high pass filter coefficients and their corresponding scaling and wavelet functions; (b) magnitude squared (power) spectra of Daubechies scaling (solid) and wavelet (dashed) functions for $N = 7$.

5.6 Fast Wavelet Transform Algorithm

Mallat's fast algorithm for the computation of the discrete wavelet transform coefficients associated with compactly supported wavelet ONBs is reviewed here. As mentioned earlier, a driving reason for the great interest in wavelet ONBs with compact support is that they facilitate computation of their associated discrete wavelet transforms using an efficient combination of FIR filters and down-sampling⁶. This algorithm is the fast wavelet transform (FWT) and is described in the following.

The FWT is especially well suited for implementation on a digital platform since its main computational elements are discrete convolutions with FIR filters and down- or up-sampling (by a factor of two) operations. A single level of the FWT decomposition algorithm consists of filtering an input signal f in parallel with H and G followed by down-sampling (removing every other sample). A unique feature of the orthonormal wavelet decompositions is that the synthesis (reconstruction) filters are given by the conjugate of the analysis (decomposition) filters. Similarly, the inverse of a single level consists of up-sampling (zero insertion between every sample) and filtering each channel with a corresponding two-filter synthesis bank of filters. Another key feature of the decomposition is in fact that the (frequency folding) effects of down-sampling are inherently negated in up-sampling due to the MRA structure imposed.

5.6.1 Filter Bank Decomposition of the Identity

Many interesting and useful signal processing techniques may be developed by writing the identity system in a creative way. The FWT algorithm is one example. With respect to a MRA of \mathcal{H} Equation (5.14),

$$|H|^2 + |G|^2 = 2$$

naturally suggests a two-channel filter bank. It may be interpreted directly as a cascade of a two-channel analysis bank and a two-channel synthesis bank of filters as depicted in Figure 5.7. If H acts as a low pass filter then G necessarily acts as a high pass filter and the simultaneous filtering by H and G constitutes a two-band analysis filter bank. Similarly, the synthesis bank adds the outputs obtained by filtering its channel inputs by \overline{H} and \overline{G} . Within the MRA structure, filtering the outputs of the H channel by \overline{H} and the G channel by \overline{G} and then adding the outputs recovers twice the input. This is because the overall system response H_o of this analysis/synthesis cascade is exactly $|H|^2 + |G|^2$. Thus, if x is an arbitrary input and y is the output of the analysis/synthesis cascade as depicted in Figure 5.7, then $y = 2x$.

⁶There are other nonorthonormal families of wavelets that also have fast algorithms associated with them ([AA95]).

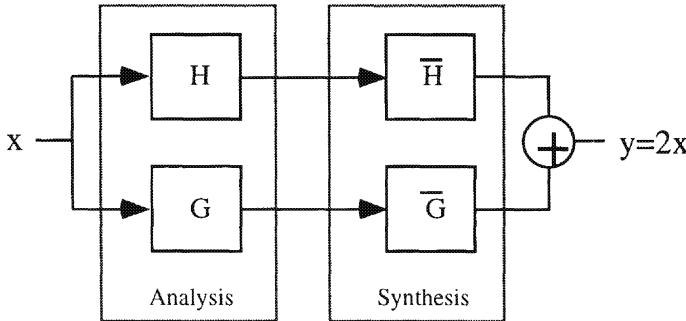


FIGURE 5.7. Two-band decomposition and synthesis bank suggested by Equation (5.17).

5.6.2 Down- and Up-Sampling

It turns out that the FWT may be made even more computationally tractable with the addition of down- and up-sampling components between the analysis and synthesis banks of Figure 5.7. Because of the imposed MRA structure the aliasing introduced by down-sampling is completely negated. This point is elaborated in the following.

Let $c = \{c_n\} \in \ell^2(\mathbb{Z})$. The down-sampling and up-sampling operators, $S_\downarrow : \ell^2(\mathbb{Z}) \mapsto \ell^2(\mathbb{Z})$ and $S_\uparrow : \ell^2(\mathbb{Z}) \mapsto \ell^2(\mathbb{Z})$, respectively, are defined as follows. Operating on a sequence with the down-sampling operator S_\downarrow has the effect of simply discarding all the odd samples; that is,

$$(S_\downarrow c)_n \triangleq \{c_{2n}\}. \quad (5.25)$$

In the Fourier domain, down-sampling has the effect of introducing aliasing by

$$(S_\downarrow c)^\wedge = 2^{1/2} D_{2^{-1}} (\widehat{c} + \tau_{1/2} \widehat{c}).$$

Operating on a sequence with the up-sampling operator S_\uparrow yields a new sequence with the values of the original sequence at the even indices and zero values elsewhere; that is,

$$(S_\uparrow c)_n \triangleq \begin{cases} c_{n/2}, & n \text{ is even,} \\ 0, & n \text{ is odd.} \end{cases} \quad (5.26)$$

In the Fourier domain, up-sampling has the effect of dilating the frequency response by

$$(S_\uparrow c)^\wedge = 2^{-1/2} D_2 \widehat{c}.$$

Although $c = S_\downarrow S_\uparrow c$ for all $c \in \ell^2(\mathbb{Z})$ it is clearly not true that c is equal to $S_\uparrow S_\downarrow c$ since all odd coefficients in c are zeroed by the operation $S_\downarrow S_\uparrow$; that is,

$$(S_\uparrow S_\downarrow c)_n = \frac{1}{2}(1 + (-1)^n)c_n,$$

so that

$$(S_{\uparrow} S_{\downarrow} c)^{\wedge} = \frac{1}{2} (\widehat{c} + \tau_{1/2} \widehat{c}).$$

Note that in terms of the up- and down-sampling operators Equations (5.14) and (5.15) may be written as

$$S_{\uparrow} S_{\downarrow} |H|^2 = S_{\uparrow} S_{\downarrow} |G|^2 = 1$$

and

$$S_{\uparrow} S_{\downarrow} (G \overline{H}) = 0.$$

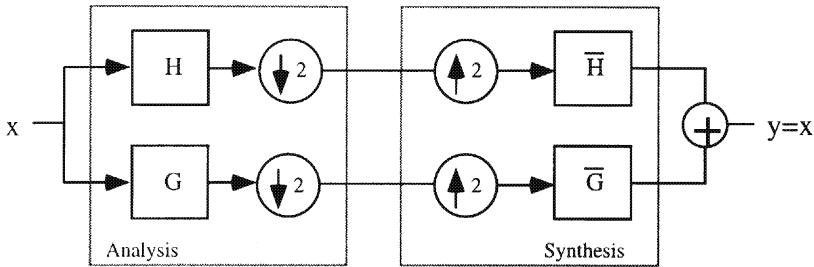


FIGURE 5.8. A single level of the fast wavelet decomposition cascaded with its corresponding fast wavelet reconstruction filters.

Figure 5.8 shows one level of an orthonormal wavelet decomposition immediately followed by its inverse. For one level of the decomposition a length M (assumed a power of two) discrete signal f is transformed into two $M/2$ length signals ($S_{\downarrow} Gf, S_{\downarrow} Hf$). Note that the number of points in the original signal and the combined transform is the same; that is, the number of samples is preserved by the FWT.

To verify that the single-level wavelet decomposition and reconstruction pictured in Figure 5.8 is in fact a factorization of the identity system it must be shown that the overall system response $H_o = 1$, where

$$H_o \triangleq \overline{H} S_{\downarrow} S_{\downarrow} H + \overline{G} S_{\downarrow} S_{\downarrow} G.$$

Expanding H_o gives

$$H_o = \frac{1}{2} \left(|H|^2 + |G|^2 + \underbrace{(\tau_{1/2} H) \overline{H} + (\tau_{1/2} G) \overline{G}}_0 \right) = 1,$$

where the bracketed term is the *aliasing* component. This component is nullified because the choice of G from Equation (5.16) yields

$$\begin{aligned} (\tau_{1/2}H)\bar{H} + (\tau_{1/2}G)\bar{G} &= (\tau_{1/2}H)\bar{H} + \bar{H}(\tau_{1/2}Q)\bar{Q}(\tau_{1/2}H) \\ &= (\tau_{1/2}H)\bar{H} [1 + \bar{Q}\tau_{1/2}Q] \\ &= (\tau_{1/2}H)\bar{H} [1 - |Q|^2] = 0, \end{aligned}$$

where the last equality follows from the constraint (5.18).

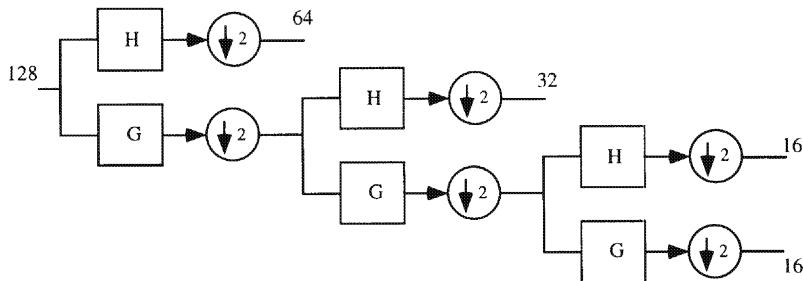


FIGURE 5.9. A three-level wavelet decomposition (analysis) with signal lengths indicated.

Figure 5.9 illustrates the FWT for three consecutive levels. To perform the FWT on an underlying analog signal f using a finite precision digital machine it is of course necessary to provide a discrete and finite representation of that analog signal. In practice, and in accordance with the classical sampling theorem, the most common discrete representation employed is the direct sampling of the underlying signal over some finite interval. Although this may not be the best approach for discretization in a given context ([SN96]), it is nevertheless a straightforward and widely practiced one.

With this said, suppose now that $c_f \in \ell^2(\{0, 1, \dots, N-1\})$; for example, $c_f = \{f(n\Delta)\}_{n=0}^{N-1}$, $\Delta > 0$ is an appropriate discrete representation of the analog signal f . Letting $G_1 \triangleq S_1 G$ and $H_1 \triangleq S_1 H$, note that the number of points in the composite transform

$$(G_1^3 c_f, H_1 G_1^2 c_f, H_1 G_1 c_f, H_1 c_f)$$

is again the same as the discrete signal c_f . In other words, under the FWT the number of points in the signal and transform domains is preserved. This fact allows the FWT to be implemented in an “in-place” algorithm in which the signal and transform occupy the same location in digital memory during the transformation process.

5.6.3 Examples

In the figures that follow, discrete wavelet transforms of two different synthetic signals are illustrated. Each DWT corresponds to one of the three Daubechies wavelets ($N = 2, 4$, and 7) displayed in Figures 5.4 through 5.6. The two synthetic signals used in the examples are called “packet” and “FMramp” and are both analytic (in $H_+^2(\mathbb{R})$).

The signal “packet” consists of a nonoverlapping (in time) concatenation of three fixed frequency (6, 11, and 20Hz) sinusoidal bursts all having the same bell-shaped envelope. Figures 5.12 through 5.14 show the magnitude of the DWT of this signal for the Daubechies wavelets $N = 2, 4$, and 7 , respectively. Each of these figures depicts the dyadic sampling in the time scale plane for seven values of scale. Each line in the figure corresponds to an output of a decomposition stage of the FWT decomposition depicted in Figure 5.9.

The signal “FMramp” is a unit magnitude pulse with a frequency modulation (FM) that ramps from about 5Hz to about 30Hz over 25 seconds and then ramps back down in about 3 seconds. This modulation and the real part of its corresponding time domain signal are shown in Figure 5.10. Its Fourier transform is shown in Figure 5.15. As in the case of the signal “packet,” Figures 5.16 through 5.18 show the magnitude DWT of “FMramp” for the Daubechies wavelets $N = 2, 4$, and 7 respectively.

The DWT plots shown in Figures 5.12 through 5.18 depict the transform in the time-scale plane with the scale axis doubling as the magnitude axis. This is possible since the scale sampling is so sparse. For each scale, the DWT is shown as a sampled output function, sampled at the scale-dependent rate dictated by the DWT. Sample points in the plot are marked explicitly by the symbol \circ .

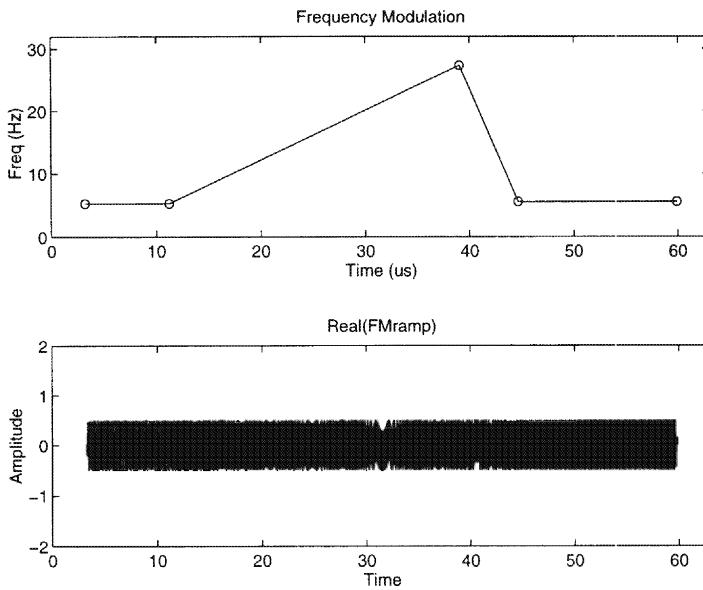


FIGURE 5.10. Frequency modulation and corresponding time domain signal for the signal “FMramp.”

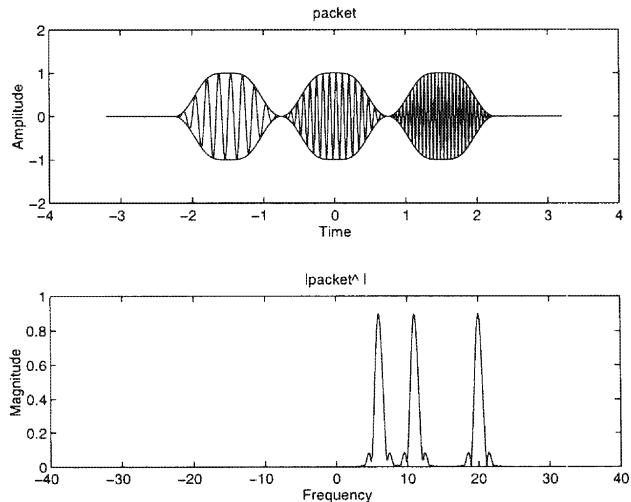


FIGURE 5.11. Synthetic signal “packet” in the time and frequency domains. The top plot shows the real part and magnitude of the time signal and the bottom plot shows the Fourier transform magnitude.

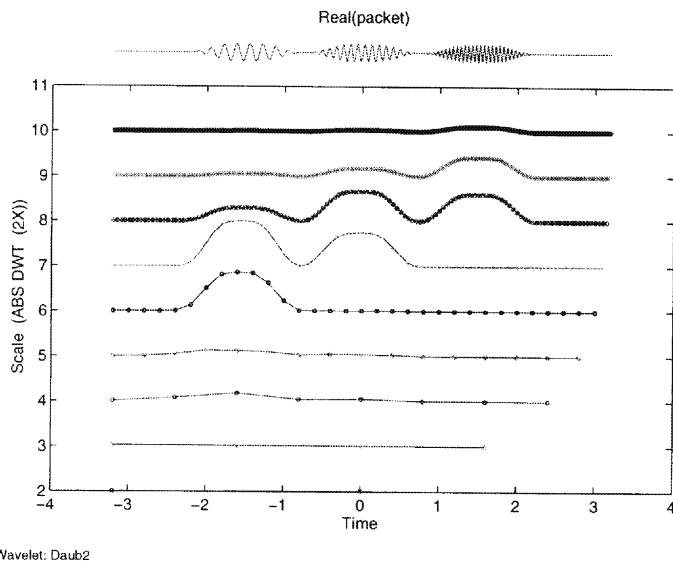


FIGURE 5.12. Daubechies 4 tap ($N = 2$) discrete wavelet transform magnitude of the signal “packet.”

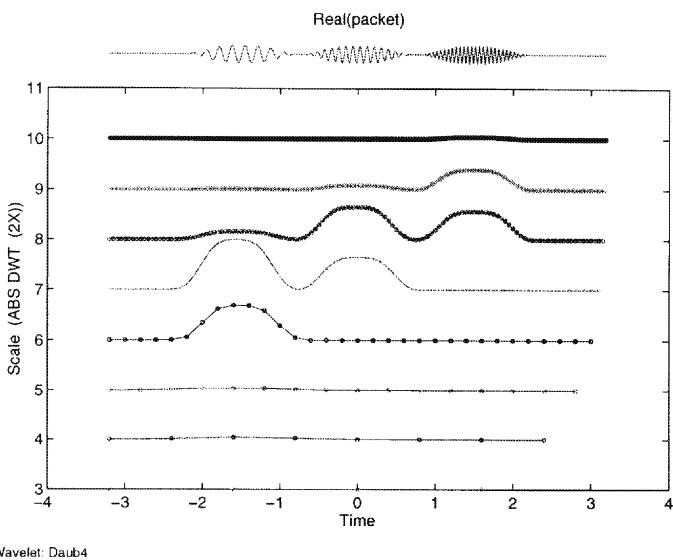


FIGURE 5.13. Daubechies 8 tap ($N = 4$) discrete wavelet transform magnitude of the signal “packet.”

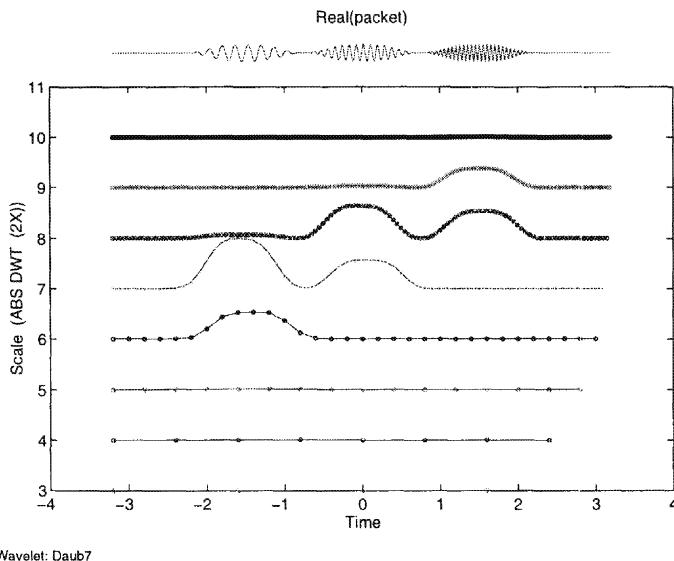


FIGURE 5.14. Daubechies 14 tap ($N = 7$) discrete wavelet transform magnitude of the signal “packet.”

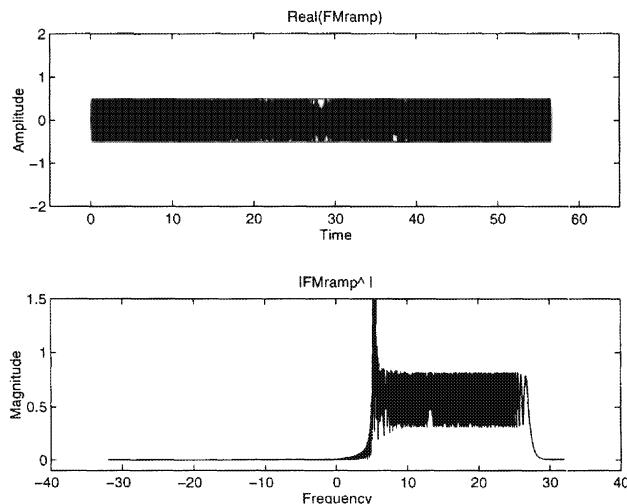


FIGURE 5.15. Synthetic signal “FMramp” in the time and frequency domains. The top plot shows the real part and magnitude of the time signal and the bottom plot shows the Fourier transform magnitude.

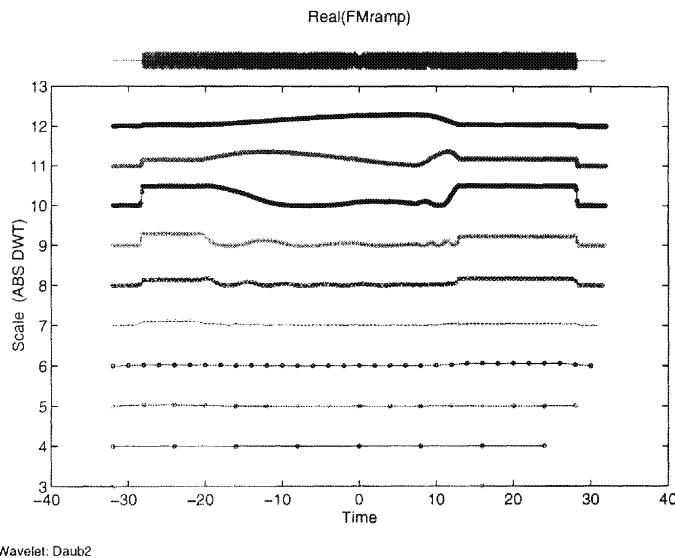


FIGURE 5.16. Daubechies 4 tap ($N = 2$) discrete wavelet transform magnitude of the signal “FMramp.”

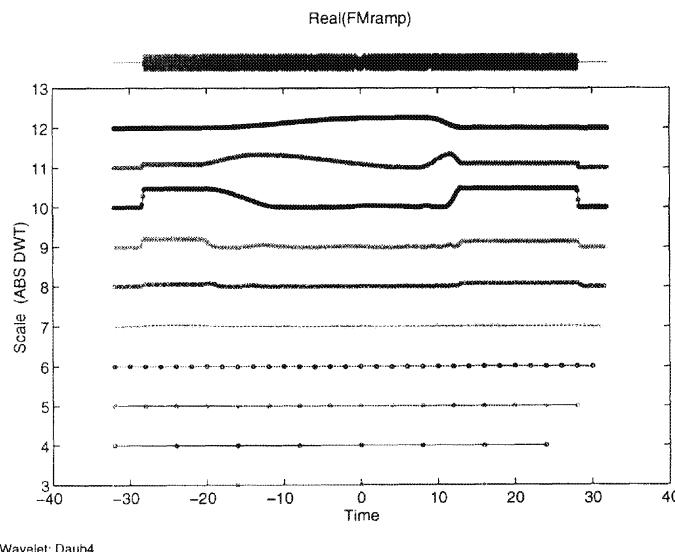


FIGURE 5.17. Daubechies 8 tap ($N = 4$) discrete wavelet transform magnitude of the signal “FMramp.”

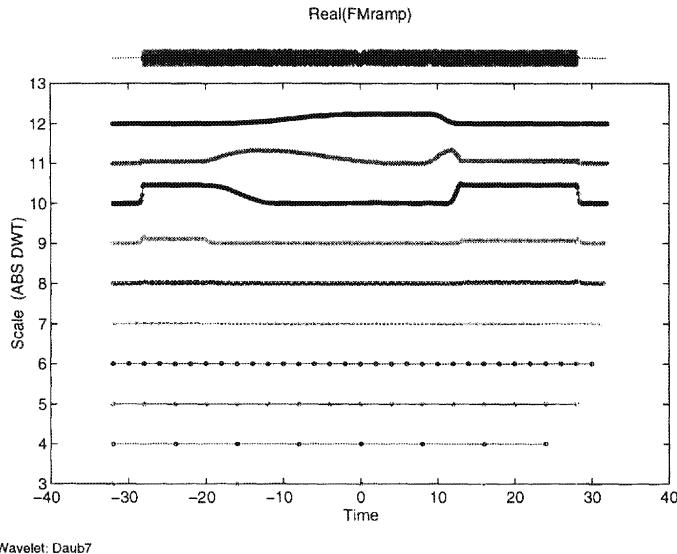


FIGURE 5.18. Daubechies 14 tap ($N = 7$) discrete wavelet transform magnitude of the signal “FMramp.”

Problems

5.1 (Haar MRA) Let ϕ be the Haar scaling function $\phi_{\text{Haar}} \stackrel{\Delta}{=} 1_{(0,1]}$. Verify the following.

- (a) The integer translations of ϕ form an orthonormal sequence and, furthermore, it is an ONB for $V_0 = \overline{\text{span}} \{ \tau_n \phi \}$.
- (b) All the properties listed in Table 5.1 (page 108) hold for this scaling function.

5.2 Show that the DTFT of $L_{\phi,k} f$ is

$$(L_{\phi,k} f)^{\wedge} = \sum_n \tau_n \left[\left(D_{2^k} \widehat{f} \right) \cdot \overline{\widehat{\phi}} \right].$$

5.3 Let ϕ be given by

$$\phi(t) = \int_{-1}^1 \cos(\pi \gamma / 2) \cdot e^{j2\pi \gamma t} d\gamma.$$

Show that

- (a) $\{ \tau_n \phi \}$ is an orthonormal sequence in $L^2(\mathbb{R})$.

(b) $\{\tau_n \phi\}$ is not an orthonormal basis for $L^2(\mathbb{R})$.

5.4 Suppose that the sequence $\{h_n\}$ has finite length. Show that if

$$\sum_k h_k h_{k-2n} = \delta_{n,0}$$

then $\{h_n\}$ must be of even length. The length is defined as $n_{\max} - n_{\min} + 1$, where n_{\max} is the largest index of all the nonzero coefficients of $\{h_n\}$ and n_{\min} is smallest.

5.5 Show that if ϕ and ψ are the scaling and wavelet functions associated with a valid MRA then

$$\sum \tau_{2n} (|\widehat{\phi}|^2 + |\widehat{\psi}|^2) = 1.$$

5.6 Show that if ϕ is compactly supported and satisfies the dilation equation

$$\phi = \sum h_n D_2 \tau_n \phi$$

for some $\ell^2(\mathbb{Z})$ sequence $h \stackrel{\Delta}{=} \{h_n\}$ then h is of finite length.

6

Overcomplete Wavelet Transform

This chapter deals with discrete wavelet transforms that are formed from the general samples of a continuous wavelet transform. Conceptually, there are few constraints on the spacing between sample points throughout the time-scale plane; however, computational consideration is restricted here to an interesting subclass of sampling sets that allow for the fast computation of the forward and inverse transforms.¹ In this case, the freedom of choice of analyzing wavelets remains nearly unrestricted by the implementation. In general, the resulting transform is one that has underlying atoms that are *nonorthogonal*, and even more important, may be *overcomplete*. Consequently, such atomic functions have associated *redundant* (inner product) representations. For these reasons, the term *overcomplete wavelet transform* (OCWT) is used to describe the transform.

The chapter starts with a formal mathematical definition of the OCWT, then presents a discussion of the merits and drawbacks of overcompleteness. This is followed by a description of the constrained implementations of the forward and inverse OCWT. After this, the topic of analyzing wavelet design is addressed and a new simply characterized wavelet is introduced. Computational examples showing the OCWT of several signals is presented next. The chapter concludes with a detailed mathematical accounting of the general relationship between irregular sampling and frames. Several of the key issues dealt with throughout the chapter are first briefly introduced in the following paragraphs.

Why Overcomplete over Orthonormal?

Mathematically speaking, orthonormal representations are appealing because of their mathematical tractability in both the analytical and computational senses. In many cases, as is the case with a compactly supported ONB, there are low complexity algorithms for the computation of the asso-

¹In fact, the spirit in which the OCWT is used computationally throughout this book is more in line with the fine “regular” sampling of the continuous wavelet transform than the arbitrary irregular sampling supported in the theoretical discussion.

ciated orthonormal representation. Practically speaking, though, the use of orthonormal representations is not necessarily best with respect to a given signal processing problem. In fact, a motivating concept behind this chapter is the idea that overcomplete wavelet representations offer definite practical advantages over conventional orthonormal representations and have application to a variety of signal processing tasks. Potential advantages and disadvantages are discussed in Section 6.1.4 and supported with concrete applications in Chapter 7. Roughly speaking, the two main tradeoffs involved are potential increases in performance for a given application versus computational complexity and storage requirements. In other words, there may be potential performance benefits from using an overcomplete wavelet transform that come at the cost of increased computation.

Implementation and Speed: Constrained Sampling

Almost all of what is developed computationally in this chapter is guided by implementation issues including speed and accuracy. This leads to the consideration of sampling sets that are constrained to have members which come from a *semilog-regular* (see Definition 6.2) time-scale sampling set. Although there are certainly valid and interesting applications in which the discrete time-scale sampling set is fixed or unalterable by the user and, in particular, not regular, these cases are not addressed in the implementations discussed here. Instead, focus is placed on cases in which the discrete sampling set may be considered as a design parameter. Several specific applications of this type including identification and compression are discussed in Chapter 7.

Wavelet Frames and the General Case

Theoretically, the treatment of the material in this chapter is concerned with much more general sampling sets. General irregular sampling in the wavelet domain is strongly connected to frames of wavelets. As such, the theory of frames and irregular sampling (in reproducing kernel Hilbert spaces) serves as the mathematical foundation for the OCWT. This chapter is concluded with a detailed mathematical analysis of reconstruction from overcomplete wavelet representations.

Analyzing Wavelet Design

Also in this chapter, the freedom gained from the more general overcomplete approach allows for the use of near arbitrary analyzing wavelets. This makes the design of OCWTs especially easy. As an example, a set of parametrically described bandlimited (PBL) wavelets is developed in Section 6.3.2. These wavelets enjoy many desirable properties including an infinite number of zero-valued moments, symmetry, a controllable tradeoff between time and frequency localization, and bandlimitedness.

Perspective: The Phi-Transform

Although motivated primarily as a general device for the discrete representation of continuous-time signals, the OCWT is closely related to the *phi-transform* originally introduced by Frazier and Jawerth ([FJ85]) to characterize function and distribution spaces ([Tri95]). Later, the phi-transform was seen as a useful tool for time–frequency analysis ([KFFJ92]). The OCWT and phi-transform are similar in that they both analyze signals using banks of localized bandpass filters with the property that the ratio of their center frequency to bandwidth remains constant, that is, a constant Q filter bank. Much of the motivation and utility of the OCWT is shared with the phi-transform as well. At the very least, the OCWT offers an alternative perspective on many of the same ideas underlying the phi-transform.

6.1 Discretization of the CWT Revisited

As discussed in detail in Chapter 5, the *discrete wavelet transform* (DWT) is a very special type of discretization of the CWT that leads to families of wavelet orthonormal bases. These orthonormal families are comprised of the power of two dilations and corresponding uniform translations of certain analyzing wavelets. It has been seen that such wavelet families and their associated representations may be related to a sampling of the CWT on a “regular” sampling set, that is, the dyadic discrete sampling set. For generally overcomplete wavelet transforms the constraints of regular sampling of the CWT and orthonormality of the underlying wavelet atoms are removed. This freedom may then be exploited to design wavelet families with many desirable properties (viz. Sections 4.2 and 6.3.2).

6.1.1 *Definition*

An overcomplete wavelet transform (OCWT) is defined here as a discrete transform given by the sample values of a continuous wavelet transform over a discrete sampling set in the time–scale plane. Specifically, an OCWT is defined with respect to the main items:

1. an analyzing wavelet g (viz. Section 4.3.1),
2. a continuous wavelet transform W_g , and
3. a discrete time–scale sampling set Γ .

Note that the definition does not constrain the time–scale sampling set to be “regular” in any sense; and, in particular, it is not required to be

the dyadic discrete sampling set associated with the DWT. Let Γ denote such a sampling set. This Γ is a countable subset or *discrete sampling set* in the time-scale plane (affine group \mathcal{G}_A) that may be written as $\Gamma \stackrel{\Delta}{=} \{(t_{m,n}, s_{m,n})\} \subset \mathcal{G}_A$ where each pair $(t_{m,n}, s_{m,n})$ is a point in the time-scale plane. A formal definition of the OCWT follows.

Definition 6.1 (OCWT) *Let g be an analyzing wavelet and a discrete time-scale sampling set*

$$\Gamma = \{(t_{m,n}, s_{m,n})\}.$$

An overcomplete wavelet transform (OCWT) of the function $f \in L^2(\mathbb{R})$ is the continuous wavelet transform $W_g|_\Gamma$ evaluated on the time-scale sampling set Γ :

$$\begin{aligned} W_g|_\Gamma : L^2(\mathbb{R}) &\rightarrow \ell^2(\mathbb{Z}^2) \\ f &\mapsto \{(W_g f)(t_{m,n}, s_{m,n})\}_{(m,n) \in \mathbb{Z}^2}. \end{aligned}$$

An OCWT $W_g|_\Gamma$ is equivalently written in the following ways.

$$\begin{aligned} W_g|_\Gamma f &\stackrel{\Delta}{=} \{W_g f(t_{m,n}, s_{m,n})\} \\ &= \{W_g f(t, s)|_{(t,s) \in \Gamma}\} \\ &= \left\{ \langle f, \tau_{t_{m,n}} D_{s_{m,n}} g \rangle_{L^2(\mathbb{R})} \right\}. \end{aligned}$$

Thus, each (fine enough) choice of discrete sampling set $\Gamma = \{(t_{m,n}, s_{m,n})\}$ and wavelet g specifies a countable set of wavelet atoms $\{\tau_{t_{m,n}} D_{s_{m,n}} g\}$ and the inner product representation associated with this set of wavelet atoms is an overcomplete wavelet transform over the span of the atoms.

Without loss of generality sampling sets Γ may be described as

$$\Gamma = \{(t_{m,n}, s_m)\}; \quad (6.1)$$

that is, the dependence of scale sample values on the index n may be removed. In practice it is helpful if there are many time sample values $t_{m,n}$ associated with the same s_m since for a fixed m

$$W_g f(t_{m,n}, s_m) = (f * D_{s_m} \tilde{g})(t_{m,n}) \quad (6.2)$$

may be computed as the output of a single fixed filter (viz. Section 6.2). With this in mind sampling sets Γ associated with the OCWT are described with single index scale values as in (6.1).

6.1.2 Semilog Regular Time-Scale Sampling

With respect to the implementation concerns of speed and accuracy, sampling sets with a “regular” structure are of special interest. One regular

time-scale discrete sample set that has already been introduced is the dyadic sampling set Γ_D associated with the discrete wavelet transform (viz. Equation (5.1)). Discussed in Section 5.6, the speed of the fast wavelet transform is due (partly) to the special “regular” structure associated with the dyadic sampling grid. The dyadic sampling set is regular in a sense that is dependent on the affine-group structure associated with the wavelet transform. Very briefly, a time-scale sampling set $\{(t_n, s_n)\}$ is regular if each point in the set has a region V_n associated with it such that the following conditions are met.

1. The entire collection of regions $\{V_n\}$ partitions the time-scale plane G_A ; that is,

$$\begin{aligned} \text{(a)} \quad & G_A = \bigcup_n V_n, \\ \text{(b)} \quad & \forall n \neq m, \quad V_n \cap V_m = \emptyset; \end{aligned}$$

and

2. each region V_n under the group action $\tau_{t_n} D_{s_n}$ maps to a fixed region V (independent of n).

This notion of regularity with respect to a general group may be made precise (viz. [OS92]).

Here, a different type of regularity is presented that leads to overcomplete wavelet transforms with fast implementations. Simply stated, this type of regular sampling is generated by sampling the time and scale axes independently: the scale samples are exponentially spaced whereas the time samples are uniformly spaced. The resulting overall time-scale sampling set is generated by the cross-product of the two one-dimensional sampling sets. A formal definition of a semilog regular sampling set is given in Definition 6.2 and further elaborated in Section 6.2 where a filter bank implementation of the OCWT is discussed.

Definition 6.2 (Semilog Regular Sampling) *A semilog regular set $\Gamma(\Delta, a_0)$ is described by two scalars $a_0 > 1$ and $\Delta > 0$ as*

$$\Gamma(\Delta, a_0) \triangleq \{n\Delta\} \times \{a_0^m\}.$$

6.1.3 Invertibility

Just as the CWT may not have an inverse for an arbitrary choice of analyzing function g , not every pairwise choice of discrete sampling set Γ and analyzing wavelet g will lead to an OCWT that is invertible. If such an OCWT is to be invertible on some space \mathcal{H} the choices of Γ and g must give rise to a wavelet frame for \mathcal{H} . More precisely, a necessary and sufficient condition for the invertibility of the OCWT on a Hilbert space \mathcal{H} is

that the underlying atoms $\{\tau_{t_{m,n}} D_{s_m} g\}$ form a frame for \mathcal{H} . Such a set of functions is said to be a wavelet frame for \mathcal{H} .

It should be noted that this formulation requires only that the underlying wavelet atoms form a frame for the space of interest (one containing the signal or signals of interest) and that this is a *much* weaker requirement than orthonormality. In particular, the atoms may be severely redundant or *overcomplete*. Within this framework, an OCWT inverse may be formulated using the general frame theory of Chapter 3 as applied to wavelet frames. Wavelet frames are explicitly addressed in Section 6.5.2 and a filter bank version of the inverse OCWT is given in Section 6.2.2. Further discussion of the inverse is deferred until then.

6.1.4 Overcompleteness and Redundancy

In this section potential advantages and disadvantages associated with the OCWT are discussed. Applications illustrating these benefits are presented in Chapter 7.

An overcomplete representation is precisely one that has underlying atoms which form a nonexact (non-Riesz) frame (viz. Section 3.3). Overcomplete representations are redundant in the sense that the underlying atoms are not independent; that is, pairs of distinct atoms do not necessarily have a zero-valued inner product (cf. Sections 3.3 and 3.4). Although the formulation of the OCWT includes orthonormal and Riesz wavelet bases as special cases, its greatest generality lies within the realm of truly overcomplete wavelet frame representations.

A natural question that arises out of this approach is what advantages and disadvantages are there to overcomplete (or redundant) wavelet representations. Overcomplete wavelet frame representations have some distinct benefits to offer in practical signal processing applications. For one thing, orthonormal bases preclude the inherent noise robustness benefits associated with redundant systems as discussed in Section 7.1.7. For another thing, in many applications the condition that the wavelet system must be orthonormal is undesirable. For instance in pattern classification a desirable property of a representation of a signal is that it be translation-invariant. In other words, translated versions of signals should yield translated versions of their discrete representations. This is clearly not satisfied for fixed sampling geometries such as are associated with orthonormal wavelet systems. Moreover, there are still reasonably fast algorithms for the computation of the wavelet representation for some overcomplete systems. Specific benefits (and disadvantages) are discussed in this section.

Generally, the advantages of overcomplete wavelet representations over conventional wavelet basis representations include robustness to imprecision in representation coefficients, for example, quantization effects, and freedom gained in the selection of the analyzing wavelet. Removing the requirement of nonredundancy allows the wavelet generating function, that

is, the analyzing wavelet, to be chosen almost arbitrarily. This freedom may be exploited to better achieve certain processing goals.

Specifically, concise and effective signal representations intended for use in detection, identification, or compression applications are facilitated by representations with underlying atoms that have properties similar to the signals of interest. For signals occurring in common natural or manmade systems (e.g., speech or radar signals), conventional Fourier, polynomial, or even compactly supported orthonormal wavelet bases do not² consist of elements that are similar to signals of interest.

Moreover, overcompleteness allows for an *irregular* sampling component in the design of signal representations. There are some unique advantages to adopting an irregular sampling strategy in a representation. Many of these advantages are inherited from the ability of an irregular sampling to be sensitive to a signal's time–frequency behavior. Supported by the theory of mathematical frames ([Teo93]) irregular sampling (in an appropriate transform domain) offers a powerful tool for signal representation. In the following some further justification for the incorporation of an irregular sampling component and overcompleteness, in general, in the design of signal representations is provided. Although the discussion is limited to the direct sampling of one-dimensional bandlimited signals, these same ideas may be extended to sampling of multidimensional representations.

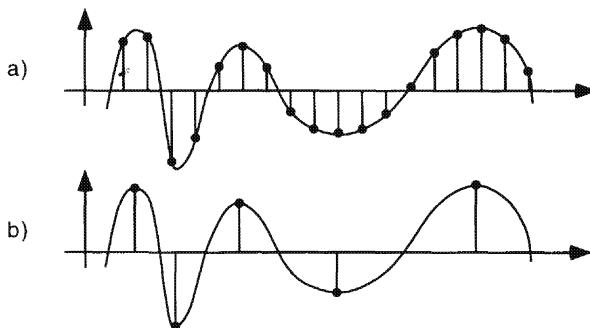


FIGURE 6.1. A signal with varying time–frequency content and two sampling strategies: (a) uniform sampling in accordance with the Nyquist criterion (allows reconstruction), and (b) irregular sampling with local extrema (may not allow reconstruction).

In some practical signal processing applications the disadvantages associated with the overcomplete wavelet representation are outweighed by

²Semiorthogonal wavelet basis elements do exhibit greater flexibility in matching desired signals of interest (viz. [AU93], and [AA95]).

the potential benefits (c.f. Chapter 7). Some of the major advantages and disadvantages of OCWTs are listed in the following.

Disadvantages of Overcomplete Representation

Overcomplete representations have the following main drawbacks.

1. OCWTs may require increased computational complexity and storage requirements as compared to the compactly supported orthonormal wavelet representations. An OCWT, however, may be implemented using fast Fourier transforms to perform the filter bank filtering. As such, the computational complexity of the OCWT is on the same order as that of an FFT, that is, $O(N \log N)$ where N is the length of the sampled signal (number of samples). Although this is a significant increase over the $O(N)$ complexity of the compactly supported orthonormal wavelet transform it is not too computationally expensive for many signal processing applications.
2. Reconstruction from overcomplete wavelet representations involves iterative and/or nonperfect reconstruction algorithms. Depending on the frame properties of the underlying wavelet atomic set and the specific implementation, a computationally tractable OCWT inverse may yield a reproduction that agrees with the original to only three or four significant digits. In many practical situations (especially where noise is inherently present) this level of reconstruction fidelity is more than sufficient to achieve many processing goals.

Advantages of Overcomplete Representation

Some advantages that overcomplete representations provide are listed in the following.

1. There is an inherent degree of noise robustness in an overcomplete representation domain that is directly proportional to the degree of redundancy in the representation; that is, the larger the degree of redundancy (or overcompleteness) the less sensitive the representation is to perturbation. Robustness, in this sense, means that reconstructions may tolerate more perturbations in the representation for the same reconstruction quality.
2. Simple representation thresholding provides inherent signal domain noise robustness. Because the underlying wavelet atomic functions are *locally* well correlated to the time–frequency structure of complex signals and not well correlated to noise, the wavelet representation of such signals necessarily spreads the noise energy over the entire time–scale plane whereas the signal energy is necessarily localized along its characteristic time–scale trajectory. Thus, any processing done in the representation domain that emphasizes strong coherent components

and suppresses weak noncoherent components will effectively perform noise suppression. Section 7.1.4 discusses these issues in detail.

3. There is great freedom in the choice³ of analyzing wavelet. In particular, this allows the specification of analyzing wavelets with a variety of desirable properties including
 - (a) arbitrary joint time–frequency localization consistent with the uncertainty principle (viz. Sections 4.2 and 6.3.1),
 - (b) linear phase (symmetric in time),
 - (c) real (or analytic), and
 - (d) arbitrary number of vanishing moments.
4. There is great freedom in the choice of discrete sampling set at which to sample the underlying continuous wavelet transform.
 - (a) Classical uniform sampling theory (viz. Theorem 2.6), dictates that for the full recovery of an Ω bandlimited signal from its uniform samples, a (uniform) bound $\Delta \leq 1/(2\Omega)$ must be placed on the sampling period Δ . Thus, for a signal that has high frequency oscillations which are well localized in time and has low frequency oscillations at most other times the classical sampling theory requires a sampling period that seems excessive for most of the signal duration. A more intuitively appealing sampling strategy is sampling at time-varying rates that are sensitive to the time–frequency content of the underlying signal. One such sampling strategy is to record samples at only the extreme points of a function (local extrema). This situation is illustrated in Figure 6.1.
 - (b) Sampling a signal with respect to its time–frequency content is an efficient method of representation. Redundancy (oversampling) is more efficiently achieved via irregular representations since redundant samples can be placed in periods of high information content. This is clearly illustrated in Figure 6.1.
 - (c) Via the theory of Hilbert space frames precise mathematical conditions in the spirit of the classical sampling theory for required sampling densities allowing full recovery of a class of signals are readily derived (viz. Theorem 6.11).
 - (d) There is biological evidence ([YWS92], [BT93]), that processing in biological systems such as the mammalian auditory and vision systems incorporates irregular sampling processes in their respective representations of sensory data.

³This is also achievable for biorthogonal representations.

5. In an orthonormal representation distances between elements in the signal domain and corresponding transformed elements in the representation domain are necessarily the same. Overcomplete representations offer some flexibility here in that distances in the signal and representation domains need not be preserved. This ability may be exploited to move closer in the representation domain elements in the signal space that are known to belong together yet are far from each other.

6.2 Filter Bank Implementation

This section fully describes a linear filter bank implementation of the OCWT and its inverse. An OCWT and its inverse may be implemented using banks of linear filters as a basic component. In terms of the arbitrary sampling in the time-scale plane of an underlying continuous wavelet transform, such an implementation places a weak constraint on the types of sampling sets that may be specified. Practically, a filter bank implementation is most useful when the time-scale sampling set is specified such that the time samples are constrained to occur along horizontal lines in the time-scale plane. In other words, the type of sampling set Γ best facilitated by a filter bank implementation has the form

$$\Gamma = \{s_m, t_{m,n}\}$$

where first the discrete scales $\{s_m\}$ are specified and then for each s_m a sequence of time samples $\{t_{m,n}\}$ is subsequently specified. The sample values of a CWT, W_g , evaluated on this discrete sampling set may be computed by sampling the output of a filter bank described by the impulse responses $\{D_{s_m} \tilde{g}\}$.

The forward and inverse OCWTs may be respectively implemented in terms of an *analysis* filter bank and a *synthesis* filter bank. It is shown that the distribution of the time-scale samples in the sampling set Γ has a direct effect on the numerical complexity of the forward and, to a much greater extent, the inverse transforms. In general, the more “uniformly” distributed the discrete sampling set, the less numerically intensive are the transforms. A type of “uniform” or “regular” sampling set of special interest is one that has a fixed spacing in the time dimension and an exponential spacing in the scale dimension (viz. Equation (6.5)). Such a sample set is labeled here as a *semilog-regular* sampling set.

For near arbitrary irregular discrete sampling sets, there are several computational methods for computing the OCWT on them. These methods are all fundamentally linked to the computation of either the inverse frame operator S^{-1} or the frame correlation pseudo-inverse R^\dagger associated with the wavelet family $\{\tau_{t_{m,n}} D_{s_m} g\}$. One tactic for efficiently implementing

nonuniform or *irregular* sampling is to first compute the transform on a fine uniform discrete sampling set and subsequently interpolate onto the desired irregular discrete sampling set. This approach, however, will introduce some error in the OCWT coefficients. In applications, such error may be tolerable when using redundant wavelet families for analysis since overcompleteness of the representation provides an inherent robustness to coefficient perturbations (viz. Section 7.1.7).

The forward and inverse implementations of the OCWT are discussed in the following subsections. Different types of wavelet implementations are discussed in [She92], [RD92], [XKZ94], and [AA95].

6.2.1 Analysis

Computation of the forward OCWT is facilitated through the specification of an associated *analysis* filter bank. In the OCWT analysis bank the filter impulse responses of the filter bank are determined by the analyzing wavelet g and the scale samples $\{s_m\}$. Such a filter bank provides a scale discretized wavelet transform. To get the fully discretized OCWT the “continuous” outputs of the filters in the bank must each be subsequently sampled at the discrete times $\{t_{m,n}\}$.

Equation (6.2),

$$W_g f(t_{m,n}, s_m) = (f * D_{s_m} \tilde{g})(t_{m,n}),$$

shows directly that the OCWT may be thought of as the linear filter bank described by the impulse responses $\{D_{s_m} \tilde{g}\}$. Figure 6.2 displays an analysis filter bank for performing an overcomplete wavelet transform, that is, a CWT evaluated on the discrete sampling set Γ . For demonstration purposes each filter in the figure contains a plot of a frequency (magnitude) response, where $|\tilde{g}|$ has a trapezoidal form supported on an interval not containing zero. As the filter index m increases, certain primary effects on the transfer functions are evident:

1. the filter bandwidth increases, and
2. the entire interval of frequency support is translated towards higher frequencies.

Consequently, lower filters (with small index m) respond to a lower and smaller range of frequencies than higher filters (with large index m) which respond to a higher and broader range of frequencies. Thus, a signal containing a broad range of frequencies will have output across many filters in the filter bank occurring at various times. In this way, the time–frequency content of the analyzed signal is exposed. Section 6.4 gives examples of the filter bank OCWTs for various single and multicomponent frequency modulated signals.

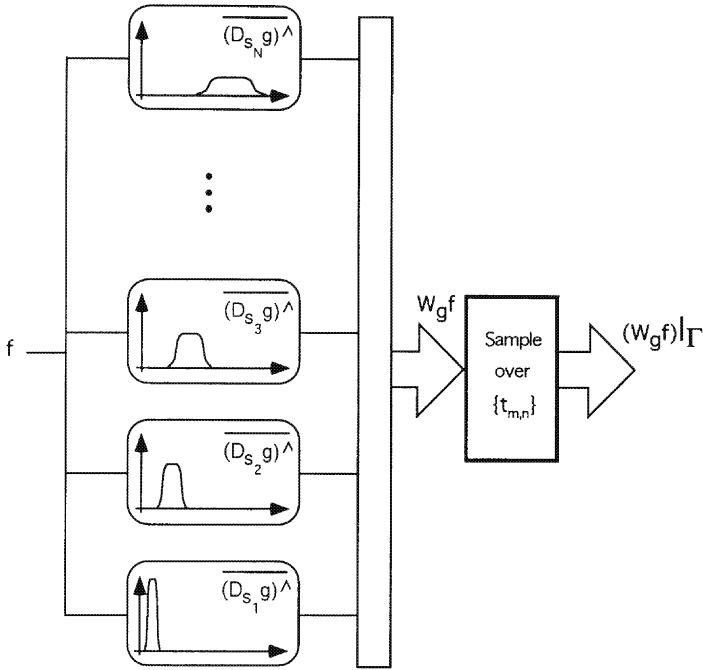


FIGURE 6.2. An overcomplete wavelet transform as an analysis filter bank.

Together, the wavelet g , the number of filters M , and the scales $\{s_m\}$ determine the frequency ranges over which the analysis bank operates. In this regard, the function

$$G(\gamma) \triangleq \sum_{m=1}^M |\widehat{g}(s_m^{-1}\gamma)|^2 \quad (6.3)$$

provides a useful measure of the analysis filter bank frequency coverage. Clearly, any information in a signal that is outside the support of G is lost when filtered by the analysis bank. Not surprisingly, G plays a key role in the OCWT inverse and, consequently, the synthesis bank. This is the topic of the next section.

6.2.2 Synthesis

An OCWT synthesis filter bank is depicted in Figure 6.3. Passing a discrete two-dimensional function of time and frequency (e.g., an overcomplete wavelet transform), through the synthesis bank yields a one-dimensional function of time. For a properly configured synthesis filter bank, the response to the OCWT of a function f is f itself. That is, the synthesis bank

as shown in Figure 6.3 performs the inverse OCWT.

For the remainder of this section, let L denote the OCWT representation, that is, $L = W_g|_{\Gamma}$, with the adjoint L^* . The OCWT representation of f is thus given by

$$(Lf)_{m,n} = W_g f(t_{m,n}, s_m).$$

To establish the adjoint L^* , let $c \in \ell^2(\mathbb{Z}^2)$ be a two-dimensional coefficient sequence $c = \{c_{m,n}\}$. From Section 3.4.2 the adjoint is

$$L^* c = \sum_m \sum_n c_{m,n} \cdot \tau_{t_{m,n}} D_{s_m} g.$$

Provided that the set $\{\tau_{t_{m,n}} D_{s_m} g\}$ forms a frame for a Hilbert space \mathcal{H} then any $f \in \mathcal{H}$ may be recovered as

$$f = L^* R^{-1} L f,$$

where $R \triangleq LL^*$ is the frame correlation (viz. Section 3.4.7). In these terms, the analysis bank performs the operation L and the synthesis bank performs an operation akin to $L^* R^\dagger$, where R^\dagger is the pseudo-inverse of the frame correlation R . Exactly how this is accomplished is now discussed.

With respect to the sampling set Γ there are certain main cases of interest: (i) the general case in which the sample points are irregularly spaced in the time-scale plane; and (ii) the uniform case in which the sample points are regularly distributed in the time-scale plane. In the general case, reconstruction algorithms have a greater complexity than in the regular case where key operations reduce to simple scalar multiplication. These two cases are detailed in the following subsections.

General Irregular Discrete Sampling Sets

Given the OCWT of a signal computed as the sampled output of a continuous wavelet filter bank, a procedure for the recovery of the original signal may be implemented in these distinct steps:

1. **(Resolving Time)** recovery of the continuous time outputs of the forward OCWT filter bank from the generally irregularly spaced time sample values; and
2. **(Resolving Scale)** recovery of the input signal through the weighted superposition of the recovered filter bank outputs from (i) followed by a linear “inverse” filter with a frequency response determined by G^{-1} .

This approach conceptually separates the overall underlying (two-index) wavelet frame into many sets (one for each scale) of (single-index) Paley-Wiener frames (viz. Section 6.5.1).

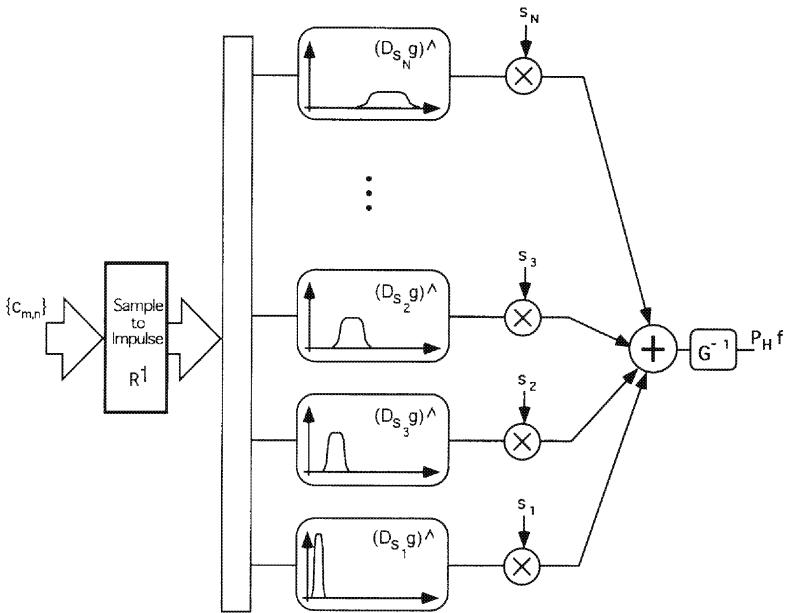


FIGURE 6.3. Inverse overcomplete wavelet transform as a synthesis filter bank.

To describe this approach in further detail, suppose that the analyzing function is well localized in frequency (e.g., bandlimited), and the number of filters in the bank (as in any practical implementation) is finite. Under these conditions, the OCWT filter bank frequency coverage, as exposed by G , is necessarily concentrated on a finite interval $[a, b]$. If the scale samples are chosen close enough so that $G > 0$ over $[a, b]$ then for bandlimited signals $f \in PW_{[a,b]}$, the OCWT filter bank continuous outputs collectively contain all the information in the original signal. In other words, no information due to the scale sampling is lost through the filter bank analysis. To ensure that no information is lost due to the time sampling it is necessary to place a constraint on the density of the time sample points as well.

In response to an input signal a bandlimited function $f \in PW_{[a,b]}$, the output y_m of the m th filter in an OCWT (with analyzing wavelet $g \in L^2(\mathbb{R})$) filter bank is

$$y_m = f * D_{s_m} \tilde{g}.$$

Thus, the output y_m has a Fourier transform that necessarily lives in the space $L^2(\text{supp } D_{s_m^{-1}} \tilde{g} \cap \text{supp } \hat{f})$. With

$$Q_m \triangleq \text{supp } D_{s_m^{-1}} \tilde{g} \cap \text{supp } \hat{f}$$

this means that $\widehat{y}_m \in L^2(Q_m)$ and the inner product on $L^2(Q_m)$ may be computed using the $L^2[a, b]$ inner product; that is,

$$\forall h \in L^2[a, b], \quad \langle h, y_m \rangle_{L^2(Q_m)} = \langle h, y_m \rangle_{L^2[a, b]},$$

since $Q_m \subseteq [a, b]$.

Using these facts, the following conditions ensure that a bandlimited function $f \in PW_{[a, b]}$ may be reconstructed from an OCWT filter bank with M filters whose output is sampled on the time-scale discrete sampling set $\Gamma = \{t_{m,n}, s_m\}_{m=1}^M$.

1. For each $m = 1, 2, \dots, M$, $\{e_{-t_{m,n}}\}_n$ is a frame for the (bandlimited) space $L^2(Q_m)$ with the frame operator S_m and correlation R_m ; and
2. for scalars A and B and almost all $\gamma \in [a, b]$,

$$0 < A \leq G(\gamma) \triangleq \sum_{m=1}^M |\widehat{g}(s_m^{-1}\gamma)|^2 \leq B < \infty.$$

Under the preceding assumptions an expression for the recovery of f from its OCWT Lf is now derived in a straightforward way. If $\widehat{f} \in L^2[a, b]$ then $\widehat{f} \cdot D_{s_m^{-1}}\widehat{g}$ is also in $L^2[a, b]$. From the first supposition and Theorem 3.13, for $m = 1, 2, \dots, M$,

$$\begin{aligned} \widehat{f} \cdot D_{s_m^{-1}}\widehat{g} &= \sum_n \left\langle \widehat{f} \cdot D_{s_m^{-1}}\widehat{g}, e_{-t_{m,n}} \right\rangle_{L^2(Q_m)} S_m^{-1} e_{-t_{m,n}} \\ &= \sum_n \left\langle \widehat{f} \cdot D_{s_m^{-1}}\widehat{g}, e_{-t_{m,n}} \right\rangle_{L^2[a, b]} S_m^{-1} e_{-t_{m,n}} \\ &= L_m^* R_m^\dagger L_m (\widehat{f} \cdot D_{s_m^{-1}}\widehat{g}). \end{aligned} \quad (6.4)$$

Noting that

$$\begin{aligned} \left\{ L_m(\widehat{f} \cdot D_{s_m^{-1}}\widehat{g}) \right\}_m &= \left\{ \left\langle \widehat{f} \cdot D_{s_m^{-1}}\widehat{g}, e_{-t_{m,n}} \right\rangle_{L^2[a, b]} \right\}_{m,n} \\ &= \{(W_g f)(t_{m,n}, s_m)\}_{m,n} = Lf \end{aligned}$$

and multiplying both sides of (6.4) by $s_m D_{s_m^{-1}}\widehat{g}$ and summing over all m yields

$$\begin{aligned} \sum_{m=1}^M \widehat{f} \cdot s_m |D_{s_m^{-1}}\widehat{g}|^2 &= \\ \widehat{f} \cdot \underbrace{\sum_{m=1}^M |D_{s_m^{-1}}\widehat{g}|^2}_{G} &= \sum_{m=1}^M L_m^* R_m^\dagger (Lf)_m s_m D_{s_m^{-1}}\widehat{g}. \end{aligned}$$

Because of the second supposition concerning G this further gives the expression for \hat{f} as

$$\hat{f} = G^{-1} \sum_{m=1}^M L_m^* R_m^\dagger (Lf)_m \cdot s_m D_{s_m^{-1}} \hat{g},$$

where $(Lf)_m \triangleq \{W_g f(t_m, s_{m,n})\}_n$. Clearly, the time domain version of f may then be recovered by computing the inverse Fourier transform of \hat{f} .

Figure 6.3 displays a schematic view of this reconstruction process. Dependent on the sample set Γ , the sample to impulse operation takes a two-dimensional coefficient sequence $c = \{c_{m,n}\}$ to an impulse “train” in the time-scale plane as

$$\sum_{m=1}^M (L_m^* R_m^\dagger c)^\vee = \sum_{m=1}^M \sum_n (R_m^\dagger c_{m,.})_n \cdot \delta(t - t_{m,n}) \cdot \delta(s - s_m).$$

For each m the frame correlation R_m is dependent on the time sampling $\{t_{m,n}\}_n$ associated with a particular filter m . In this context, the continuous output of each irregularly time sampled filter is reconstructed independently. Subsequently, each individual “continuous” reconstruction is linearly superposed and passed through the “inverse” filter having the frequency response G^{-1} over the frequency support of all functions of interest.

Reconstructions based on the frame algorithms discussed in Sections 3.4.7 and 7.1.7 may be employed to perform the operations of R_m^{-1} and R_m^\dagger , respectively.⁴ This is discussed further in Section 6.5.1. Alternatively, the same algorithm may be applied with an entire wavelet frame instead of to the frame of complex exponentials associated with each sampled filter output independently. This is discussed further in Section 6.5.2. It should also be mentioned that there are numerically efficient frame-based reconstruction algorithms that have been developed ([Grö93a] and [FGS95]) for such problems. For more information regarding irregular sampling theory in bandlimited spaces and with respect to wavelet and Gabor transforms the reader is referred⁵ to [BH90], [Ben92], [Ben93], [FG92], and [FG93]. Some recent results for irregular sampling in nonbandlimited spaces may be found in [AF97].

Semilog Regular Sampling Sets

From the previous section it is evident that the complexity of the inverse would be substantially reduced if the operators R_m^\dagger and G^{-1} were simple

⁴Recall that $R_m^\dagger = R_m^{-1}$ on the range of L_m .

⁵The NUmerical Harmonic Analysis Group (NUHAG) at the University of Vienna maintains a vast collection of original research in this area at the URL <http://tyche.mat.univie.ac.at/welcome>.

scalar multiplies. It turns out that restricting the sampling set Γ to be semilog regular (viz. Definition 6.2), has exactly this effect. In a semilog regular discrete sampling set time sample values $t_{m,n}$ are uniformly spaced (as a function of n) and scale sample values s_m are exponentially spaced. A sample set of this kind is described by

$$t_{m,n} = n\Delta \quad \text{and} \quad s_m = a_0^m,$$

where $\Delta > 0$ is the uniform (over all scales and time) sampling period and $a_0 > 1$ controls the overlap between filters. Figure 6.4 depicts the time-scale semilog regular sampling set $\Gamma(0.25, 1.1)$. Note that the scale axis is labeled in terms of corresponding frequency.

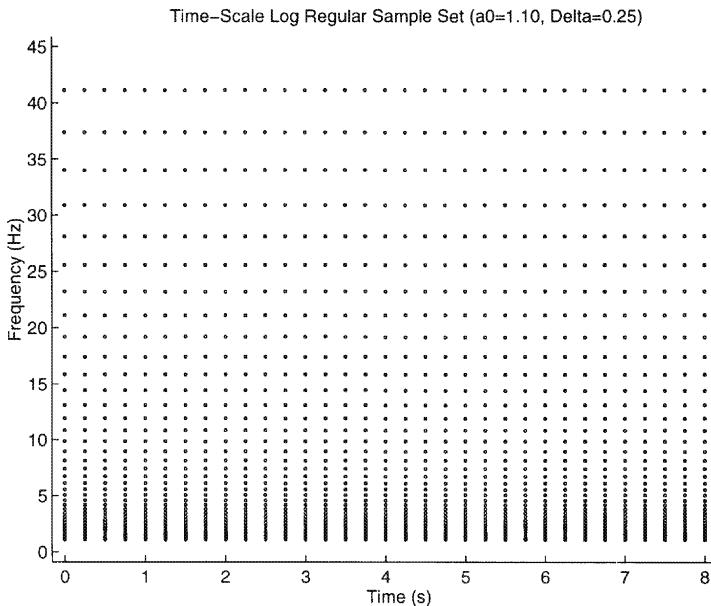


FIGURE 6.4. A time-scale semilog regular sampling set of the form of Equation (6.5) with a time sample period of 0.25 and a dilation factor of 1.1.

For highly redundant filter banks the value of a_0 should be near unity (e.g., $a_0 = 1.05$), and, in particular, much less than 2. A special case of a discrete sampling set of this form is determined by the two parameters a_0 and Δ and is given by

$$\Gamma(\Delta, a_0) \triangleq \{n\Delta\} \times \{a_0^m\}. \quad (6.5)$$

Such a discrete sampling set, although regular in the sense of uniform spacing over scales, leads, in general, to overcomplete OCWT representations.

Figure 6.4 shows a sample set of this form for the values of dilation factor $a_0 = 1.1$ and a sample period of $\Delta = 0.25$.

Some comments regarding a semilog regular discrete sampling set are in order.

1. Note for comparison that the dyadic sampling Γ_d of Equation (5.1) associated with the discrete wavelet transform is a discrete sampling set having an exponential scale sampling 2^n ($a_0 = 2$) and a scale dependent uniform time sampling with period $2^{-n}m$. This type of sampling set is regular with respect to the affine group structure.
2. At the cost of some accuracy, an alternative approach to irregular sampling in the time-scale plane may be implemented by interpolating schemes on a “fine enough” semilog regular discrete sampling set $\Gamma(\Delta, a_0)$.

Consider now the OCWT inverse in the case that the time-scale discrete sampling set is semilog regular in the sense previously described. It is shown that the operators R_m^\dagger and G^{-1} can indeed be approximated by simple scalar multiplies in this case. Because the time samples are uniformly spaced by Δ , the corresponding correlation inverse R_m^\dagger can be shown (Exercise 6.2) to reduce to multiplication by Δ ; that is,

$$\forall c \in L(\mathcal{H}), \quad R_m^{-1}c = \Delta \cdot c.$$

Restricting the regular discrete sampling set to be of the form (6.5) would further reduce the inverse complexity since R_m^{-1} would then be independent of m . Moreover, for values of $a_0 > 1$ that are close to 1 the filter bank frequency support function

$$G(\gamma) = \sum_m |\widehat{g}(a_0^{-m}\gamma)|^2$$

is nearly constant. This fact can be shown by approximating the integral (4.9) as

$$\begin{aligned} \text{constant} &= \int_{\mathbf{R}} |s|^{-1} |\widehat{g}(s^{-1}\gamma)|^2 ds \\ &\approx \sum_m |a_0|^{-m} |\widehat{g}(a_0^{-m}\gamma)|^2 \cdot (a_0^{m+1} - a_0^m) \\ &= (a_0 - 1) \sum_m |\widehat{g}(a_0^{-m}\gamma)|^2 = (a_0 - 1) \cdot G(\gamma). \end{aligned}$$

Thus G^{-1} is also a constant and the overall inverse OCWT on a semilog regular sample set has a substantially reduced complexity as compared to general irregular sample sets.

In practice, it is also necessary to confine analyses to a finite bandwidth. In terms of the wavelet filter bank implementation the practical constraint

translates into a finite number of filters that need to cover the bandwidth of interest. Thus, partial summations of the function G become directly relevant to the inverse computation. Fortunately, wavelets g that are essentially supported in frequency on an interval and have good frequency decay lead to partial summations G_{part} that will remain constant over a desired bandwidth. In symbols,

$$G_{\text{part}}(\gamma) = \sum_{m=m_1}^{m_2} |\widehat{g}(a_0^{-m}\gamma)|^2 \approx G(\gamma) \approx \text{constant}$$

over some interval of frequencies $[\gamma_1, \gamma_2]$ whose length $\gamma_2 - \gamma_1$ is dependent on $m_2 - m_1$ in a monotonically increasing fashion. This fact is illustrated in Figure 6.7. Thus, the constant G^{-1} may be estimated from G_{part} as

$$G^{-1} \approx \frac{1}{\max_{\gamma} G_{\text{part}}(\gamma)}.$$

6.3 Time–Frequency Localization and Wavelet Design

A wavelet transform draws its strength from its potential ability to localize the time–frequency coherent energy in a given signal in the time–scale plane. In turn, a wavelet transform’s time–frequency localizing ability is directly inherited from the analyzing wavelet. If the analyzing wavelet is not well localized in time and/or frequency then the corresponding wavelet transform will exhibit the same nonlocality. Because there is a fundamental tradeoff between the competing objectives of time localization and frequency localization, the design of the analyzing wavelet is a major concern with respect to the extraction of the time–frequency content within a signal of interest. This localization tradeoff is governed by the uncertainty principle and is discussed in the next section.

6.3.1 The Uncertainty Principle

Via the classical uncertainty principle it is well known that there is a fundamental tradeoff between the time and frequency localization of a signal. Localization in one domain necessarily comes at the cost of localization in the other. With respect to the design of jointly time–frequency well-localized analyzing wavelets, a specific mechanism for achieving this trade off is presented in the form of a parametrically described family of wavelets.

In order to quantify the idea behind the uncertainty principle it is necessary to specify a measure of the amount of *localization* that a signal possesses. Specifically what is wanted is a way to compute a single number

that quantifies how the energy in a signal is distributed over its domain. Domains of interest are time and frequency. This same measure is to be used to quantify both the time localization and frequency localization associated with a particular function.

Borrowing from the theory of probability, a standard measure of localization of a function of a single variable $f \in L^2(\mathbb{R})$ may be specified as the “variance”

$$\|(t - t_0) \cdot f(t)\|^2 = \int_{\mathbb{R}} (t - t_0)^2 \cdot |f(t)|^2 dt, \quad (6.6)$$

where t_0 is the first order moment

$$t_0 = \frac{\int_{\mathbb{R}} t \cdot |f(t)| dt}{\int_{\mathbb{R}} |f(t)| dt}.$$

If f were a valid probability distribution (having unit L^1 -norm) then these quantities would be the variance and mean, respectively, of the distribution in the strict probabilistic sense. In this more general definition of variance, the normalization requirement is relaxed. In any case, the larger the variance of a given signal, the less localized (more spread out) the signal is in its domain. Note that this measure of signal spread is more intuitively geared toward functions that are single humped in nature; that is, their main feature is an increase on the left to a central maximum and a decrease to the right. For signals that are not nearly single humped, the variance measure may be a misleading indicator of signal spread (viz. Exercise 6.6).

The uncertainty principle places a fixed positive lower bound on the product of the time variance and frequency variance for a given signal f . Thus, there is a fundamental constraint on the amount of simultaneous time and frequency localization that is possible. More precisely,

$$\|tf(t)\| \cdot \|\gamma\hat{f}(\gamma)\| \geq \frac{1}{4\pi} \|f\|^2. \quad (6.7)$$

The validity of the principle can be quickly verified for (differentiable) real-valued signals having decay faster than $t^{-1/2}$ by applying the Cauchy–Schwarz inequality to the product $tf(t)f'(t)$ to yield

$$|\langle tf(t), f'(t) \rangle| \leq \|tf(t)\| \cdot \|f'(t)\| = \|tf(t)\| \cdot \|j2\pi\gamma\hat{f}(\gamma)\|.$$

The inner product on the left side of this equation may be written as

$$\begin{aligned} \langle tf(t), f'(t) \rangle &= \int tf(t)f'(t) dt = \int t \cdot \frac{1}{2}\partial_t f^2(t) dt \\ &= \frac{1}{2} \left(tf^2(t) \Big|_{-\infty}^{\infty} - \int f^2(t) dt \right) = \frac{1}{2} \|f\|^2. \end{aligned}$$

This directly yields (6.7).

More generally, the uncertainty relationship between time and frequency localization may be better understood by looking to the underlying mathematical structure behind it. Namely, the two operations of monomial multiplication M and differentiation ∂ are key players. Simply stated, the uncertainty principle says that provided these operators satisfy certain conditions the product of the norms of these two operators acting on any function is bounded away from zero.⁶ Lemma 6.3 and Theorem 6.4 give the precise mathematical conditions that ∂ and M must satisfy for the uncertainty principle to hold. In Lemma 6.3 a general operator A is introduced which for the purposes of the uncertainty principle should be thought of as the composition $A = \partial M$. These operators are defined in the following.

Lemma 6.3 *Suppose that $A \in \mathcal{B}(\mathcal{H}, \mathcal{H})$ is a bounded linear operator mapping the Hilbert space \mathcal{H} to itself such that*

1. $A > 0$ (A is positive definite), and
2. $A + A^* = I$ (the sum of A and its adjoint is the identity).

Under these assumptions

$$\forall f \in \mathcal{H}, \quad \|f\|^2 = 2 \cdot \langle f, Af \rangle.$$

Proof: For every $f \in \mathcal{H}$,

$$\begin{aligned} \|f\|^2 &= \langle f, (A + A^*)f \rangle \\ &= \langle f, Af \rangle + \langle f, A^*f \rangle \\ &= \langle f, Af \rangle + \overline{\langle f, Af \rangle} \\ &= 2 \cdot \operatorname{Re}\{\langle f, Af \rangle\} = 2 \langle f, Af \rangle. \end{aligned}$$

■

The operator M is the *moment* (first order monomial multiplication) operator defined by

$$(Mf)(t) \stackrel{\triangle}{=} tf(t)$$

and the operator ∂ is the *derivative* operator

$$(\partial f)(t) \stackrel{\triangle}{=} \frac{d}{dt} f(t) = \dot{f}(t).$$

Note that these operators are Fourier duals in the sense that

$$(\partial f)^{\widehat{}} = j2\pi \cdot M \widehat{f}.$$

⁶The classical Heisenberg uncertainty principle may be formulated in this way where the operators ∂ and M are akin to position and momentum in physical systems.

Moreover, the modulation operator is self-adjoint

$$M^* = M$$

and the derivative operator is negatively self-adjoint

$$\partial^* = -\partial.$$

Using these expressions it is easily verified that

$$\partial M + (\partial M)^* = I.$$

Theorem 6.4 (Uncertainty Principle) *For all differentiable functions $f \in L^2(\mathbb{R})$*

$$\|tf(t)\| \cdot \|\gamma\widehat{f}(\gamma)\| \geq \frac{1}{4\pi} \|f\|^2$$

with equality being attained only by Gaussian signals of the form

$$f(t) = Ce^{-at^2},$$

for $a > 0$ and $C \in \mathbb{R}$.

Proof: Letting $A = \partial M$, Lemma 6.3 yields

$$\begin{aligned} \|f\|^2 &= 2|\langle f, \partial M f \rangle| \\ &= 2|\langle -\partial f, M f \rangle| \\ &\leq 2\|Mf\| \cdot \|\partial f\| \quad (\text{with equality } \iff 2aMf = \partial f \text{ for } a > 0) \\ &= 2\|Mf\| \cdot \|j2\pi\gamma\widehat{f}(\gamma)\| \\ &= 4\pi \cdot \|Mf\| \cdot \|\widehat{Mf}\|. \end{aligned}$$

The equality condition gives directly that

$$\dot{f}(t) = 2atf(t)$$

which is satisfied only by Gaussian signals of the form $f(t) = Ce^{-at^2}$. ■

6.3.2 Parametric Bandlimited Wavelets

In the vein of Section 4.2, a simple family of wavelets may be designed in the frequency domain and specified by three parameters, center frequency γ_c , bandwidth γ_b , and order N (these translate into the γ_i parameters used in (6.8)). The order parameter N is used as a mechanism to tradeoff time and frequency localization (6.9). This tradeoff is clearly illustrated in Figure 6.5. The figure shows examples of four specific PBL wavelets that correspond to order values of $N = 0, 3, 6$, and 9 with center frequency and bandwidth

both fixed at 20Hz. Frequency magnitudes (for positive frequencies) are plotted in the left graph of the figure (in a stacked fashion) and the time domain (real parts and magnitude envelopes) are plotted on the right (also in stacked fashion). From the figure it is evident that for larger values of N the frequency localization of the PBL wavelet increases whereas the time localization decreases. Although greater frequency localization comes necessarily at the cost of lesser time localization, the time localization does not suffer too badly. This is a design feature of the PBL wavelets.

PBL Wavelets

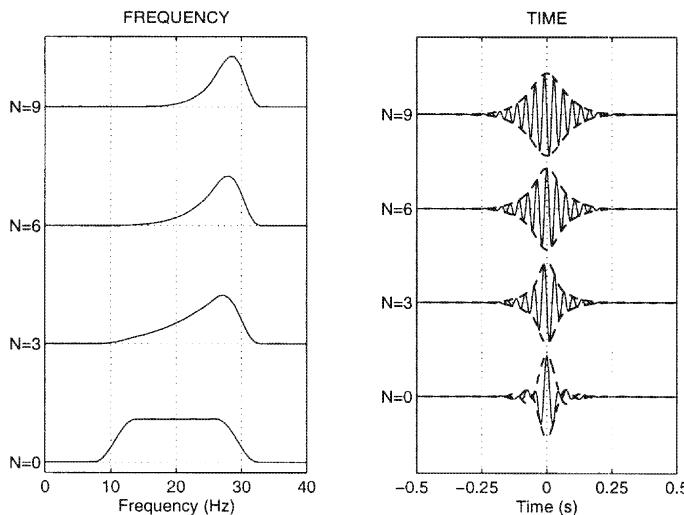


FIGURE 6.5. Parametric bandlimited wavelets in time and frequency for fixed center frequency of 20Hz and bandwidth of 20Hz and values of $N = 0, 3, 6$, and 9 .

Designed in the frequency domain, the filter transfer function of the analyzing PBL wavelet \hat{g}_0 has the following parametric form for order $N = 0$.

$$\hat{g}_0(\gamma) = \begin{cases} 0, & |\gamma| \leq \gamma_1, \\ \sin^2\left[\frac{\pi}{2\gamma_r}(\gamma - \gamma_1)\right], & |\gamma| \in [\gamma_1, \gamma_2), \\ 1, & |\gamma| \in [\gamma_2, \gamma_3), \\ \cos^2\left[\frac{\pi}{2\gamma_f}(\gamma - \gamma_3)\right], & |\gamma| \in [\gamma_3, \gamma_4), \\ 0, & \text{otherwise.} \end{cases} \quad (6.8)$$

Higher-order filter transfer functions, $\hat{g}_N, N > 0$, are derived from the zeroth-order filter as

$$\hat{g}_N(\gamma) = \gamma^N \cdot \hat{g}_0(\gamma). \quad (6.9)$$

6.4 OCWT Examples

A set of six pulsed signals $\{p_k\}_{k=1}^6$ with various frequency modulations

$$f_{k,l}, \quad k = 1, 2, \dots, 6, \quad l = 1, 2, \dots, L_k,$$

where L_k is the number of modulated components in the k th signal, has been synthesized to form a small data set for numerical demonstrations. Pulses are synthesized according to

$$p_k(t) = A(t) \cdot e^{j2\pi \sum_{l=1}^{L_k} \int_{-\infty}^t f_{k,l}(s) ds},$$

where A is the pulse envelope (constant over almost all the signal length with a relatively fast but continuous onset and offset). Over the signal duration, the modulations $f_{k,l}$ for each FM signal in the data set are displayed in Figure 6.6.

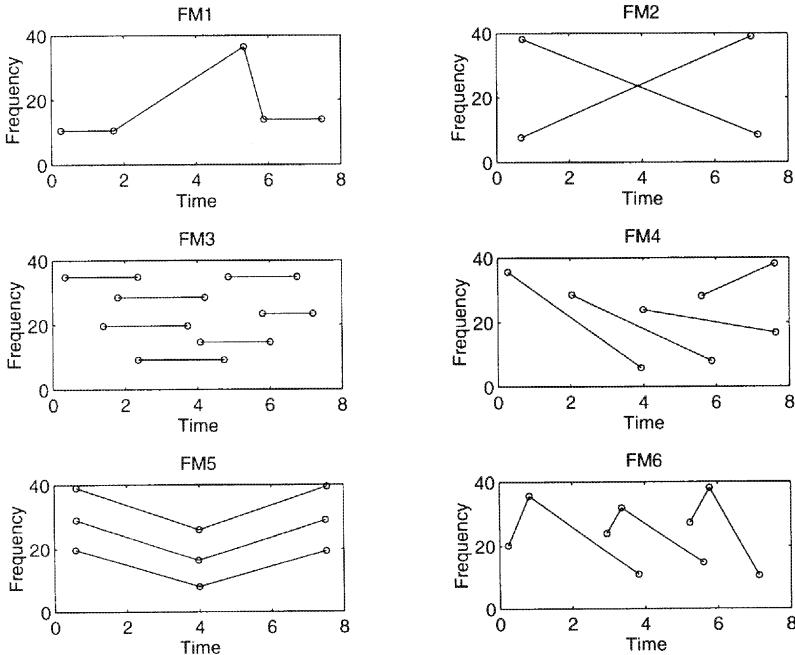


FIGURE 6.6. Frequency modulations of the six-signal synthetic test set.

Figure 6.7 shows a Morlet filter bank and Figure 6.8 shows a PBL filter bank. The Morlet filter bank is generated using Equation (4.1) with parameters center frequency $\gamma_c = 40$, bandwidth $\gamma_b = 5$, and $s_m = 1.03^m$,

for $m = 1, 2, \dots, 64$. The PBL filter bank is generated using (6.9) with parameters center frequency $\gamma_c = 50$, bandwidth $\gamma_b = 5$, order $N = 20$, and $s_m = 1.02^m$, for $m = 1, 2, \dots, 64$. These filter banks have been used to compute various OCWTs of synthetic FM signals.

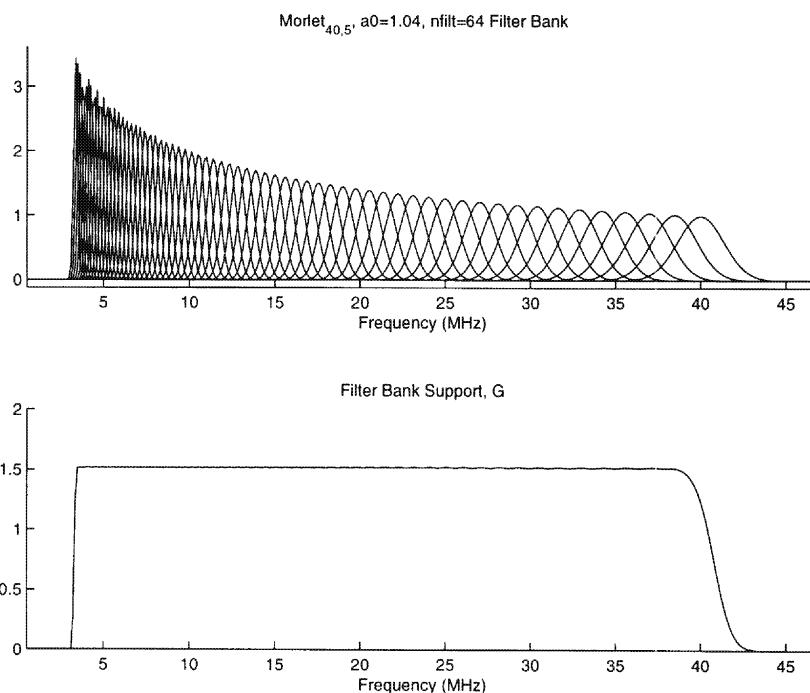


FIGURE 6.7. Frequency domain transfer functions of a Morlet filter bank ($M = 64$ filters) and the associated frequency support function of Equation (6.3).

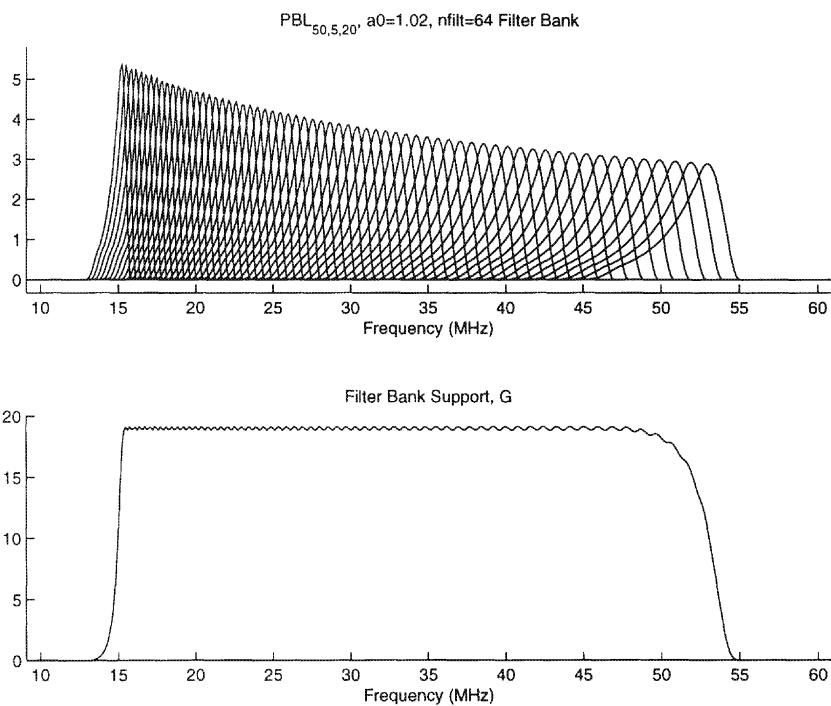


FIGURE 6.8. Frequency domain transfer functions of a PBL filter bank ($M = 64$ filters) and the associated frequency support function of Equation (6.3).

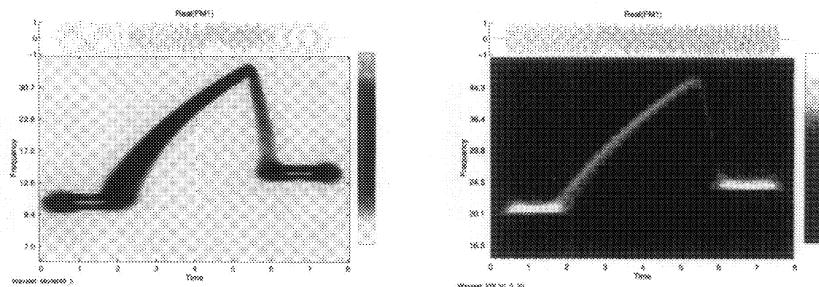


FIGURE 6.9. FM signal 1: (Left) Morlet OCWT; (right) PBL OCWT (signal upshifted in frequency by 10).

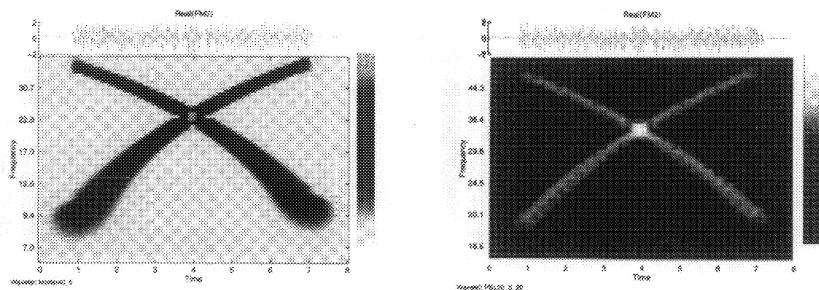


FIGURE 6.10. FM signal 2: (left) Morlet OCWT; (right) PBL OCWT (signal upshifted in frequency by 10).

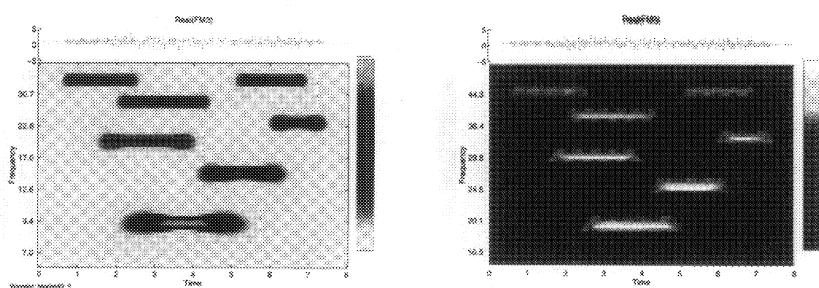


FIGURE 6.11. FM signal 3: (left) Morlet OCWT; (right) PBL OCWT (signal upshifted in frequency by 10).

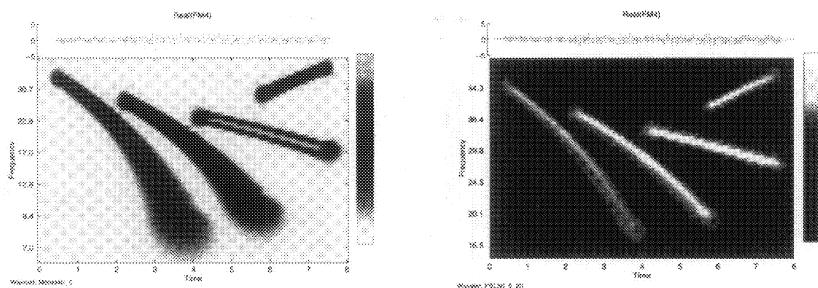


FIGURE 6.12. FM signal 4: (left) Morlet OCWT; (right) PBL OCWT (signal upshifted in frequency by 10).

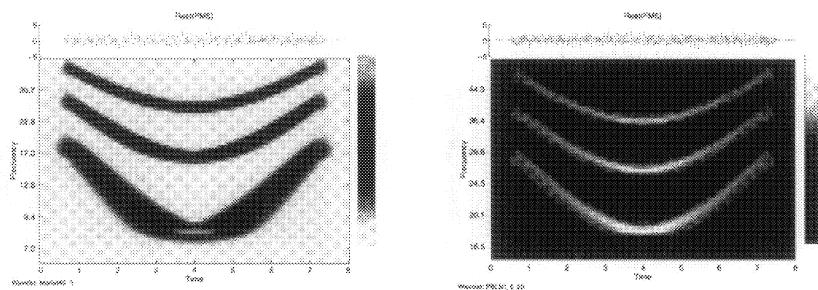


FIGURE 6.13. FM signal 5: (left) Morlet OCWT; (right) PBL OCWT (signal upshifted in frequency by 10).

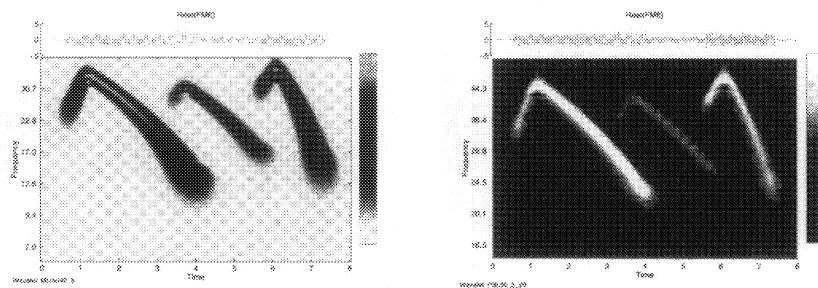


FIGURE 6.14. FM signal 6: (left) Morlet OCWT; (right) PBL OCWT (signal upshifted in frequency by 10).

Figures 6.9 through 6.14 display the OCWTs of the FM six-signal test set with respect to the Morlet and PBL filter banks shown in Figures 6.7 and 6.8, respectively. In the case of the PBL OCWT each signal is first upshifted in frequency by 10Hz⁷ before being passed through the filter bank.

Note how the energy in each signal is deposited along the well-defined FM trajectories of each signal. The associated spread of the energy along these trajectories is governed by the choice of analyzing wavelet and its localization in time and frequency. In general, lower frequencies have larger spread and higher frequencies have less spread. Better localization in the time or frequency domain of the analyzing wavelet will lead to a sharper ridge in the corresponding dimension of the time-scale plane.

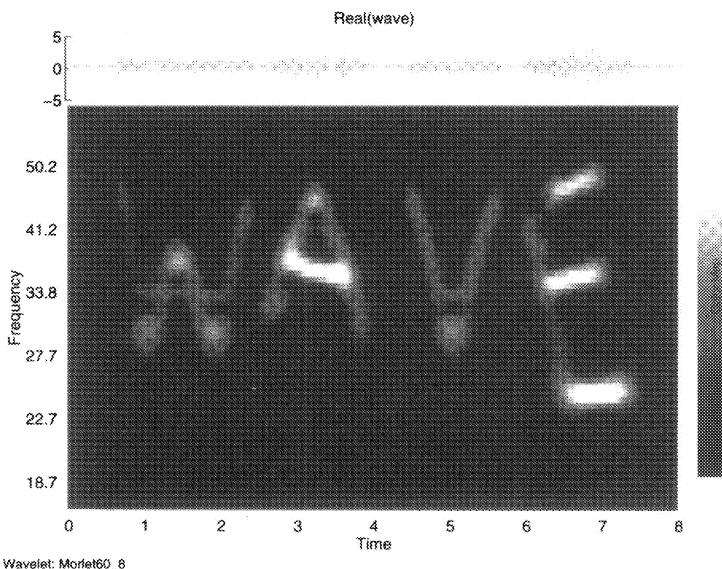


FIGURE 6.15. Morlet OCWT of FM signal "WAVE."

To illustrate further the frequency resolving ability of various OCWT filter banks many interesting signals may be constructed. Figures 6.15 and 6.16 show two “interesting” synthetic signals. In the former the signal has a frequency modulation that spells out the word “WAVE” and the latter has a modulation that resembles a “happy” face. This illustrates one *very* simple way that a signal may be constructed to carry information via its

⁷Explicit units of time and frequency have been intentionally unspecified allowing them to be simultaneously interpreted as any one of the following: (i) seconds (s) and Hertz (Hz), (ii) milliseconds (ms) and kiloHertz (kHz), or (iii) microseconds (μ s) and megahertz (MHz), depending on the application.

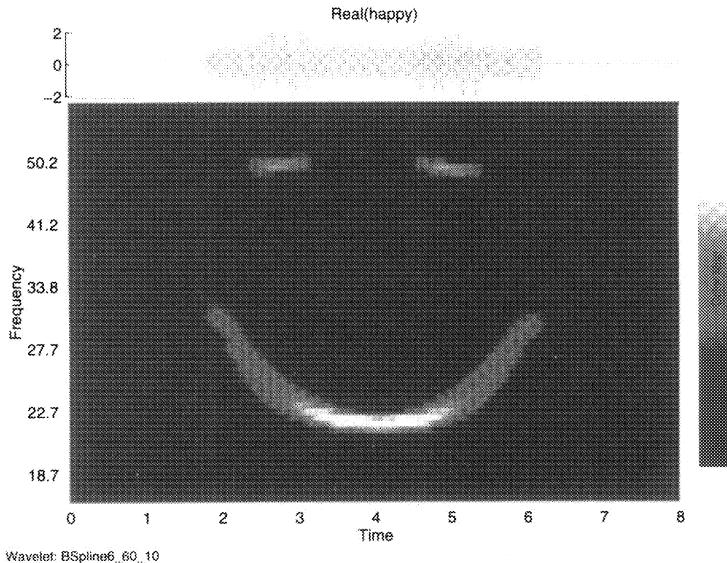


FIGURE 6.16. B-spline OCWT of FM signal “happy.”

frequency modulation and how the wavelet transform can be used to extract it.

6.5 Irregular Sampling and Frames

A key mechanism for the design of overcomplete signal representations and, in particular, the OCWT, is the, in general, irregular sampling of an associated continuous transform. Discussed in this section is the relationship between frames and irregular sampling in two main cases: direct sampling of a bandlimited function; and the sampling of a continuous wavelet transform of a general L^2 function. These lead, respectively, to the study of Paley–Wiener and wavelet frames.

6.5.1 Paley–Wiener Frames

General frames for PW_Ω

Consider a general one-dimensional sampling set $\Gamma = \{t_n\}$ made up of points from the real line \mathbb{R} and a signal f that is bandlimited by $\Omega > 0$ (i.e., $f \in PW_\Omega$). The main question of interest is

How is f recovered from its samples $\{f(t_n)\}$?

To answer this question the frame properties of the set of functions

$$\{\tau_{t_n} d_{2\pi\Omega}\}$$

are directly relevant. This set of functions has the representation

$$Lf = \{\langle f, \tau_{t_n} d_{2\pi\Omega} \rangle\} = \{f(t_n)\}$$

with the adjoint

$$L^*c = \sum c_n \tau_{t_n} d_{2\pi\Omega}.$$

A sufficient condition which ensures that f is recoverable from Lf is that

$$\{\tau_{t_n} d_{2\pi\Omega}\}$$

constitute a frame for PW_Ω . In this case L is a frame representation and from Algorithm 3.24 all $f \in PW_\Omega$ may be written as

$$f = L^*R^{-1}Lf.$$

By Plancherel and Parseval it is clear from the definition of a frame (viz. Equation (3.8)), that a set $\{\phi_n\}$ is a frame for \mathcal{H} if and only if $\{\widehat{\phi}_n\}$ is a frame for $\widehat{\mathcal{H}} \triangleq \{\widehat{f} : f \in \mathcal{H}\}$. With this notation $\widehat{PW}_\Omega = L^2[-\Omega, \Omega]$. Because of the basic Fourier relation

$$(\tau_{t_n} d_{2\pi\Omega})^\sim = e_{-t_n} 1_{[-\Omega, \Omega]},$$

the frame properties of the set $\{\tau_{t_n} d_{2\pi\Omega}\}$ in PW_Ω are the same as the frame properties of the set $\{e_{-t_n} 1_{[-\Omega, \Omega]}\}$ in $L^2[-\Omega, \Omega]$. Thus, the frame properties of such sets of complex exponentials are of direct interest in sampling theory for bandlimited functions.

By definition (viz., Definition 3.11), a sequence constituting a frame for a Hilbert space must reside in that space. Notationally, this requirement is abused by saying that $\{e_{-t_n}\}$ is a frame for $L^2[a, b]$ meaning that

$$\{e_{-t_n} 1_{[a, b]}\}$$

is a frame for $L^2[a, b]$. Here $a < b$ are real numbers. It is easy to see that if $\{e_{-t_n}\}$ is a frame for $L^2[a, b]$ then $\{e_{-t_n}\}$ is a frame for $L^2[a + c, b + c]$ for any real c . In the time domain this corresponds to the statement that $\{\tau_{t_n} e_{(a+b)/2} d_{2\pi[(b-a)/2]}\}$ is a frame for $PW_{[a, b]}$ if and only if

$$\{\tau_{t_n} e_c e_{(a+b)/2} d_{2\pi[(b-a)/2]}\}$$

is a frame for $PW_{[a+c, b+c]}$.

Early on, frame properties of sets of complex exponentials $\{e_{-t_n}\}$ were studied by Duffin and Schaeffer ([DS52]). One of their fundamental results

is a sufficient condition on a sampling set $\Gamma = \{t_n\}$ to generate a frame of complex exponentials for $L^2[-\Omega, \Omega]$; that is, a condition that ensures that the set $\{e_{-t_n}\}$ forms a frame for $L^2[-\Omega, \Omega]$. This condition is related to the notion of uniform density. A set $\Gamma = \{t_n\}$ has uniform density $\rho(\Gamma)$ if the following conditions hold.

- (i) Γ is uniformly discrete; that is, there is a $d > 0$ so that for all $n \neq m$, $|t_n - t_m| > d$, and
- (ii) $|t_n - n(\rho(\Gamma))^{-1}| < L$ for some constant $L < \infty$.

Since $|n\Delta - m\Delta| = |n - m|\Delta$ and $|n\Delta - n(1/\Delta)^{-1}| = 0$ a uniform sampling set Γ_Δ with sampling period Δ is a uniformly dense sampling set with density $\rho(\Gamma_\Delta) = 1/\Delta$.

Theorem 6.5 (Duffin–Schaeffer [DS52, Theorem I]) *If $\Gamma = \{t_n\}$ is a uniformly dense sequence with uniform density $\rho(\Gamma) > 2\Omega > 0$ then $\{e_{-t_n}\}$ is a frame for $L^2[-\Omega, \Omega]$.*

The following corollary is true by the Fourier transform isomorphism relating $L^2[-\Omega, \Omega]$ and PW_Ω .

Corollary 6.6 *If $\Gamma = \{t_n\}$ is a uniformly dense sequence with uniform density $\rho(\Gamma) > 2\Omega > 0$ then $\{\tau_{t_n}d_{2\pi\Omega}\}$ is a frame for PW_Ω .*

It is interesting to note that a sequence which is uniformly dense may have arbitrarily large, although finite, gaps. The largest such gap is the parameter L in the definition of a uniformly dense sequence. For a uniformly dense sequence large gaps, however, must be compensated for by portions of high density elsewhere in the set. The uniform discreteness of such a sampling set prevents it from having any finite cluster points. Thus there can be no interval in which the density of sampling points is unbounded. Roughly speaking one can think of a uniformly dense sequence of uniform density Δ^{-1} as a tessellation of the uniform sampling $\{n\Delta\}$. By tessellation of the set $\{n\Delta\}$ it is meant that a second set $\{t_n\}$ in which each sampling point t_n may be identified with a point $n\Delta$ through the relation $|t_n - n\Delta| < L$. Figure 6.17 illustrates a uniformly dense sequence derived as a tessellation of uniform sequence of density 1 on a compact interval.

Jaffard ([Jaf91]) has characterized sampling sets $\Gamma = \{t_n\}$ that generate frames of complex exponentials $\{e_{-t_n}\}$ for $L^2[-\Omega, \Omega]$ in terms of unions of uniformly discrete and uniformly dense sampling sequences.

Consider collections of functions of the form $\{\tau_{t_n}g\}$. In the previous discussion the function g was taken to be $g = d_{2\pi\Omega}$ but more general functions can be used. In this case, the frame properties of the collection generated by the translations of g are of interest. This leads again to consideration of the frame properties of the complex exponentials since

$$(\tau_{t_n}g)^\wedge = e_{-t_n}\widehat{g}.$$

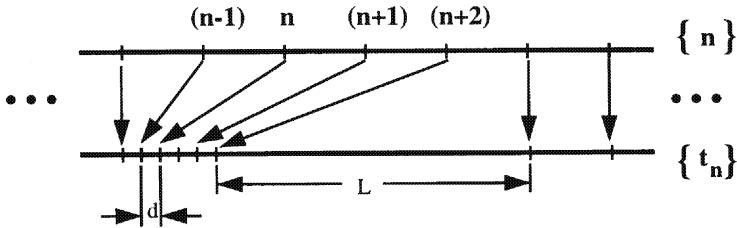


FIGURE 6.17. A uniformly dense sequence generated as a tessellation of a uniform sequence on a compact interval.

For arbitrary functions g there is no reason to expect that this collection will form a frame for $L^2[-\Omega, \Omega]$. In fact such a collection could only possibly form a frame for $L^2(\text{supp } \widehat{g})$. This is because if f is a function such that the support of \widehat{f} is exclusive of the support of \widehat{g} (i.e. $\text{supp } \widehat{f} \cap \text{supp } \widehat{g} = \emptyset$), then

$$\langle \widehat{f}, e_{-t_n} \widehat{g} \rangle = \langle \widehat{f} \widehat{g}, e_{-t_n} \rangle = 0.$$

Thus, if the essential support of \widehat{g} does not cover $[-\Omega, \Omega]$ then there are nonzero functions $\widehat{f} \in L^2[-\Omega, \Omega]$ for which there is no $A > 0$ so that

$$A \|\widehat{f}\|^2 \leq \sum |\langle \widehat{f}, e_{-t_n} \widehat{g} \rangle|^2.$$

Exact frames for PW_Ω

Definition 6.7 (Kadec–Levinson sequence) *A real sequence $\{t_n\}$ is a Kadec–Levinson sequence if for some $\Omega > 0$ the condition*

$$\sup_n |t_n - \frac{n}{2\Omega}| < \frac{1}{4} \left(\frac{1}{2\Omega} \right)$$

is satisfied.

Paley and Wiener first dealt with the question of when a set of complex exponentials $\{e_{-t_n}\}$ forms a basis for the space $L^2[-\Omega, \Omega]$. Levinson ([Lev40]) showed that the constant 0.25 is the smallest such constant that ensures the completeness of $\{e_{-t_n}\}$ in $L^2[-\Omega, \Omega]$. Later Kadec ([Kad64]) gave a direct proof that $\{e_{-t_n}\}$ is an exact frame. These results are stated in Theorem 6.8 which appears in [Ben92, Theorem 34].

Theorem 6.8 *Suppose the sampling set $\{t_n\}$ is a Kadec–Levinson sequence for a given $\Omega > 0$. Then $\{e_{-t_n}\}$ is an exact frame for $L^2[-\Omega, \Omega]$.*

A Kadec–Levinson sequence $\Gamma = \{t_n\}$ for $\Omega > 0$ is uniformly discrete with uniform density $\rho(\Gamma) = 2\Omega$. Clearly, if the sequence were uniformly

discrete the uniform density would be 2Ω . To see that a KL sequence is uniformly discrete assume without loss of generality that $t_m > t_n$ and write

$$\begin{aligned} t_m - t_n &\geq \inf(t_m - t_n) \\ &= \inf_m t_m - \sup_n t_n \\ &\geq (m - \frac{1}{4})\Delta - (n + \frac{1}{4})\Delta = (m - n - \frac{1}{2})\Delta, \end{aligned}$$

so that $|t_m - t_n| > d = \frac{1}{2}\Delta$. Figure 6.18 depicts sequences that are of the Kadec–Levinson type. Each element t_n in the sequence is restricted to lie within a region of length $1/(4\Omega)$ centered at $n\Delta$. This region is indicated by parentheses in the figure.

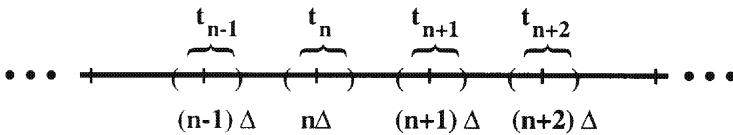


FIGURE 6.18. A Kadec–Levinson sequence.

6.5.2 Wavelet Frames

In the case of the Paley–Wiener type frames discussed previously frames were generated from just the translations of a single function g . Adding a dilation operation yields potential wavelet frames. Wavelet frames are generated by the dilation and translation of a single⁸ function g . As usual, the discrete dilation and translation values have no constraints on their spacing except that the associated atomic functions form a frame for the space of interest. In this context the general theory presented in Chapter 3 is specialized to wavelet frames and the associated representation L is the OCWT.

Consider a two-dimensional sampling set $\Gamma = (t_{m,n}, s_m)$ contained in the affine group $\mathcal{G}_A = \mathbb{R} \times \mathbb{R} \setminus \{0\}$ and a signal $f \in L^2(\mathbb{R})$. The main question of interest is

How is f recovered from its wavelet transform samples $\{W_g f(t_{m,n}, s_m)\}$, that is, its OCWT?

In direct analogy to the Paley–Wiener one-dimensional sampling case, the

⁸Families of functions generated from a single function are said to be *coherent*.

frame properties of collections of the form

$$\{\tau_{t_{m,n}} D_{s_m} g\}$$

determine the answer to this question. If this collection were a wavelet frame for $L^2(\mathbb{R})$ then f would be recoverable from its OCWT via a frame reconstruction algorithm, for example, Algorithm 3.24.

Because translation in time acts as modulation by a complex exponential in frequency, the frame properties of sets of complex exponentials play as equally important a role in wavelet sampling as they do in direct sampling. The following lemma describes how frames of complex exponentials together with dilation can be used to generate frames for large spaces.

Lemma 6.9 *Let $\text{supp } \widehat{g} = [a, b]$ where $a < b$ are real numbers. Let $\{t_{m,n}\}$ and $\{s_m\}$ be real sequences. Suppose that*

$$\{e_{-t_{m,n}}\} \text{ is a frame for } \widehat{V}_m \triangleq L^2[s_m a, s_m b]$$

with bounds $s_m A_m$ and $s_m B_m$. If there are constants A and B such that for $\gamma \in \mathbb{R}$

$$0 < A \leq G(\gamma) \triangleq \sum_m |\widehat{g}(s_m^{-1}\gamma)|^2 \leq B < \infty \quad \text{a.e.}$$

then

$$\{\tau_{t_{m,n}} D_{s_m} g\} \text{ is a frame for } V \triangleq \bigcup_m PW_{[s_m a, s_m b]}$$

with bounds $A \cdot (\inf_m A_m)$ and $B \cdot (\sup_m B_m)$.

Proof: For $f \in V$ write

$$\begin{aligned} \sum_{m,n} |\langle f, \tau_{t_{m,n}} D_{s_m} g \rangle|^2 &= \sum_{m,n} |\langle \widehat{f}, e_{-t_{m,n}}(D_{s_m} g) \rangle|^2 \quad (\text{Parseval}) \\ &= \sum_{m,n} |\langle \widehat{f}(\overline{D_{s_m} g}), e_{-t_{m,n}} \rangle|^2 \\ &\leq \sum_m s_m B_m \|\widehat{f}(\overline{D_{s_m} g})\|^2 \\ &= \sum_m s_m B_m \int |\widehat{f}(\gamma) \overline{(\widehat{D_{s_m} g})(\gamma)}|^2 d\gamma \\ &\leq \left(\sup_m B_m \right) \sum_m \int s_m \cdot |\widehat{f}(\gamma)|^2 |(D_{s_m^{-1}} \widehat{g})(\gamma)|^2 d\gamma \\ &= \left(\sup_m B_m \right) \int |\widehat{f}(\gamma)|^2 \sum_m |\widehat{g}(s_m^{-1}\gamma)|^2 d\gamma \\ &\leq B \left(\sup_m B_m \right) \int |\widehat{f}(\gamma)|^2 d\gamma \end{aligned}$$

$$\begin{aligned}
&= B \left(\sup_m B_m \right) \|\widehat{f}\|^2 \\
&= B \left(\sup_m B_m \right) \|f\|^2 \quad (\text{Plancherel}).
\end{aligned}$$

The first inequality follows from the assumption that $\{e_{-t_{m,n}}\}$ frames \widehat{V}_m and the fact that $\widehat{f}(D_{s_m}g) \in \widehat{V}_m$ for each m . The lower bound is analogous. ■

It is clear from this lemma that for $\{\tau_{t_{m,n}} D_{s_m} g\}$ to be a tight frame then the function G must be constant a.e. The following corollary addresses the tightness issue.

Corollary 6.10 *With the assumptions of Lemma 6.9 $\{\tau_{t_{m,n}} D_{s_m} g\}$ is a tight frame for $V \triangleq \bigcup_m PW_{[s_m a, s_m b]}$ if and only if G is constant a.e. and for all m the frame $\{e_{-t_{m,n}}\}_n$ is tight with $s_m A_m = s_m B_m = C$.*

Theorem 6.11 gives conditions on a sampling set in \mathcal{G}_A to give rise to a wavelet frame for $H_+^2(\mathbb{R})$. It is a direct consequence of Lemma 6.9.

Theorem 6.11 *Let $0 < a < b$ be two strictly positive real numbers and let $g \in PW_{[a,b]}$ and $\widehat{g} \in L^\infty(\mathbb{R})$ with $\text{supp } \widehat{g} = [a, b]$.*

Suppose that $\{t_{m,n}\} \subseteq \mathbb{R}$ and $\{s_m\} \subseteq \mathbb{R}^+$ are two sequences such that

$$\{e_{-t_{m,n}}\} \text{ is a frame for } L^2[s_m a, s_m b]$$

and there is a d such that

$$\forall m, \quad 1 < d \leq \frac{s_{m+1}}{s_m} \leq \frac{b}{a}.$$

Then

$$\{\tau_{t_{m,n}} D_{s_m} g\} \text{ is a frame for } H_+^2(\mathbb{R}).$$

Proof: From Lemma 6.9

$$G = \sum_m s_m \cdot |D_{s_m^{-1}} \widehat{g}|^2.$$

Since the support of $D_{s_m^{-1}} \widehat{g}$ is $I_m \triangleq [s_m a, s_m b]$, the condition on $\{s_m\}$ that $s_{m+1}a \leq s_m b$ ensures that $I_m \cup I_{m+1} = [s_m a, s_{m+1} b]$ and, in particular, has no gaps of nonzero measure. Since $g \in L^\infty(\mathbb{R})$ one may conclude that there are constants A and B so that $0 < A \leq G \leq B < \infty$ a.e. Clearly,

$$V = \lim_{M \rightarrow \infty} \bigcup_{m=-M}^M (PW_{[s_m a, s_m b]}) = PW_{(0, \infty)} = H_+^2(\mathbb{R}).$$

Thus, by Lemma 6.9 $\{\tau_{t_{m,n}} D_{s_m} g\}$ is a frame for $H_+^2(\mathbb{R})$. ■

Note that a related theorem which concludes that $\{\tau_{t_m, n} D_{s_m} g\}$ is a frame for all of $L^2(\mathbb{R})$ can be arrived at in various ways. Two ways in particular are the following: consider negative scales in addition to positive scales; or allow less restrictive analyzing functions g with some negative frequency support. Finally, it should be noted that [Grö93b] provides a more general theorem for irregular *weighted* wavelet systems in which conditions on a sampling set in \mathcal{G}_A are asserted that ensure that the weighted system is a frame for $L^2(\mathbb{R})$.

6.5.3 Irregular Sampling in the Time-Scale Domain

Wavelet frames are related to the irregular samples of a function's wavelet transform (i.e., its OCWT), in this section. Although the discussion is directed specifically at the wavelet transform, it hints at the greater generality given in the reproducing kernel space view of Section 4.6.4.

An underlying theme in the theory of sampling is the connection between boundedness, reproducing kernels, and sampling expansions. In Section 4.6.4 it is shown that, via the Riesz representation theorem, the condition that all members of a functional Hilbert space satisfy a pointwise bound implies a reproducing kernel for that space. Thus, loosely speaking, boundedness leads to reproducing kernels. In Section 4.6.5 such a pointwise bound is easily established (via Cauchy-Schwarz) for members of $W_g(\mathcal{H})$. The reproducing kernel for $W_g(\mathcal{H})$ is given explicitly in Theorem 4.13 as

$$K = c_g^{-2} \cdot (W_g g),$$

where

$$c_g^2 = \int |\gamma|^{-1} |\widehat{g}(\gamma)|^2 d\gamma.$$

The accompanying reproducing formula is

$$\forall F \in W_g(\mathcal{H}), \quad F(x) = (F * K)(x) = \langle F, T_x K \rangle_{L^2_\mu(\mathcal{G})}$$

for all $x \in \mathcal{G}_A$. From this one may conclude that knowledge of the samples $\{F(x_n)\}$ of a function $F \in W_g(\mathcal{H})$ is equivalent to knowledge of the inner products $\{\langle F, T_{x_n} K \rangle_{L^2_\mu(\mathcal{G})}\}$.

This observation leads to the consideration of discretizations

$$\mathcal{L} : W_g(\mathcal{H}) \mapsto \ell^2(\mathbb{Z})$$

defined as

$$\mathcal{L}F = \left\{ \langle F, T_{x_n} K \rangle_{L^2_\mu(\mathcal{G})} \right\} = \{F(x_n)\}$$

with the adjoint $\mathcal{L}^* : \ell^2(\mathbb{Z}) \mapsto W_g(\mathcal{H})$

$$\mathcal{L}^*c = \sum c_n T_{x_n} K,$$

where $F \in W_g(\mathcal{H})$ and $c = \{c_n\} \in \ell^2(\mathbb{Z})$. Recall that for a fixed $x_0 \in \mathcal{G}_A$ the x_0 -translation $T_{x_0}F$ of a function $F \in L_\mu^2(\mathcal{G})$ is $T_{x_0}F(x) = F(x_0^{-1}x)$. Because $\mathcal{L}F$ consists of irregular samples of F , this discretization allows us to develop expansions that recover F from its irregular samples in \mathcal{G}_A provided the collection

$$\{T_{x_n}K\}$$

is a frame for a large enough subspace of $W_g(\mathcal{H})$. For example, if $\{T_{x_n}K\}$ were a frame for $W_g(\mathcal{H})$ then any $F \in W_g(\mathcal{H})$ could be reconstructed (viz. (3.24)), from $\mathcal{L}F$ as

$$F = \mathcal{L}^* \mathcal{R}^{-1}(\mathcal{L}F),$$

where $\mathcal{R} = \mathcal{L}\mathcal{L}^*$ is the frame correlation. Since $\mathcal{L}F = \{F(x_n)\}$ this is a sampling expansion for reconstructing a function F that is an element of the reproducing kernel Hilbert space $W_g(\mathcal{H})$ from its sample values. Together with the previous boundedness arguments, this observation justifies the statement that boundedness implies RKHS implies sampling expansions.

The main interest is, however, not to reconstruct wavelet transforms from their irregular samples. The main interest is, rather, to develop discrete representations of signals from \mathcal{H} with associated reconstruction algorithms. It turns out that irregular samples of wavelet transforms are in fact exactly the type of discrete representation that is sought. The two are essentially related by the wavelet transform and its “inverse.” To make this notion of inverse precise, Proposition 6.12 identifies the proper spaces in which the wavelet transform has an inverse.

Proposition 6.12 *Let $\Pi(x) = \tau_t D_s$ for $x = (t, s) \in \mathcal{G}_A$. The wavelet transform W_g has a well-defined inverse when restricted to a subspace $\mathcal{H}_W \stackrel{\triangle}{=} \text{span}\{\Pi(x)g\}_{x \in \mathcal{G}_A}$ mapping onto its closed range $W_g(\mathcal{H}_W) = W_g(L^2(\mathbb{R}))$.*

Proof: Clearly, W_g maps onto its range. It remains to show that $W_g : \mathcal{H}_W \mapsto W_g(\mathcal{H}_W)$ is one-to-one. Let $f_1, f_2 \in \mathcal{H}_W$ and suppose $W_g f_1 = W_g f_2$. This means that for all $x \in \mathcal{G}_A$

$$\langle f_1, \Pi(x)g \rangle = \langle f_2, \Pi(x)g \rangle$$

or

$$\langle f_1 - f_2, \Pi(x)g \rangle = 0.$$

From this one concludes that $(f_1 - f_2) \perp \mathcal{H}_W$. Since \mathcal{H}_W is a linear subspace $(f_1 - f_2) \in \mathcal{H}_W$. Thus, $f_1 - f_2$ must be zero. It has been shown that $W_g : \mathcal{H}_W \mapsto W_g(\mathcal{H}_W)$ is bijective and therefore has a well-defined inverse $W_g^{-1} : W_g(\mathcal{H}_W) \mapsto \mathcal{H}_W$. ■

It can be shown by approximate identity arguments ([HW89]) that the wavelet transform W_g has a well-defined inverse on its range. In fact, if $f \in \mathcal{H}_W$ then formally

$$f = \frac{1}{c_g^2} \left\langle (W_g f)(x), \overline{\Pi(x)g} \right\rangle_{L_\mu^2(\mathcal{G})}$$

in L_μ^2 . Figure 6.19 depicts the wavelet transform mapping W_g and its properties on different domains and ranges of interest.

In light of Proposition 6.12 and the operator \mathcal{L}^* it is clear that only functions which are contained in \mathcal{H}_W have any hope of being represented by linear combinations of $\{\pi(x_n)g\}$. Thus, it is required that $\mathcal{H} \subseteq \mathcal{H}_W$.

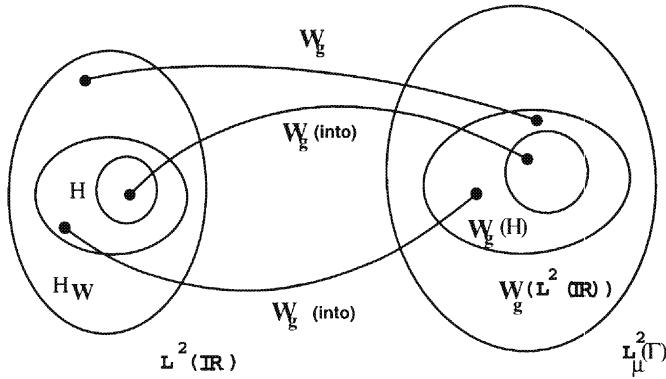


FIGURE 6.19. The wavelet transform W_g on different domains.

To make the connection between irregular sampling in $W_g(\mathcal{H})$ and discrete representation in \mathcal{H} fix $x_0 \in \mathcal{G}_A$ and consider the function $\Pi(x_0)g \in \mathcal{H}$. Since $\Pi(x_0)g$ is an element in \mathcal{H} one may compute its wavelet transform as

$$\begin{aligned} (W_g \Pi(x_0)g)(x) &= \langle \Pi(x_0)g, \Pi(x)g \rangle \\ &= \langle g, \Pi(x_0^{-1})\Pi(x)g \rangle \\ &= \langle g, \Pi(x_0^{-1}x)g \rangle \\ &= (W_g g)(x_0^{-1}x) = T_{x_0} W_g g(x) = c_g^2 T_{x_0} K(x). \end{aligned}$$

From this calculation it follows that for a fixed $x \in \mathcal{G}_A$ the functions $\Pi(x)g$ and $c_g^2 T_x K$ are wavelet transform pairs. This relationship is denoted as

$$\forall x \in \mathcal{G}_A, \quad \Pi(x)g \xleftrightarrow{V_g} c_g^2 T_x K.$$

The sampling operator \mathcal{L} whose domain is $W_g(\mathcal{H})$ has an associated discretization operator L whose domain is \mathcal{H} . Let $F = W_g f$. Then by the

reproducing property and Corollary 4.11

$$F(x_n) = \langle F, T_{x_n} K \rangle_{L^2_\mu(\mathcal{G})} = \langle f, \Pi(x_n)g \rangle.$$

This suggests the discretization $L : \mathcal{H} \mapsto \ell^2(\mathbb{Z})$ defined as

$$Lf = \{F(x_n)\}$$

with the adjoint

$$L^*c = \sum c_n \Pi(x_n)g.$$

Thus, questions about the frame properties of the sequence $\{T_{x_n} K\}$ in $L^2_\mu(\mathcal{G})$ may be translated to questions about the frame properties of the sequence $\{\Pi(x_n)g\}$ in \mathcal{H} .

Object	Range	Domain
Space	$W_g(\mathcal{H}_W) = \text{span } \{T_x K\}$	$\mathcal{H}_W = \text{span } \{\Pi(x)g\}$
Kernel	K	g
Translation	T_x	τ_x
Group/Representation	$(\mathcal{G}_A, \cdot), \Pi(x)$	$(\mathbb{R}, +), \tau_t$
Atoms	$\{T_{x_n} K\}$	$\{\pi(x_n)g\}$
Discretization	$\mathcal{L}F = \{F * K(x_n)\}$	$Lf = \{\langle f, \Pi(x_n)g \rangle\}$
Adjoint	$\mathcal{L}^*c = \sum c_n T_{x_n} K$	$L^*c = \sum c_n \pi(x_n)g$

TABLE 6.1. Wavelet transform relations.

It has been seen that the OCWT is a complete characterization of a signal provided the underlying wavelet atoms constitute frames for a large enough space. Owing to the generality of the underlying square integrable group representation, the flexibility of this approach to signal representation is tremendous. This is manifested in the large number of free parameters in the representation. Each choice of wavelet analyzing function g and discrete set leads to a discrete representation. Roughly speaking, there are as many different discrete representations as there are discrete sets and functions g .

Problems

6.1 Suppose that $\{\phi_n\}$ frames the Hilbert space \mathcal{H} with bounds A and B .

- (a) Show that $\{D_a \phi_n\}$ frames the Hilbert space $D_a \mathcal{H}$ with the same bounds.
- (b) If $\{g_m\}$ is a set of functions in $L^2(\mathbb{R})$ such that

$$0 < a < G \triangleq \sum |g_m|^2 < b < \infty$$

show that the set $\{g_m \cdot \phi_n\}_{m,n}$ is a frame for \mathcal{H} with bounds aA and bB .

6.2 Let L be the frame representation for a Hilbert space \mathcal{H} with respect to the harmonic complex exponentials $\{e_{-n\Delta}\}$ for $\Delta > 0$. If R is the associated frame correlation show that the inverse R^{-1} applied to a coefficient sequence $c \in L(\mathcal{H})$ is

$$R^{-1}c = \Delta c.$$

6.3 Let g be the first-order frequency B-spline wavelet given in the frequency domain as

$$\widehat{g}(\gamma) = \begin{cases} 1 - |\gamma - 2|, & |\gamma - 2| < 1, \\ 0, & \text{else.} \end{cases}$$

Given desired positive frame bounds A and B and a frequency range $(\gamma_0, \gamma_1) \subset \mathbb{R}^+$ find the largest values of $a_0 > 1$, $m_1 \in \mathbb{Z}$, and the smallest value of $m_2 \in \mathbb{Z}$ such that

$$\forall \gamma \in [\gamma_0, \gamma_1], \quad A \leq \sum_{m=m_1}^{m_2} |\widehat{g}(a_0^{-m} \gamma)|^2 \leq B.$$

Compute specific values of a_0 , m_1 and m_2 for $A = B = 1$ and $(\gamma_0, \gamma_1) = (10, 20)$.

6.4 Using the setup in Exercise 6.3 find $\Delta > 0$ such that

$$\{\tau_{n\Delta} D_{a_0^m} g\}_{m=m_1}^{m_2}$$

tightly frames $\text{PW}_{[10,20]}$. Write down a reconstruction formula for the OCWT inverse on $\text{PW}_{[10,20]}$; that is, determine an expression for the recovery of f from $\{(W_g f)(n\Delta, a_0^m)\}$.

Hint: Use Theorem 2.4.

6.5 Let $\{s_k\}_{k=1}^M$ be an increasing set of scale values such that

$$1 < s_1 < s_2 < \dots < s_M < \infty$$

and let $g \in PW_{[a,b]}$. Find necessary conditions on the scale sequence $\{s_k\}_{k=1}^M$ for the scale discretized wavelet transform $W_g f(t, s_m)$ of a function $f \in L^2(c, d)$ to be invertible (on $L^2(c, d)$). Here $b > a$ and $d > c$ are all positive scalars and $t \in \mathbb{R}$.

6.6 Compute the variance via Equation (6.6) of the following signals.

- (a) $f(t) = 1_{[-a,a]}$ for $a > 0$.
- (b) $f(t) = \begin{cases} 1 - \frac{|t|}{a}, & |t| < a, \\ 0, & \text{else} \end{cases}$ for $a > 0$.
- (c) $f(t) = e^{-\pi(t-10)^2}$.
- (d) $f(t) = e^{-\pi(t+10)^2}$.
- (e) $f(t) = e^{-\pi(t-10)^2} + e^{-\pi(t+10)^2}$.

6.7 Compute the Fourier transforms of the signals in Exercise 6.6 and verify the uncertainty principle for each.

6.8 Prove Corollary 6.6.

6.9 Let $\{t_n\}$ be a sequence given by

$$t_n = \begin{cases} \frac{(n-\frac{1}{8})}{2\Omega}, & n \text{ odd} \\ \frac{(n+\frac{1}{8})}{2\Omega}, & n \text{ even.} \end{cases}$$

Does there exist a function $f \in PW_\Omega$ such that $\forall n$, $f(t_n) = 0$ and $f(t) \neq 0$ a.e.?

6.10 *Oversampling in PW_Ω .* Let $\Delta < 1/(2\Omega)$. Using Theorems 6.5 and 2.4 show that the fine uniform sampling of signals in PW_Ω ,

$$Lf = \{f(n\Delta)\},$$

has an associated tight and *overcomplete* frame

$$\{\tau_{n\Delta} d_{2\pi\Omega}\}$$

with frame bounds $A = B = \Delta^{-1}$.

6.11 *Weighted OCWT is tight.* Let $a, b \in (0, \infty)$. Show that if

$$\forall \gamma \in [a, b], \quad \sum_m |\widehat{g}(a_0^{-m} \gamma)|^2 = A > 0,$$

then the set of functions

$$\left\{ a_0^{m/2} \tau_{n\Delta} D_{a_0^m} g \right\}_{m,n}$$

forms a tight overcomplete frame for $PW_{[a,b]}$ provided that $\Delta < 1/(2(b-a))$. What are the bounds on the *unweighted* frame

$$\left\{ \tau_{n\Delta} D_{a_0^m} g \right\}_{m,n}?$$

7

Wavelet Signal Processing

A basic motivation behind transform methods is the idea that some sorts of processing are better (or perhaps only possibly) achieved in the transform domain rather than in the original signal domain. In this sense, the utility of a transform is measured by its ability to facilitate desired signal processing tasks in the transform domain via algorithms that are digitally tractable, computationally efficient, concise, and noise robust. The efficacy of general wavelet transforms comes from the fact that wavelet domain algorithms exhibit all of these benefits when dealing with signals that are characterized by their time–frequency behavior. This chapter explores applications of overcomplete wavelet transforms in problems of data compression, noise suppression, digital communication, and signal identification.

In light of the tremendous depth and breadth of applicability of wavelet techniques in signal processing, this chapter only touches on a few of the many potential applications of wavelets and wavelet theory. Additional applications and different aspects of these same applications may be found in [Ben96], [AU96], [SN96], [VK95], [AS96], and [You93], among a vast array of many many others.

7.1 Noise Suppression

Noise in signals and systems is, by definition, unwanted and problematic; unfortunately, it is also unavoidable in real applications. In this section, a frame-based approach to the problem of suppressing noise in corrupted signals is developed. A general suppression method is formulated using the fundamental principle that noise may be viewed as *incoherence* with respect to a set of atomic functions. Mathematical considerations limit attention to atomic sets that form frames for large enough spaces. The concept of coherence suggests a natural thresholding procedure that operates in the frame coefficient domain to suppress incoherence in the signal domain. Attention is further limited to overcomplete wavelet frames (via the OCWT) and shown to represent signals with characteristic time–frequency behavior with a high degree of coherence.

In the following subsections, coherence is first discussed qualitatively and then developed into a quantitative theory in Section 7.1.5. There, Theorem 7.8 serves as the centerpiece of the theory that gives bounds on error levels as a function of coherence.

7.1.1 Noise Domains

With respect to frame representations of analog signals, there are certain domains in which noise may perturb a signal:

1. the (analog) signal domain, and
2. the (discrete) coefficient domain.

Respectively, these two domains may be further precisely specified as

1. a Hilbert space $\mathcal{H} \subseteq L^2(\mathbb{R})$, and
2. the signal domain's image under L ; that is, $L(\mathcal{H}) \subseteq \ell^2(\mathbb{Z})$,

where L is the frame representation operator (viz. Section 3.4.2, page 40).

Sources of noise in the analog signal domain generally come from unwanted and unavoidable factors in a signal environment, for example, cockpit noise in an airplane or thermal noise in a weak radio transmission. Sources of noise in the coefficient domain are quite different. Because of their digital nature they are susceptible to perturbations due to quantization and/or bit errors. From the standpoint of the analog world, coefficient domain noise is only of concern inasmuch as it reflects its disturbance back to the analog domain. After all, if a certain type of perturbation in a coefficient sequence has little effect (as measured via some reconstruction algorithm) on the analog signal that it represents then there is little need to be concerned with it. With this perspective, the term *noise suppression* is used in the following to mean the suppression of noise in the analog domain; although noise robustness in the coefficient domain is also of auxiliary concern to the extent that it supports robust discrete representation.

7.1.2 Problem

Implicit in the intent of noise suppression is a means to separate the *noise* w from the *signal* f_* through some processing of the observed linear mixture

$$f = f_* + \sigma \cdot w, \quad (7.1)$$

where σ is a noise level and w is a unit energy noise process. Specifically, w is a random process such that for each time t , $w(t)$ is a random variable that is normally distributed with zero mean and unit variance. Furthermore, for

any pair of distinct times $t_1 \neq t_2$ the random variables $w(t_1)$ and $w(t_2)$ are statistically independent.¹

Distinguishing between noise and signal given only the observed signal f is an impossible task in its own right. Some additional information or context is needed. To provide context, one may look to sets of atomic functions and define the concept of *coherence*. Simply put, a signal is coherent with respect to a set of functions if its (inner product) representation with respect to that set is *succinct* in the sense that relatively few coefficients in the representation domain have large magnitude. The concept of coherence is made precise in Section 7.1.4.

Accordingly, noise may be considered as a lack of coherence or *incoherence* with respect to an atomic set. Coherence may further be related to information-carrying capacity with respect to a decoder with knowledge of the atomic set. In this case, highly coherent signals possess an inherent large capacity to contain and convey information as opposed to highly incoherent signals such as white noise, which have little or no capacity to hold information.

Since the signals of interest here are assumed to be described by their time–frequency behavior, families of wavelets present themselves as natural candidates for atomic sets. Coherence with respect to a wavelet atomic set may be given the distinction of *time–frequency coherence* and is described further in Section 7.1.4. In general, time–frequency coherent functions are those that are well described by their time–frequency behavior.

With the previous background established, the main question addressed here is stated as follows.

How is a good approximation to a (time–frequency) coherent signal recovered from an *incoherently* perturbed version?

The term “good” is used loosely here. Attempts to specify quality precisely include considering L^2 error; however, subjective and perceptual metrics may be, in many applications, more appropriate. Here, only L^2 error metrics are considered.

7.1.3 Approach

Thresholding (of some type) in the wavelet coefficient domain is one of the main approaches that have been applied to the noise suppression problem ([Don93] and [Don95]). A main emphasis in previous research is the use of thresholding orthonormal wavelet representations to suppress incoherence.

¹In other words, the random process w is an independent and identically distributed (i.i.d.) process. It should be noted that this assumption is not too critical (i.e., it may be relaxed), in the general approach to noise suppression presented here.

Here, thresholding in the overcomplete wavelet domain is used to suppress incoherence.

Thresholding

To attain the goal of noise suppression in the signal domain, a general and predominant transform method uses thresholding as its main component. In this approach, a corrupted signal is transformed with respect to a set of functions (to yield a coefficient (inner product) representation); then, the smaller magnitude coefficients are de-emphasized; and, finally, the processed coefficients are inverse-transformed back to the original signal domain to yield a new processed signal. If the set of functions is chosen appropriately, this approach can be shown to perform near optimally ([Don95]) in the sense of providing the best mean square error estimate of the uncorrupted signal under a constraint of smoothness. In this section the thresholding approach is discussed from the perspective of coherence.

Consider the following setup. Let $\mathcal{H} \subseteq L^2(\mathbb{R})$ be a Hilbert space of interest and $\{\phi_n\} \subseteq \mathcal{H}$ be a frame for \mathcal{H} with frame representation operator L . As before, let $f_* \in \mathcal{H}$ be coherent with respect to $\{\phi_n\}$ and let f be a noise-corrupted version of f_* ; that is, Equation (7.1)

$$f = f_* + \sigma \cdot w$$

holds where w is noncoherent with respect to $\{\phi_n\}$.

In Sections 3.4.7 and 7.1.8, it is established that if $\{\phi_n\}$ is a frame for all of $L^2(\mathbb{R})$ then

$$f = L^* R^\dagger L f$$

perfectly recovers the noisy signal. Since the main objective is to reject the noise portion w , direct application of $L^* R^\dagger$ to the observation $L f$ replicates the noise as well as it replicates the signal. The question naturally arises as to whether there is some processing that may be performed in the coefficient domain $L(\mathcal{H})$ which will act as a noise suppressant in the signal domain \mathcal{H} . Coefficient thresholding is, in fact, one appropriate processing method. The thresholding process is now explained in detail.

A threshold operator² F_δ is defined with respect to the representation $c_* = L f_*$ of a particular signal $f_* \in \mathcal{H}$. It nullifies or truncates coefficient sequences in places where the representation $L f_*$ of f_* has small magnitude coefficients less than the threshold δ (a positive real number). The threshold operator is defined formally in Definition 7.1.

Definition 7.1 (Threshold Operator) *Let $\{\phi_n\}$ be a frame for \mathcal{H} and take any $c_* \in \ell^2(\mathbb{Z})$ and (threshold) $\delta > 0$. A threshold operator $F_\delta = F_{\delta, c_*}$ is defined as*

$$F_\delta : L(\mathcal{H}) \rightarrow \ell^2(\mathbb{Z}),$$

² “F” is for “finite.”

where

$$(F_\delta c)_n \triangleq \begin{cases} c_n, & |(c_*)_n| \geq \delta \\ 0, & \text{otherwise.} \end{cases} \quad (7.2)$$

Note that a thresholded element of $L(\mathcal{H})$ is in general no longer in $L(\mathcal{H})$ (viz. Exercise 7.1). It is easy to establish that F_δ is a linear bounded operator. In fact, $\|F_\delta\| = 1$. As a matter of notation the threshold operator F_{δ,c_*} for $c_* \in L(\mathcal{H})$ is sometimes written as F_{δ,f_*} where it is understood that c_* is given by the unique coefficient sequence satisfying $f_* = L^*c_*$.

Reconstruction from Thresholded Representation

Why should thresholding be expected to perform noise suppression?

A reconstruction procedure that starts not from the whole sequence Lf but from the truncated sequence $F_{\delta,f}Lf$ is straightforwardly given by

$$f_\delta \triangleq L^*R^\dagger F_{\delta,f}Lf.$$

Applied to a coherent and incoherent combination f , the operator $L^*R^\dagger F_{\delta,f}$ works to recover the coherent portion of the mixture and suppress the incoherent portion.

To see this, let the δ -thresholded version of the sequence Lf be denoted by $c_\delta \triangleq F_{\delta,f}Lf$. Expanding f and using the linearity of the operators $F_{\delta,f}$ and L we have

$$c_\delta = F_{\delta,f}L(f_* + w) = F_{\delta,f}Lf_* + F_{\delta,f}Lw.$$

The hope of this scheme for noise suppression lies in the expectations that

- (a) for a coherent signal f_* the truncation $F_{\delta,f}Lf_* \approx Lf_*$, and
- (b) for a noncoherent signal w the truncation $F_{\delta,f}Lw \approx 0$.

Consequently it is expected that $c_\delta \approx Lf_*$. In this case, c_δ may be viewed as a noisy version of Lf and Algorithm 7.13 may be used with initial data $c_0 = c_\delta$ to perform the noise suppression.

If $\{\phi_n\}$ is a frame for \mathcal{H} with frame representation operator L then (by definition) the following norm equivalence property holds. Namely, there are positive scalars A and B such that

$$\forall f \in \mathcal{H}, \quad A\|f\|^2 \leq \|Lf\|^2 \leq B\|f\|^2.$$

Such a relationship may be interpreted as enforcing an approximate energy transfer between the domains \mathcal{H} and $L(\mathcal{H})$. In other words, for all signals $f \in \mathcal{H}$, a scaled (by a factor between A and B) amount of the energy $\|f\|^2$ in the signal domain is *redistributed* in the coefficient domain.

How that energy is redistributed, however, is dependent on the signal's degree of coherence with the underlying frame $\{\phi_n\}$. In fact, for a signal f_*

which is coherent with respect to the frame $\{\phi_n\}$, the frame representation norm equivalence necessarily implies that these few coefficients must contain most of the signal energy and hence have a relatively large magnitude. Similarly, a pure noise signal w , being incoherent with respect to the set $\{\phi_n\}$, must have a frame representation in which the noise energy is spread out over a very large number of coefficients. Hence, these coefficients must have a relatively small magnitude.

Thresholding of the discrete representation is the key operation that may be identified with the suppression of noise. Letting $L = L_\Phi$, the wavelet representation of f is

$$(Lf)_n = \langle f, \phi_n \rangle = \underbrace{\langle f_*, \phi_n \rangle}_{\text{large}} + \sigma \cdot \underbrace{\langle w, \phi_n \rangle}_{\text{small}}.$$

By the assumption of coherence, the term due to noise must necessarily be small and the term due to the coherent portion f_* must be large. Thus, for an appropriately chosen threshold δ , the contribution due to the noise will be nullified whereas preserving the contribution due to the coherent portion. In light of this, the small coefficients in Lf due to the noncoherent portion w will be suppressed while the larger coefficients due to the coherent portion f will be preserved under truncation. Thus, the iterative Algorithm 7.13 initialized with $c_0 = F_{\delta,f} Lf$ is in fact a technique for the suppression of noise ([TB94b]).

Even though the main objective is to suppress signal domain noise, a thresholding approach, as outlined here and detailed in Section 7.1.3, is one that operates in the coefficient domain. Thresholding intentionally introduces a disturbance directly in the coefficient domain. In this case perturbed representations need not remain in the range $L(\mathcal{H})$ of the frame representation L . Thus, for iterative algorithms aimed at reconstructing a signal from its frame representation to be successful, care must be taken to ensure that the reconstruction algorithm will converge on all of $\ell^2(\mathbb{Z})$. This issue is addressed in Section 7.1.7.

Wavelet Processing

Model

A general model for noise suppression using wavelets is depicted in Figure 7.1. Suppose that

$$\Psi \triangleq \{\psi_{m,n} : \psi_{m,n} = \tau_{t_{m,n}} D_{s_m} g\}$$

is a wavelet frame for a Hilbert space \mathcal{H} of interest where g is the analyzing wavelet and $\{t_{m,n}, s_m\}$ is a discrete set in the time-scale plane.

Major elements and the processing flow of the model are explained in the following.

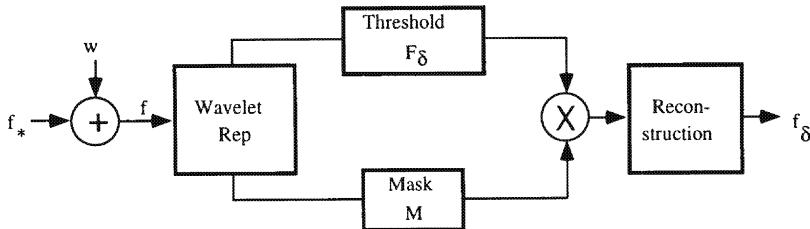


FIGURE 7.1. Noise model and incoherence suppression processing.

1. **(Signal f_*)** A time-frequency coherent signal, $f_* \in \mathcal{H}$, with respect to the wavelet frame Ψ is input into the model.
2. **(Noise w)** Corrupting noise w is additively mixed with f_* and represents an incoherent disturbance with respect to Ψ .
3. **(Wavelet Representation L_Ψ)** The observed corrupted signal f is transformed to the wavelet domain to yield the wavelet coefficients $L_\Psi f$.
4. **(Threshold Operator F_δ)** Thresholding is performed so as to zero out small magnitude wavelet coefficients and retain the value of large magnitude coefficients. In this case, the general threshold operator F_{δ, c_*} of Equation (7.2) is specialized to use a c_* which is the same as its argument (i.e., it performs $F_{\delta, c}$). Written out explicitly this gives

$$(F_\delta c)_{m,n} = \begin{cases} c_{m,n}, & |c_{m,n}| > \delta, \\ 0, & \text{otherwise} \end{cases},$$

where $c \in L(\mathcal{H}) \subseteq \ell^2(\mathbb{Z}^2)$ (c.f., Equation (7.2)).

5. **(Masking Operator M)** A mask can be used to effectively perform time-frequency filtering. Like operators, masks operate on coefficients in such a way as to preserve certain coefficients and zero out other coefficients. The difference is that threshold operators zero out only small coefficients, whereas masks may be defined so as to zero out arbitrary coefficients; that is,

$$(Mc)_{m,n} = \begin{cases} c_{m,n}, & (m, n) \in Q, \\ 0, & \text{otherwise,} \end{cases}$$

where $Q \subseteq \mathbb{Z}^2$ is fixed but arbitrary. Thus, mask operators are more general than threshold operators. A mask is useful in cases in which information-carrying coefficients are identifiable in ways other than their magnitude. In particular, as used in the software, masks allow a user to manually specify coefficients to preserve in a transform.

For signals with characteristic time–frequency behavior this ability is especially useful.

6. **(Reconstruction $L_\Psi^* R_\Psi^\dagger$)** Thresholded coefficients are used to construct a noise-suppressed version via an appropriate reconstruction algorithm. There are two basic types of reconstruction algorithms discussed here, iterative and closed form, and these are elaborated in the following.

Reconstruction Algorithms

Complexity of the reconstruction algorithms necessarily depends on the representation L_Ψ , and on the wavelet frame Ψ , and, in turn, on the underlying time–frequency sample set that defines it (viz. Section 5.1). The cases that are distinguished are

1. generally irregular time–frequency sampling sets; and
2. semilog regular time–frequency sampling.

Respectively, these lead to two different types of reconstruction algorithms, an iterative method and a closed form method. Following the presentation of previous material, the treatment of the general case is mostly theoretical in nature whereas the semilog-regular case is developed with practical and fast digital implementation in mind. The latter case is also illustrated numerically.

For general wavelet representations, with possibly irregular sampling sets in the time frequency domain, reconstruction algorithms that are iterative in nature may be formulated. In this case, a noise-suppressed version of the noisy signal is constructed via an iterative recovery procedure that starts with $F_\delta L_\Psi f$. This algorithm is detailed in Section 3.4.

For semilog-regular sampling sets the forward and inverse OCWT may be utilized for representation and reconstruction, respectively.

7.1.4 Coherence

Intuitively, the noise in a signal is that part which lacks structure or coherence. For example, in audio signals noise can be readily identified with *incoherent* garbled or hissing sounds. In general, additive white noise is a noncoherent disturbance. Implied in the use of the term “coherence” is some point of reference with which other points are coherent. Here, the view that a set of properly chosen atomic functions serves as an appropriate point of reference is adopted. In this view, inner product representation of signals with respect to the atomic set facilitates the defining line between noise (incoherence) and signal (coherence). As such, a coherent signal is one that exhibits a concentration of energy in the representation domain and an incoherent signal is one whose energy is diffusely spread throughout the representation domain.

In speech, examples of coherent structures are the phonemic primitives that have been learned by the individual; and a pertinent collection of appropriate primitive functions may be extracted from the study of the cochlear mechanics of the ear ([BT92]). Interestingly, these cochlear atomic functions are (to a good approximation) sets of wavelets ([BT93], [YWS92], [All85], [PK75]). This fact suggests that wavelet atomic functions are, in some evolutionary sense, optimal for processing sound.

In the remainder of this section, coherence is first discussed in the general case where the atomic functions form a frame for a space of interest. Second, the discussion is specialized to the notion of *time-frequency coherence*, where the atomic functions are additionally restricted to be well localized in time and frequency. Time-frequency coherence is in fact the main idea that is exploited in the approach to noise and interference suppression in signals. Implicit in the concept of time-frequency coherence is a set of atomic functions by which all coherent structures are *easily* constructed. Families of wavelets provide exactly this type of atomic set.

Signal Noise and Frame Coherence

In qualitative terms, a signal f is coherent with respect to an atomic set $\{\phi_n\}$ if the energy $\|Lf\|^2$ in the inner product representation is concentrated, that is, well localized in the representation domain.

If it is further required that f may be well approximated by a linear combination of members from an atomic set then, clearly, the set cannot be chosen arbitrarily. Moreover, for practical implementations an additional requirement is that such approximations be carried out in a numerically stable manner. Both of these requirements are satisfied for atomic sets $\{\phi_n\}$ that form frames for large enough Hilbert spaces, that is, ones that contain all signals of interest.

For these reasons attention is limited to collections $\{\phi_n\}$ that are frames. Consequently, considered as noise are those signals that are incoherent with respect to the frame $\{\phi_n\}$ and considered as “nonnoisy” are those signals that are coherent with respect to the frame. This view of coherent signal versus incoherent noise admits a thresholding procedure in the frame representation domain for the recovery of a signal embedded in noise. This procedure is outlined in Section 7.1.3.

Coherence further implies that a signal may be well approximated by a relatively small linear combination of atoms (viz. Theorem 7.8; cf. Exercises 7.2 and 7.3). Mallat and Zhang ([MZ93]) have used the related approach of matching pursuits to find concise signal decompositions from arbitrary “dictionaries” of functions.

To make the idea of coherence quantitatively precise, formal coherence measures may be constructed using the concept of a coherence distribution. Definitions 7.2 and 7.4 give these definitions, respectively.

Definition 7.2 (Coherence) Let $\{\phi_n\}$ be a frame for the space \mathcal{H} with representation L and fix $f_* \in \mathcal{H}$.

1. **(Coherence Functional)** For a fixed value of $\delta > 0$, Coh_δ is the mapping

$$\begin{aligned} Coh_\delta : \quad \mathcal{H} &\rightarrow [0, \infty) \\ f &\mapsto \frac{\|F_{\delta, f_*} Lf\|^2}{\|Lf\|^2}. \end{aligned}$$

2. **(Coherence Distribution)** For a specific $f_* \in \mathcal{H}$ the coherence distribution (CD) $Coh_\delta f_*$ is a function of δ given by

$$Coh_\delta f_* \triangleq \frac{\|F_{\delta, f_*} Lf_*\|^2}{\|Lf_*\|^2}.$$

As a function of δ , the coherence distribution (CD)³ describes how the energy in a δ -truncated representation decays. Intuitively, a long decay indicates a high degree of coherence with respect to the atoms $\{\phi_n\}$. Conversely, a fast decay indicates a low degree of coherence. Thus, given a known noise example w , it is desirable to have an atomic set $\{\phi_n\}$ to which w is not highly coherent so that the CD of w decays rapidly.

To yield a single scalar value that measures this degree of coherence, the L^1 norm of the CD as a function of threshold serves nicely. In addition, it is also useful to have a measure of the relative degree of coherence between two signals. Absolute and relative scalar measures of coherence are given formally in Definition 7.4.

Note that the coherence, as a functional, is *nonlinear*. In particular, this functional is neither additive (i.e., in general, the coherence of the sum of two signals is not equal to the sum of the coherences), nor is it homogeneous (i.e., the coherence of a multiplicatively scaled signal is not equal to the scaled coherence). These issues are addressed in Exercise 7.5. As a consequence of its definition, a coherence distribution must satisfy the properties given in Fact 7.3.

Fact 7.3 (Coherence Distribution Properties) If Coh_δ is the coherence distribution/functional with respect to the frame $\{\phi_n\}$ for \mathcal{H} of Definition 7.2 then it satisfies the properties:

1. **(bounded)**

$$\forall \delta \geq 0, \quad 0 \leq Coh_\delta f \leq 1;$$

2. **(monotonically decreasing)**

$$\delta_1 < \delta_2 \implies Coh_{\delta_1} f \geq Coh_{\delta_2} f;$$

³In cases where the dependence of the CD on the underlying frame $\{\phi_n\}$ is important, the coherence is written explicitly as $Coh_{\delta, \{\phi_n\}}$.

3. (closed range)

$$\lim_{\delta \rightarrow 0} \text{Coh}_\delta f = 1,$$

and

$$\lim_{\delta \rightarrow \|Lf\|_\infty} \text{Coh}_\delta f = 0;$$

4. (scaling)

$$\text{Coh}_\delta(af) = \text{Coh}_{(\delta|a|^{-1})}f,$$

for all $a \in \mathbb{C}$.

Proof: Exercise.

The following definition provides two simple scalar measures of the degree of coherence that signals have with respect to a given frame and to each other.

Definition 7.4 (Coherence Measure) Let Coh_δ be the coherence distribution given in Definition 7.2. Measures of absolute and relative coherence are defined as

1. (Absolute Coherence)

$$\text{COH}(f) \triangleq \|\text{Coh}_\delta f\|_1 = \int_0^\infty |\text{Coh}_\delta f| d\delta,$$

and

2. (Relative Coherence)

$$\text{RCOH}(f, g) \triangleq \frac{\text{COH}(f)}{\text{COH}(g)} = \frac{\|\text{Coh}_\delta f\|_1}{\|\text{Coh}_\delta g\|_1}.$$

An interesting property of the absolute coherence is that it is homogeneous; that is, for all $a \in \mathbb{C}$,

$$\text{COH}(af) = |a| \cdot \text{COH}(f).$$

This means that an amplified ($a > 1$) function af is more coherent (with respect to a frame) than its unscaled original f . This points to the need for a relative measure of coherence. Since the original motivation is to separate noise from signal, it makes sense to measure the degree of coherence of signal relative to the coherence of a pure noise signal. In cases in which signal f_* and noise w are known separately, the relative coherence $\text{RCOH}(f_*, w)$ between them may be computed to yield a *coherence signal-to-noise ratio* (CSNR); that is,

$$\text{CSNR} = \text{RCOH}(f_*, w) \triangleq \frac{\|\text{Coh}_{\delta, \{\phi_n\}} f_*\|_1}{\|\text{Coh}_{\delta, \{\phi_n\}} w\|_1}.$$

The CSNR is a useful indicator of noise separability for different representations, and thus, different atomic sets $\{\phi_n\}$ (frames). Frames with a large CSNR give good separation between noise and signal. This idea is explored further in the following section where signals (nonnoise) are considered to be those defined by their time–frequency behavior.

Time–Frequency Coherence

Both naturally occurring and manmade signals exhibit defining time fluctuations in their concentrations of spectral energy. Such signals contain or may be designed to contain large amounts of information.⁴ Signals that are described concisely by the evolution of their spectral energy over time are necessarily coherent with respect to sets of atomic functions which are well localized in time and frequency, for example, families of wavelets. Such signals are said to be *time–frequency coherent*.

Examples of time–frequency coherent signals include radar, sonar, speech, and music signals. Specifically, frequency-modulated signals such as chirp signals, are, by definition, characterized by their time–frequency behavior. Effective methods (transformations) for extracting useful information from time–frequency coherent signals remain a focus of research in the signal processing community (e.g., [KFFJ92], [Dau90], [Hla92], [BO94], [CKBB93], [MF85], [Coh89], [KLB92], [BMG92], [NK93], [TB94a], [Sad96], and [MR97]). Many of these efforts can be cast in terms of coherence with respect to frames of wavelets.

In the remainder of this section, the coherence of a set of synthetic frequency-modulated signals with respect to several different representations is examined. First, consider the single linear frequency-modulated signal⁵ “ $e_{10}\text{chirp}@1$ ” having a chirp rate of $1\text{MHz}/\mu\text{s}$ and upshifted in frequency by 10MHz . Figure 7.2 shows the computed CSNRs for this chirp signal using four different representations:

- the signal (time) domain itself,
- the frequency domain,
- the discrete wavelet transform domain (using the Daub D_3 filter), and
- the OCWT domain (using a 32-bank Morlet Filter).

⁴Perhaps the most accessible example of this concept occurs in music. A musical score may be thought of as a method of documenting the gross time–frequency fluctuations in an audible signal. Of course, fine time–frequency fluctuations are imparted into the resulting music by the musician’s interpretation and the specific instrument used. In this sense, conventional western music scores are time–frequency representations of audio signals.

⁵See the MATLAB function `sampled_signal` discussed on page 271 for a complete description of synthetic signal construction.

Displayed in each of the four plots are the CDs for the noise and signal in each domain. Each plot contains a legend indicating the values of absolute coherence for both the signal and noise. The y -axis of each plot is labeled with the corresponding CSNR. Here, and throughout this book, the noise employed is additive white Gaussian noise (AWGN). Both the signal and noise have been energy normalized so that their L^2 norms are 1. Thus, the sum of the signal and noise would have an SNR of 1 (0dB). Note that for this signal, the largest CSNR is attained with the OCWT representation, the second largest with the FFT, the third largest with the DWT, and the smallest in original TIME domain.

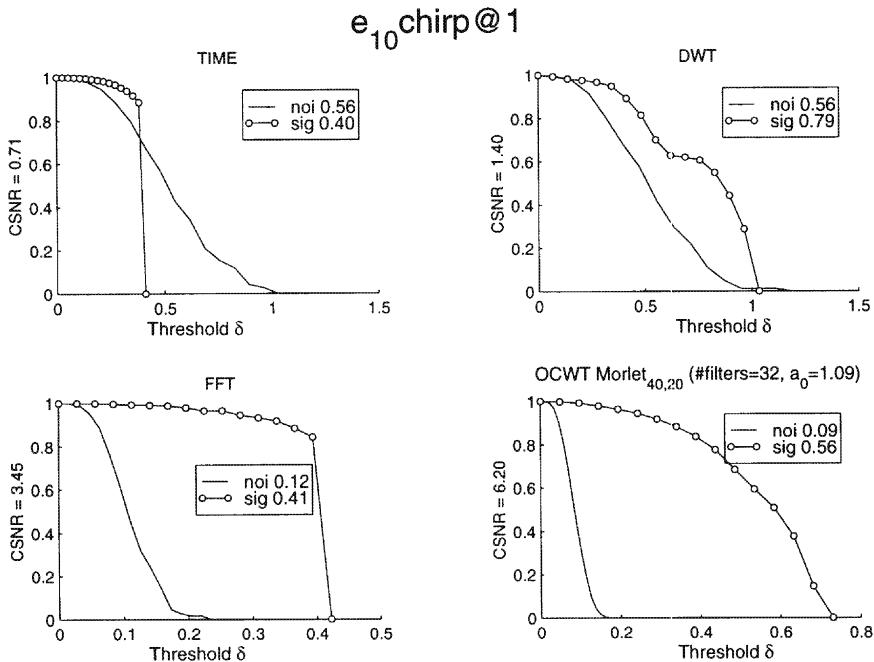


FIGURE 7.2. Absolute and relative coherence (CSNR) measures of normalized chirp and Gaussian noise signals with respect to four different representation domains.

Relative coherences with respect to the preceding four representations have been computed for 16 time-frequency coherent signals and reported in Table 7.1. These signals consist of linear modulated chirps with different rates and various combinations of sinusoidal packets. In addition, two different modulations (shift in frequency of 5 or 10MHz) have been applied to the various signals. Frequency upshifting by x is indicated by attaching the filename with the prefix “ e_x .” The rate of single component chirp signals

is indicated in the filename using the syntax “chirp@rate”; for example, $e_{10}\text{chirp}@2$ indicates a chirp signal with a rate of $2\text{MHz}/\mu\text{s}$ upshifted in frequency by 10MHz. The signal “chirp2” is a two-component chirp signal with rates of 1 and 3; the signals “packet” and “opacket” are nonoverlapping and overlapping, respectively, sequences of sinusoidal packets at fixed frequencies of 6, 11, and 20MHz.

This data set, though small, provides an opportunity to investigate the effects of modulation and chirp rate on coherence with respect to different representations. The coherency ordering seen for the chirp signal of Figure 7.2 is shown to be representative of the entire data set. In fact, Table 7.1 shows this order of relative coherency is maintained for every signal in the data set; that is,

$$CSNR_{\text{OCWT}} > CSNR_{\text{FFT}} > CSNR_{\text{DWT}} > CSNR_{\text{TIME}}.$$

This provides quantitative support to the statement that frequency-modulated signals are most highly coherent with respect to overcomplete wavelet frames. Relationships between coherence and modulation and/or chirp rate that are anecdotally supported by Table 7.1 are itemized as follows.

- The relative coherence values for a pure white Gaussian noise are signal nearly 1 for all four representations considered.
- Coherence in the time domain is insensitive to modulation and chirp rate.
- Coherence in the Fourier domain is insensitive to modulation.
- Slower chirps are more coherent with respect to the DWT, FFT, and OCWT representations.

7.1.5 Frame Localization and Coherence

From the theory of frames, any analog signal from an appropriate space is known to have an invertible representation in terms of a *countable* set of coefficients. Although it is present in the background of all computational signal processing techniques, frame representation is, nevertheless, mainly a *theoretical* device. Any practical implementation is necessarily limited to operations with *finite* representations, that is, representations that consist of a finite set of coefficients. To connect the two, an obvious approach is to select a subset of all possible coefficients. This approach leads naturally to the following questions.

1. How is a finite set of coefficients chosen?
2. What are bounds on the associated approximation error?

<i>Signal</i>	TIME	DWT	FFT	OCWT
noise	0.99	0.95	0.99	1.03
$e_5\text{chirp}2$	0.86	1.44	2.40	4.81
$e_{10}\text{chirp}2$	0.87	1.31	2.46	4.08
$e_5\text{chirp}@0.5$	0.71	1.93	4.73	8.01
$e_5\text{chirp}@1$	0.70	1.66	3.53	7.89
$e_5\text{chirp}@2$	0.71	1.41	2.43	6.50
$e_5\text{chirp}@4$	0.72	1.23	1.80	5.22
$e_{10}\text{chirp}@0.5$	0.74	1.54	4.81	7.22
$e_{10}\text{chirp}@1$	0.71	1.40	3.45	6.20
$e_{10}\text{chirp}@2$	0.70	1.23	2.39	5.81
$e_{10}\text{chirp}@4$	0.71	1.03	1.71	5.01
$e_{10}\text{packet}$	0.90	1.36	4.08	6.12
$e_5\text{opacket}$	1.35	1.88	4.01	7.02
$e_{10}\text{opacket}$	1.35	1.83	4.17	5.95
sine	1.80	3.93	6.25	14.05
$e_5\text{sine}$	1.76	3.20	6.08	13.35
$e_{10}\text{sine}$	1.78	2.89	6.10	12.55

TABLE 7.1. Computed CSNRs for different signals. Here the DWT representation is computed using the Daubechies D3 wavelet and the OCWT representation is computed with the Morlet(40,20) filter bank, ($a_0 = 1.09$, #filters = 32).

Here, approximation error refers to the error incurred by performing reconstruction based only on the finite set of coefficients.

To provide at least one answer to these questions, the threshold operator F_δ , discussed in Section 7.1.3, is a natural choice. Recall that F_δ takes a countably infinite representation to a finite representation by selecting only a finite subset of coefficients. In most cases of interest, the threshold operator F_δ is motivated by a specific signal $f_* = L^* c_* \in \mathcal{H}$ and an auxiliary threshold parameter δ . This idea is made precise in Definition 7.1 on page 174. In what follows, the process of applying a threshold operator on a given representation is often referred to as “truncation.”

In this section, localized frames resulting from truncation of a global frame for a Hilbert space \mathcal{H} are examined. The truncation process results in a finite-dimensional subspace of \mathcal{H} which itself is framed by finitely many elements taken from the global frame. Being a frame for the localized space, the finite-dimensional frame has an associated local frame representation.

Reconstruction from local frame representations are examined and bounds for the corresponding error are related to the coherence distribution.

As usual, assume that $\{\phi_n\}$ is a global frame for a Hilbert space $\mathcal{H} \subseteq L^2(\mathbb{R})$ with frame representation L and frame correlation R . Fix $\delta > 0$ and consider a particular element $f_* \in \mathcal{H}$ and the localization associated with the selection operator F_{δ,f_*} of Definition 7.1.

Since every choice of signal f_* and threshold δ may yield a different threshold operator, the set $\{F_{\delta,f_*}\}$ is actually an infinitely large family of operators. In turn, each threshold operator F_{δ,f_*} has associated with it a space $\mathcal{H}_\delta(f_*)$, which is *localized* around the signal f_* with respect to the frame $\{\phi_n\}$. Moreover, Proposition 7.5 shows that the localized space $\mathcal{H}_\delta(f_*)$ is finite-dimensional. Precisely,

$$\begin{aligned}\mathcal{H}_\delta(f_*) &\stackrel{\Delta}{=} \text{span} \{ \phi_n : |\langle f_*, \phi_n \rangle| > \delta \} \\ &= \text{span} \{ \phi_n : n \in J_\delta(f_*) \},\end{aligned}$$

where

$$J_\delta(f_*) = \{n : |\langle f_*, \phi_n \rangle| > \delta\}$$

and $\text{card } J_\delta(f_*) < \infty$. In fact, Proposition 7.5 shows that $\text{card } J_\delta(f_*) < C \cdot \text{Coh}_\delta f_*$ for some $C < \infty$.

For all $c \in L(\mathcal{H})$ and $\delta > 0$, the threshold operator F_{δ,f_*} provides an orthogonal decomposition of $L(\mathcal{H})$ as

$$c = F_{\delta,f_*} c + (I - F_{\delta,f_*})c$$

and

$$\|c\|^2 = \|F_{\delta,f_*} c\|^2 + \|(I - F_{\delta,f_*})c\|^2.$$

Such a threshold operator F_{δ,f_*} partitions c into two segments: one for which c_* has elements larger than δ and one for which c_* has elements less than or equal to δ . The two following propositions show that

- the former segment resides in a finite-dimensional space (whose dimension is proportional to δ^{-2}), and
- it is always possible to find a $\delta > 0$ which ensures that an arbitrarily small positive percentage of the energy from the whole sequence will be contained in this first finite-dimensional segment.

These are consequences of the fact that the representation is of finite energy, that is, a member of $L(\mathcal{H})$.

Proposition 7.5 *Let L be a frame representation with associated upper frame bound B . Suppose that F_{δ,f_*} is a threshold operator (Definition 7.1) for $f_* \in \mathcal{H}$ and $\delta > 0$. For all $\delta > 0$*

$$\dim \{F_{\delta,f_*} L(\mathcal{H})\} \leq \frac{B \cdot \text{Coh}_\delta f_*}{\delta^2} \|f_*\|^2.$$

Proof: Note that

$$\dim \{F_{\delta, f_*} L(\mathcal{H})\} = \dim \{\mathcal{H}_\delta(f_*)\} = \text{card } J_\delta(f_*).$$

Write

$$\|F_{\delta, f_*} Lf_*\|^2 = \sum_{|(Lf_*)_n| \geq \delta} |(Lf_*)_n|^2 \geq \delta^2 \text{card } J_\delta(f_*).$$

Thus,

$$\begin{aligned} \text{card } J_\delta(f_*) &\leq \delta^{-2} \cdot \|F_{\delta, f_*} Lf_*\|^2 \\ &= \delta^{-2} \cdot (\text{Coh}_\delta f_*) \|Lf_*\|^2 \\ &\leq \frac{B \cdot \text{Coh}_\delta f_*}{\delta^2} \|f_*\|^2 \end{aligned}$$

from the definition of coherence. ■

Proposition 7.6 Suppose F_δ is a threshold operator (Definition 7.1). Given $\epsilon > 0$ there is a $\delta > 0$ so that

$$\forall c \in \ell^2(\mathbb{Z}), \quad \|(I - F_\delta)c\|^2 < \epsilon \|c\|^2. \quad (7.3)$$

Proof: Clearly,

$$\lim_{\delta \rightarrow 0} \|F_\delta c\|^2 = \|c\|^2 < \infty.$$

Therefore, for $\epsilon > 0$ arbitrary there is some $\delta > 0$ so that

$$|\|c\|^2 - \|F_\delta c\|^2| < \epsilon \|c\|^2.$$

Since $\|c\|^2 = \|F_\delta c\|^2 + \|(I - F_\delta)c\|^2$ it may be concluded that

$$\|(I - F_\delta)c\|^2 < \epsilon \|c\|^2.$$
■

Equation (7.3) expresses the notion that the operator $F_{\delta, f_*} L$ extracts the most significant frame coefficients with respect to the specific signal $f_* \in \mathcal{H}$. Here the term “most significant” is quantified by the parameter $\epsilon \in (0, 1)$. For example, a value of $\epsilon \approx 0$ indicates that almost every coefficient is significant, and a value of $\epsilon \approx 1$ indicates that almost every coefficient is insignificant. Even more, for a given ϵ and function $f \in \mathcal{H}$, a δ satisfying (7.3) is given directly by the CD of f , that is, any δ that satisfies

$$\text{Coh}_\delta f > 1 - \epsilon.$$

Because the CD is monotonic decreasing there is a range of acceptable threshold values $\delta \in [0, \delta_M]$ and a maximum (best) threshold δ_M . This is stated and shown formally as a corollary to the proposition.

Corollary 7.7 Given $f_* \in \mathcal{H}$ and $\epsilon > 0$ there is a $\delta > 0$ such that

$$\text{Coh}_\delta f_* > 1 - \epsilon.$$

Proof: Rewrite Equation (7.3) as

$$\epsilon > \frac{\|(I - F_{\delta, f_*})L f_*\|}{\|L f_*\|^2} = 1 - \text{Coh}_\delta f_*.$$

■

Theorem 7.8 relates the error incurred due to reconstructing a signal from its thresholded representation to its coherence with respect to the atoms underlying the representation. In particular, it shows that as the coherence of a signal approaches 1, the reconstruction error approaches 0. More than this, it provides a precise statement of the notion that a coherent signal can be well represented by the most important (e.g., largest) coefficients in its frame expansion and implies a natural decomposition of the space \mathcal{H} as

$$\mathcal{H}_\delta(f_*) \oplus \mathcal{H}_\delta(f_*)^\perp.$$

Theorem 7.8 Given a signal $f_* \in \mathcal{H}$, suppose $\{\phi_n\}$ is a frame for \mathcal{H} with representation operator L , frame correlation R , and frame bounds A and B . For all $\delta \geq 0$

$$\frac{\|f_* - f_\delta\|^2}{\|f_*\|^2} \leq \frac{B}{A} \cdot [1 - \text{Coh}_{\delta, \{\phi_n\}} f_*],$$

where

$$f_\delta \triangleq L^* R^\dagger F_{\delta, f_*} L f_*.$$

Proof: First, note that from Equation (3.15)

$$L f_\delta = LL^* R^\dagger F_{\delta, f_*} L f_* = P_{L(\mathcal{H})} F_{\delta, f_*} L f_*.$$

Because L is a frame representation

$$\forall g \in \mathcal{H}, \quad A\|g\|^2 \leq \|Lg\|^2 \leq B\|g\|^2.$$

In particular, letting $g = f_* - f_\delta$ gives

$$\begin{aligned} A\|f_* - f_\delta\|^2 &\leq \|L(f_* - f_\delta)\|^2 \\ &= \|Lf_* - Lf_\delta\|^2 \\ &= \|Lf_* - P_{L(\mathcal{H})} F_{\delta, f_*} L f_*\|^2 \\ &= \|P_{L(\mathcal{H})} (I - F_{\delta, f_*}) L f_*\|^2 \\ &\leq \|P_{L(\mathcal{H})}\| \cdot \|(I - F_{\delta, f_*}) L f_*\|^2 \\ &= \|(I - F_{\delta, f_*}) L f_*\|^2 \\ &= [1 - \text{Coh}_{\delta, \{\phi_n\}} f_*] \cdot \|L f_*\|^2 \\ &\leq [1 - \text{Coh}_{\delta, \{\phi_n\}} f_*] \cdot B\|f_*\|^2, \end{aligned}$$

from which the result follows. The manipulations are justified, respectively, as frame definition applied to $f_* - f_\delta$, linearity of L , substitution for Lf_δ calculated previously, substitution of $Lf_* = P_{L(\mathcal{H})}Lf_*$ and distributive properties, property of norms, operator norm inequality, $\|P_{L(\mathcal{H})}\| = 1$ ($P_{L(\mathcal{H})}$ is the orthogonal projection onto the range of L), definition of coherence (Definition 7.2), and finally, upper frame bound applied to f_* . ■

Note that the bound on the relative reconstruction error given in Theorem 7.8 is not sharp for general frames in the sense that the actual relative error may be much smaller than the upper bound. Certain items affect the sharpness of this bound:

1. the tightness of the frame $\{\phi_n\}$ (if the frame were tight ($A = B$) then the first and last inequalities in the proof would become equalities); and
2. the exactness of the frame $\{\phi_n\}$ (if the frame were exact then $P_L(\mathcal{H})$ would be the identity operator since $L(\mathcal{H}) = \ell^2(\mathbb{Z})$ by Theorem 3.18(a), and the middle inequality would become an equality).

From these observations, it may be concluded that, in the case of tight and exact frames $\{\phi_n\}$,

$$\frac{\|f_* - f_\delta\|^2}{\|f_*\|^2} = 1 - \text{Coh}_{\delta, \{\phi_n\}} f_*.$$

In particular, this is true for all orthonormal bases and, hence, is true for the DWT. Moreover, nonexactness (overcompleteness) leads, in general, to smaller relative errors. This is because the smaller $L(\mathcal{H})$ is, the smaller $\|P_{L(\mathcal{H})}(I - F_{\delta, f_*})Lf_*\|$ may be. Corollary 7.9 formally states these observations and provides another possible benefit of overcompleteness over orthonormal representations.

Corollary 7.9 *If, in addition to the assumptions of Theorem 7.8, the frame $\{\phi_n\}$ for \mathcal{H} is tight and exact then*

$$\frac{\|f_* - f_\delta\|^2}{\|f_*\|^2} = 1 - \text{Coh}_{\delta, \{\phi_n\}} f_*.$$

Proof: Exercise.

Together, Corollary 7.9, Theorem 7.8, and the fact that

$$\text{CSNR}_{\{\phi_n\}} > \text{CSNR}_{\{u_n\}},$$

where $\{\phi_n\}$ is overcomplete and $\{u_n\}$ is orthonormal support the notion that redundant representations are better for noise suppression.

7.1.6 Reconstruction Error from Thresholded OCWTs

In this section, a nonrigorous argument is given as to why the reconstruction error bounds of Theorem 7.8 apply to the OCWT with $A = B$ even though the underlying frame is not tight. With this knowledge, Theorem 7.8 can be used to give bounds on the performance of the wavelet noise suppression scheme outlined in Section 7.1.3.

The idea of the argument is that the combination of OCWT frame representation and thresholding with a fixed threshold is equivalent to thresholding a tight frame with a scale-dependent threshold; and, furthermore that this variable threshold reconstruction has an associated error that is bounded above by reconstruction with respect to a fixed threshold.

Attention is limited to OCWT representations with respect to the underlying frame

$$\Psi_{\text{OCWT}} \triangleq \{\tau_{n\Delta} D_{a_0^m} g\}_{n,m}$$

for $PW_{[a,b]}$, where $b > a > 0$, $a_0 > 1$, $\Delta^{-1} > 2(b-a)$, and g is an analyzing wavelet. A direct application of Theorem 7.8 will not lead to a useful bound since the frame bounds A and B of Ψ_{OCWT} are necessarily spread far apart. Fortunately, the OCWT analysis, thresholding, and synthesis may be equivalently implemented with respect to the weighted frame

$$\Psi'_{\text{OCWT}} \triangleq \left\{ a_0^{m/2} \tau_{n\Delta} D_{a_0^m} g \right\}_{n,m}.$$

This weighted frame is tight (Exercise 6.11).

Figures 7.3 and 7.4 depict the equivalent implementations for a simple two channel filter bank. This corresponds to limiting attention to the two scale indices $m = 0, 1$. In Figure 7.3 the representation W'_g with respect to Ψ' is thresholded with a fixed threshold δ and then reconstructed based on the thresholded version. In this case, Theorem 7.8 is directly applicable where $L = W'_g$.

Figure 7.4 is an equivalent view of the processing accomplished in Figure 7.3 with the channel modifications:

1. channel m in the analysis bank is multiplicatively scaled by a value $a^m > 1$;
2. the channel m threshold is likewise multiplicatively scaled by a^m ; and
3. the thresholded output on channel m is multiplicatively scaled by a^{-m} ,

for $m = 0, 1$. Stated in this way, these modifications may be easily extended to filter banks with more than two filters simply by expanding the range of m . Clearly, the overall processing is unchanged by these modifications with the exception that $W_g f$ is made explicitly available at the output of

the analysis bank. Thus, the modified filter bank may be interpreted as a *variable* (channel-dependent) thresholding of the representation $W_g f$.

In the OCWT implementation of the thresholding scheme, however, a fixed threshold is applied to the representation W_g with respect to the unweighted frame Ψ . It turns out that simply replacing the variable thresholding in Figure 7.4 with a fixed thresholding by δ still allows the conclusion of Theorem 7.8 to hold. To see this consider the difference between thresholding $W_g f$ with the variable threshold and the fixed threshold. Since $a > 1$, replacing $a\delta$ with δ effectively lowers the threshold for channel two and results in a thresholded representation with more nonzero coefficients. Thus, reconstruction based on more nonzero coefficients must lead to better approximations to f . The same argument may be extended to OCWT filter banks with an arbitrary number of channels.

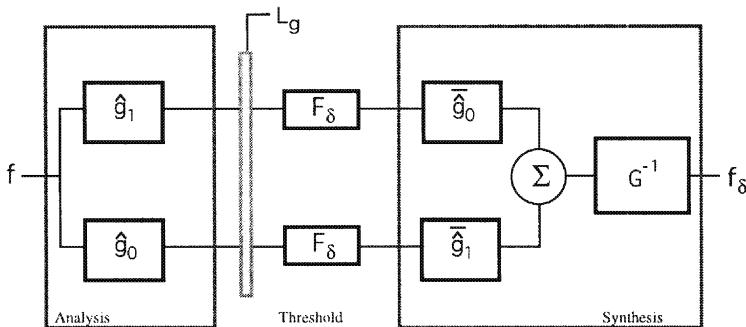


FIGURE 7.3. Filter bank implementation schematic for a fixed threshold reconstruction from a tight frame representation.

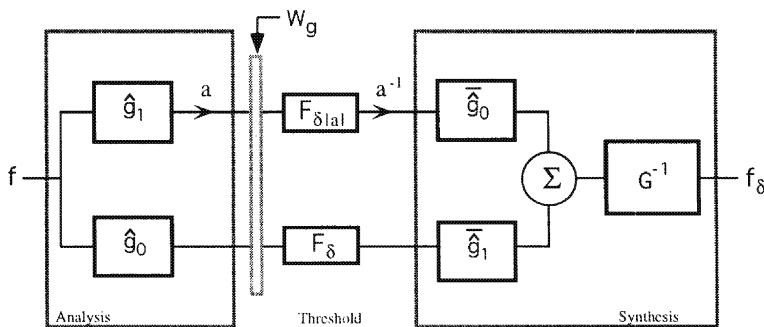


FIGURE 7.4. Equivalent filter bank implementation of a fixed threshold reconstruction which acts as a variable threshold on a weighted tight frame.

7.1.7 Numerical Experiment

To empirically investigate the noise suppression ability of the wavelet thresholding scheme proposed in Section 7.1.3, Experiment 7.10 details a general numerical simulation. Signals used in the experiment should come from a set \mathcal{F} having frequency support inside the range of frequencies covered by the filter bank. This is to ensure that no signal information is lost in the analysis filter bank.

Numerical Experiment 7.10 (Noise Suppression)

For a fixed OCWT⁶ W_g and each desired level of SNR do the following:

1. Synthesize a set of time-frequency coherent signals $\mathcal{F} = \{f_{*j}\}$.
2. For each signal in \mathcal{F} repeat the following for N_{trials} number of trials.
 - (a) Corrupt the signal. Add a scaled by σ Gaussian distributed random signal w to yield the instance

$$f_j = f_{*j} + \sigma \cdot w.$$

The value of σ is determined to yield the desired SNR level α (viz. Section 7.1.9).

- (b) Compute the OCWT $W_g f_j$.
- (c) For each threshold of interest in $\{\delta_k\}$ do the following:
 - i. Compute the thresholded reconstruction r_k

$$r_{j,k} = W_g^{-1} F_{\delta_k} W_g f_j.$$

- ii. Compute the relative error

$$ERR_{j,k} = \frac{\|f_j - r_{j,k}\|}{\|f_j\|}.$$

Because of the large number of free parameters in the simulation, only a limited number of instances are illustrated in this section. In particular, only a single OCWT filter bank is considered throughout. This OCWT is implemented with a Morlet filter bank having the transfer functions shown in Figure 7.5. In addition, the set of signals \mathcal{F} consists of only three signals taken from Table 7.1. Namely, they are

1. ($e_{10}\text{chirp}@1$) an upshifted by 10MHz linear FM chirp at a rate of 1MHz/ μ s;

⁶Practically, an OCWT is specified by the choice of the following filter bank parameters: dilation constant a_0 , number of filters n_f , and an analyzing wavelet g .

2. (e_5 chirp@4) an upshifted by 5MHz linear FM chirp at a rate of 4MHz/ μ s; and,
3. (e_{10} packet) an upshifted by 10MHz train of 3 sinusoidal bursts (packets) with frequencies at 6, 11, and 20MHz.

Detailed results of the noise suppression experiment are presented in figure and table form in the remainder of this section. Specific parameters used in the experiment are given in Table 7.2.

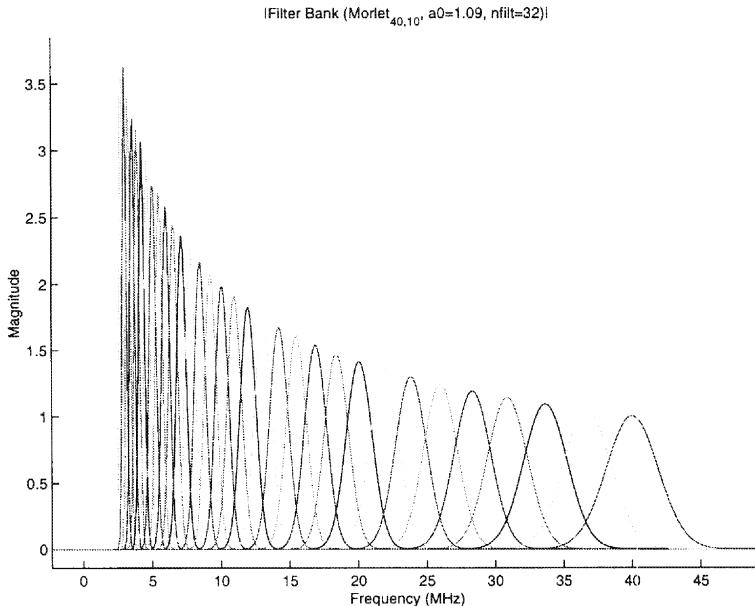


FIGURE 7.5. Morlet Filter bank with 32 channels used in Numerical Experiment 7.10.

For the special case of no noise ($\text{SNR} = \infty$) the relative error versus relative threshold δ as computed in the experiment is plotted for each of the signals in \mathcal{F} in Figures 7.6, 7.8, and 7.10. The relative threshold δ_{rel} is related to the absolute threshold δ via the coefficient maximum magnitude; that is,

$$\delta = \delta_{\text{rel}} \cdot \|L f_*\|_\infty.$$

Also plotted in these figures are the upper bounds computed from the coherences as

$$(1 - \text{Coh}_\delta f_*)^{1/2}$$

as predicted by Theorem 7.8. Note that there is a significant gap in the upper bound given by the theorem and the actual measured error value. The two facts (from Section 7.1.6) that

Parameter	Symbol	Values
SNR (dB)	α	-6, -3, 0, 2.5, 5, 7.5, 10, 20, 30, ∞
Thresh	δ	0.01, 0.1, 0.2, 0.3, 0.4, 0.5
# Trials	N_{trials}	16
Dilation Const.	a_0	1.09
# Filters	n_f	32
Wavelet	g	g_{Morlet}
Sample Period	Δ	1/128
Interval	T	4

TABLE 7.2. Parameters for Numerical Experiment 7.10.

1. the bound is sharp for orthonormal bases with fixed thresholding, and
2. the effective variable (channel-dependent) thresholding performed leads to smaller reconstruction error,

suggest that this gap is a performance benefit gained from using the OCWT representation. Tables 7.3 through 7.5 give the same data as the preceding figures in tabular form. In addition, the absolute threshold values for each relative threshold are listed in the tables as well.

Companion Figures 7.7, 7.9, and 7.11 show the time domain reconstructions for the three different (relative to the maximum coefficient magnitude) threshold values of 0.1, 0.2, and 0.3. The y -axes of these graphs are labeled with the corresponding error incurred and may be verified using the values given in the error versus threshold graphs described previously.

In Figures 7.12, 7.16, and 7.20, wavelet transform magnitudes of a train of three concatenated replicas of each signal are displayed. The first replica in the train has no noise added whereas the second and third replicas have SNRs of 10dB and 5dB, respectively.

Plots showing the reconstruction error versus SNR for the three different relative thresholds of 0.1, 0.2, and 0.3 are displayed in Figures 7.13, 7.17, and 7.21. Note that the asymptotes (corresponding to the SNR approaching ∞) of these curves are bounded above by the theoretical values given by Theorem 7.8 and equal to the measured values shown in Figures 7.6, 7.8, and 7.10. For a fixed threshold δ , the error curve is seen to be flat for large values of SNR and then becomes nearly vertical as the SNR becomes smaller. The point at which the error as a function of SNR becomes vertical indicates the SNR level at which noise suppression using this threshold breaks down. For a larger threshold, this breakpoint moves toward lower SNRs, but at the cost of a higher relative error.

Finally, for the two SNR cases of 5dB and 10dB time domain reconstruction comparisons are presented for each signal. Figures 7.14, 7.18, and 7.22 show the SNR = 10dB cases and Figures 7.15, 7.19, and 7.23 show the SNR = 5dB cases. Each of these figures contains these separate plots:

1. **(Upper Left)** noisy observation $f + w$;
2. **(Upper Right)** original uncorrupted version f ;
3. **(Lower Left)** thresholded reconstruction f_δ ,
 - (a) $\delta_{\text{rel}} = 0.1$ for the SNR = 10dB case, and
 - (b) $\delta_{\text{rel}} = 0.2$ for the SNR = 5dB case; and
4. **(Lower Right)** the relative error signal $(f - f_\delta)/\|f\|$.

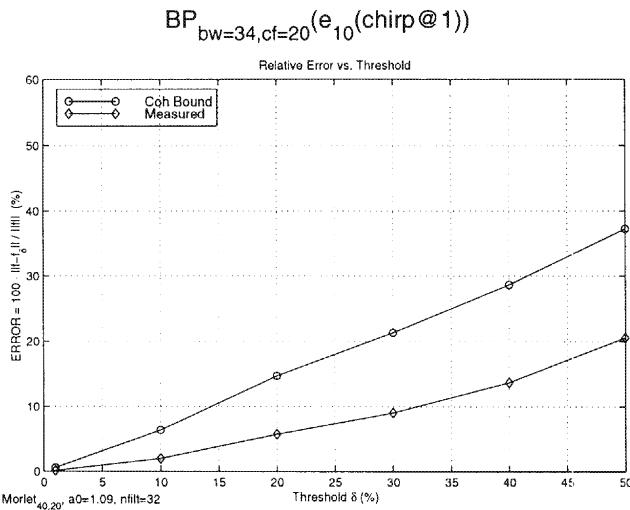


FIGURE 7.6. Coherence bound and measured values as function of threshold for the signal $e_{10}\text{chirp}@1$.

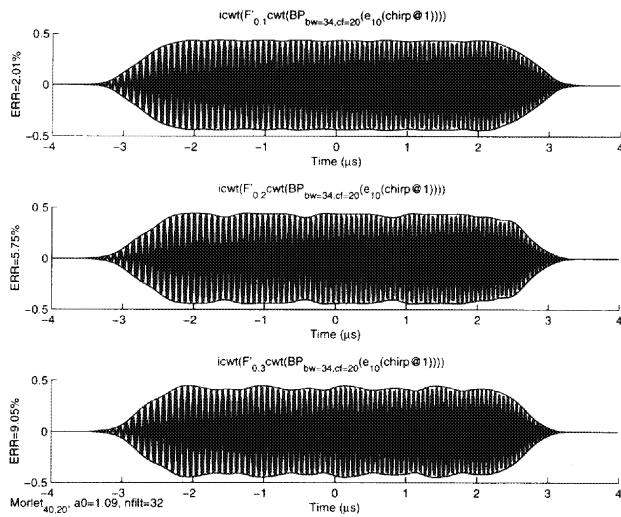


FIGURE 7.7. Reconstructions associated with relative threshold values of 0.1, 0.2, and 0.3 for the signal $e_{10}\text{chirp}@1$.

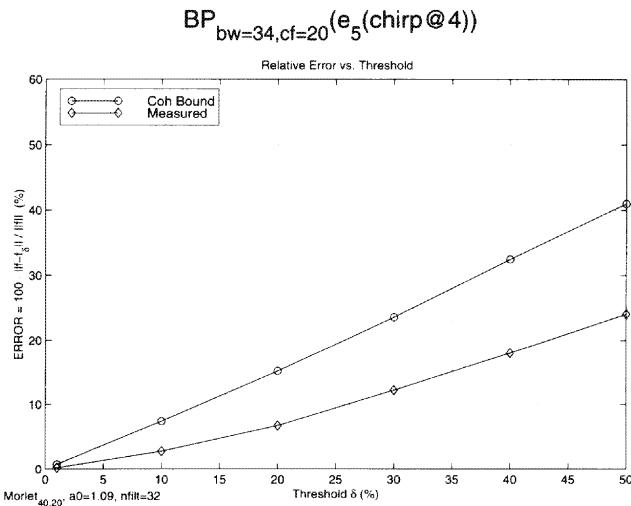


FIGURE 7.8. Coherence bound and measured values as function of threshold for the signal $e_{10}\text{chirp}@4$.

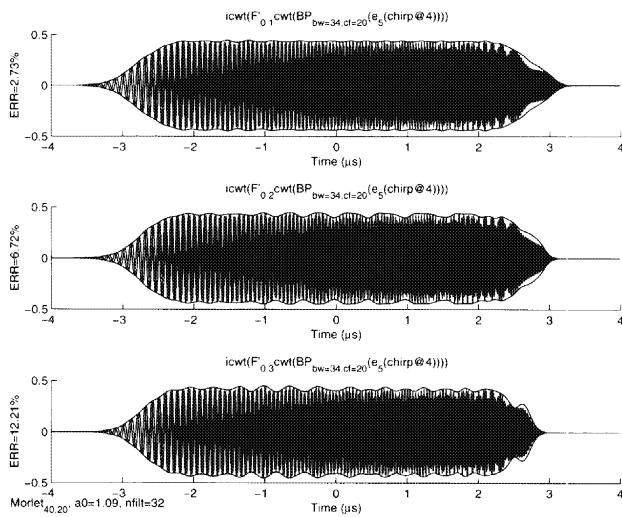


FIGURE 7.9. Reconstructions associated with relative threshold values of 0.1, 0.2, and 0.3 for the signal $e_5\text{chirp}@4$.

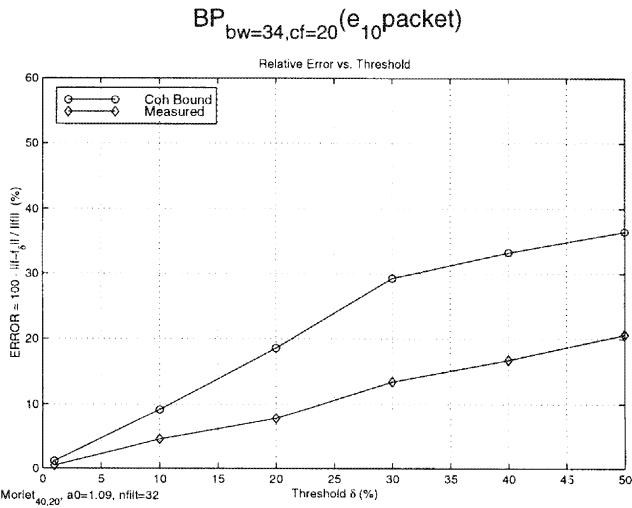


FIGURE 7.10. Coherence bound and measured values as function of threshold for the signal $e_{10}\text{packet}$.

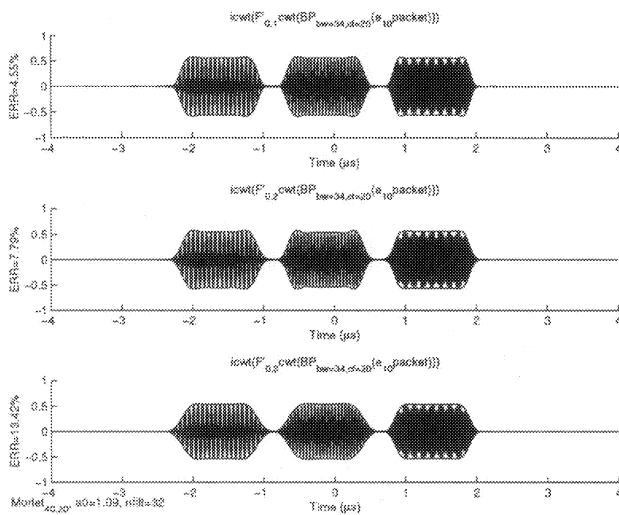


FIGURE 7.11. Reconstructions associated with relative threshold values of 0.1, 0.2, and 0.3 for the signal $e_{10}\text{packet}$.

δ_{rel}	δ	Coh_δ	$\sqrt{1 - Coh_\delta}$	Err
0.010	0.008	1.000	0.006	0.001
0.100	0.078	0.996	0.064	0.020
0.200	0.156	0.978	0.147	0.058
0.300	0.234	0.955	0.213	0.090
0.400	0.312	0.918	0.287	0.137
0.500	0.390	0.861	0.373	0.206

TABLE 7.3. Coherence, error bound, measured error, and absolute and relative thresholds for the signal $e_{10}\text{chirp}@1$.

δ_{rel}	δ	Coh_δ	$\sqrt{1 - Coh_\delta}$	Err
0.010	0.007	1.000	0.007	0.002
0.100	0.067	0.995	0.074	0.027
0.200	0.134	0.977	0.152	0.067
0.300	0.201	0.944	0.236	0.122
0.400	0.268	0.895	0.325	0.180
0.500	0.335	0.832	0.410	0.241

TABLE 7.4. Coherence, error bound, measured error, and absolute and relative thresholds for the signal $e_5\text{chirp}@4$.

δ_{rel}	δ	Coh_δ	$\sqrt{1 - Coh_\delta}$	Err
0.010	0.008	1.000	0.011	0.004
0.100	0.081	0.992	0.092	0.046
0.200	0.161	0.966	0.185	0.078
0.300	0.242	0.914	0.293	0.134
0.400	0.323	0.889	0.333	0.167
0.500	0.404	0.867	0.365	0.206

TABLE 7.5. Coherence, error bound, measured error, and absolute and relative thresholds for the signal $e_{10}\text{packet}$.

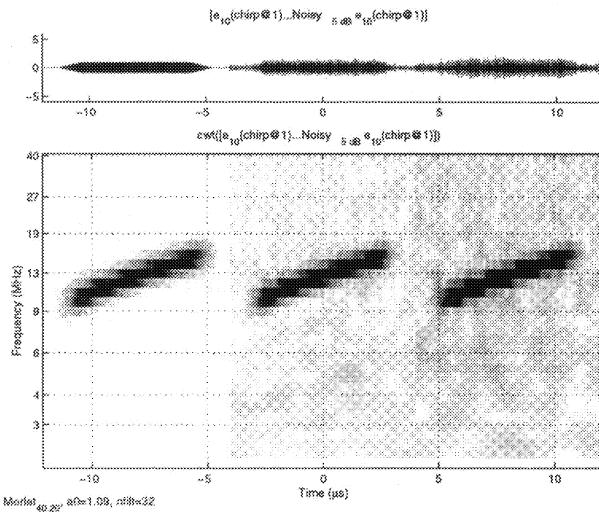


FIGURE 7.12. Wavelet transform magnitude for concatenation of three versions of $e_{10}\text{chirp}@1$ with SNRs of ∞ , 10, and 5dB.

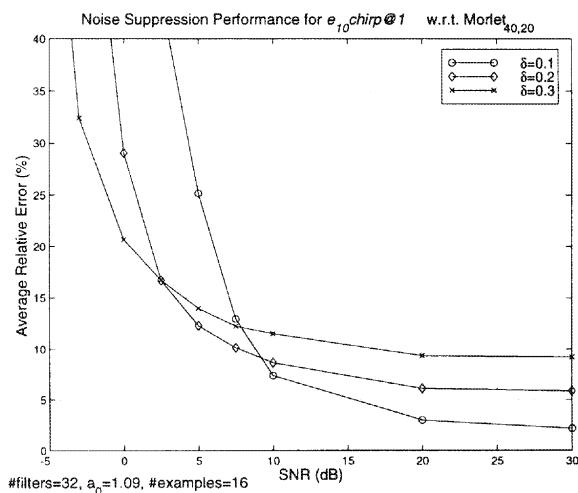


FIGURE 7.13. Measured (16 trial avg.) relative error as a function of SNR for signal $e_{10}\text{chirp}@1$ using relative threshold values of 0.1, 0.2, and 0.3.

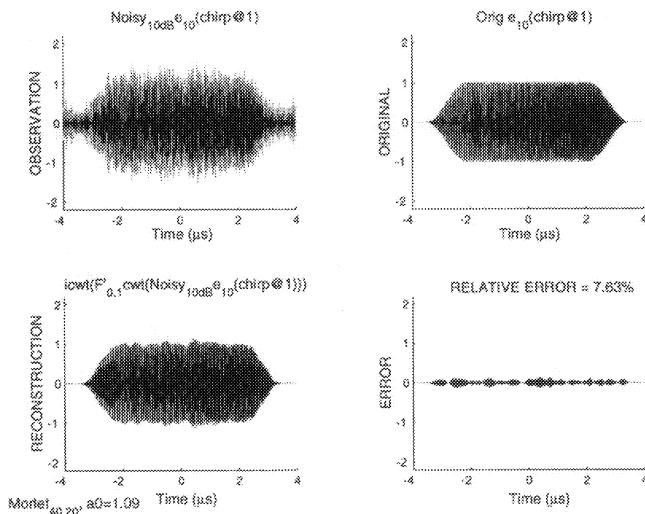


FIGURE 7.14. Time reconstruction of $e_{10}\text{chirp@1}$ from its SNR = 10dB noisy observation using a relative threshold of 0.1.

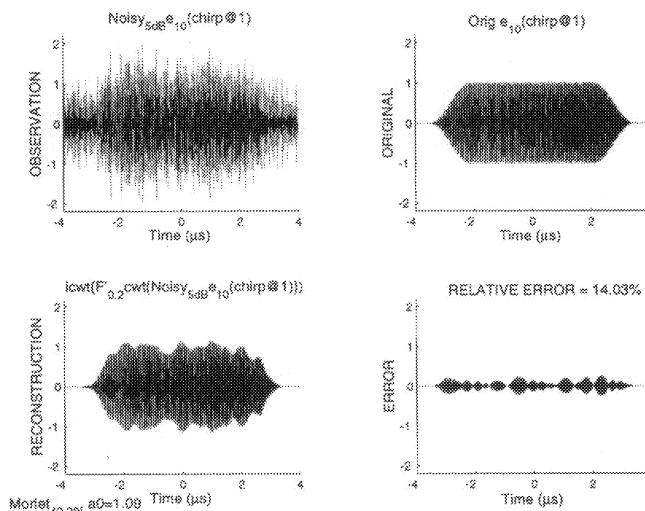


FIGURE 7.15. Time reconstruction of $e_{10}\text{chirp@1}$ from its SNR = 5dB noisy observation using a relative threshold of 0.2.

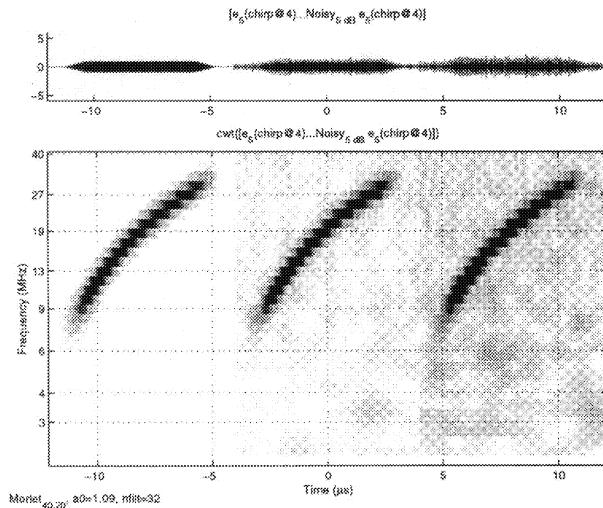


FIGURE 7.16. Wavelet transform magnitude for concatenation of three versions of $e_5\text{chirp}@4$ with SNRs of ∞ , 10, and 5dB.

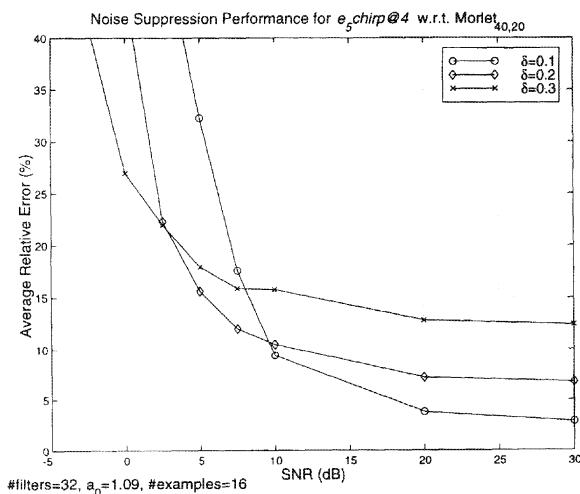


FIGURE 7.17. Measured (16 trial avg.) relative error as a function of SNR for signal $e_5\text{chirp}@4$ using relative threshold values of 0.1, 0.2, and 0.3.

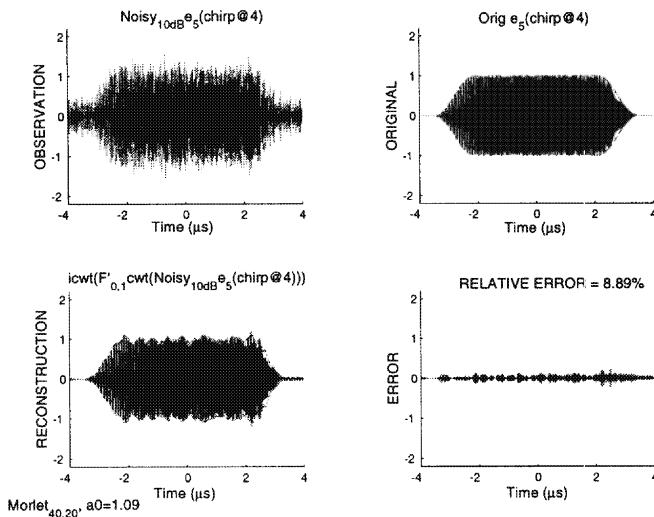


FIGURE 7.18. Time reconstruction of e_5 chirp@4 from its SNR = 10dB noisy observation using a relative threshold of 0.1.

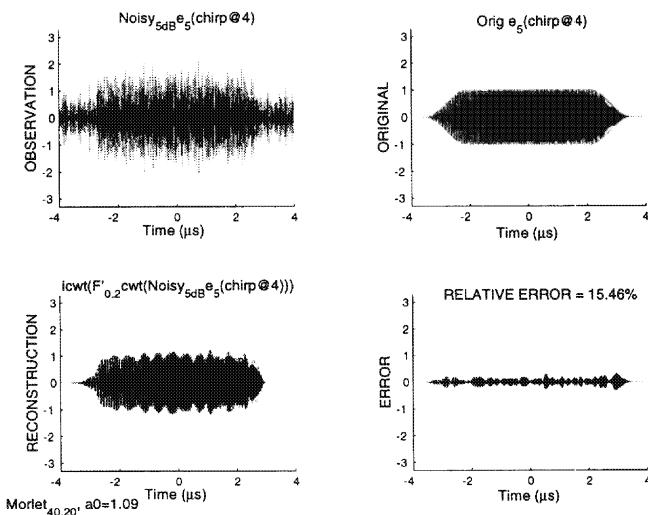


FIGURE 7.19. Time reconstruction of e_5 chirp@4 from its SNR = 5dB noisy observation using a relative threshold of 0.2.

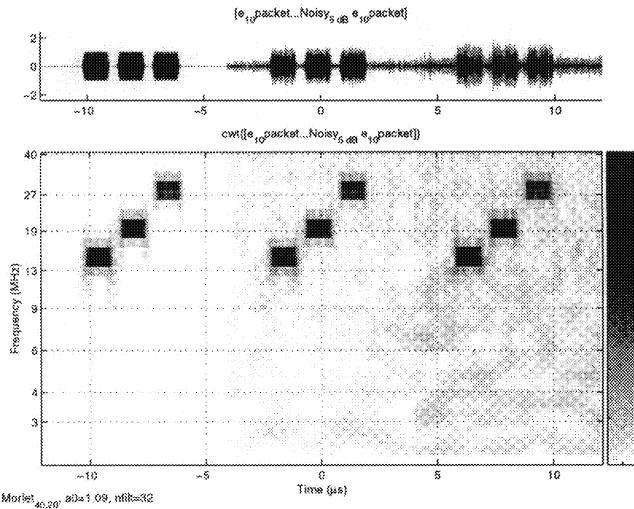


FIGURE 7.20. Wavelet transform magnitude for concatenation of three versions of e_{10} packet with SNRs of ∞ , 10, and 5dB.

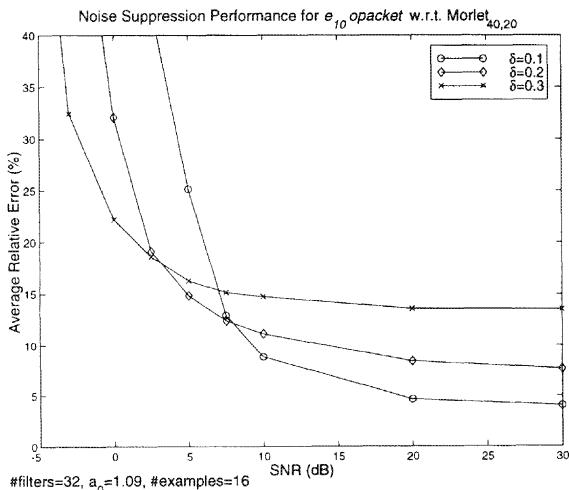


FIGURE 7.21. Measured (16 trial avg.) relative error as a function of SNR for signal e_{10} packet using relative threshold values of 0.1, 0.2, and 0.3.

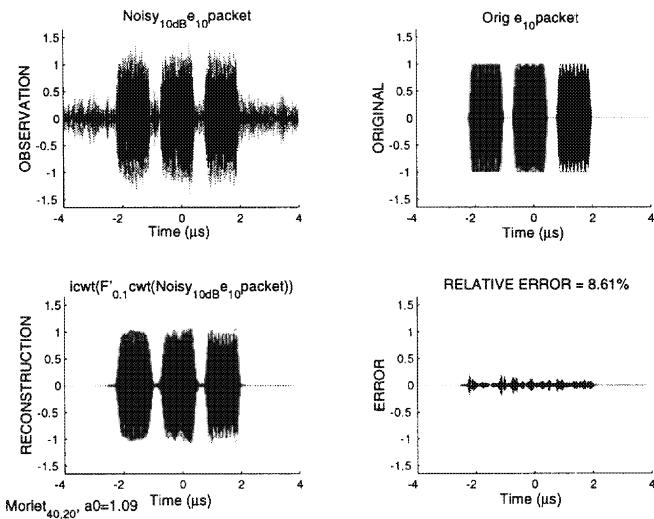


FIGURE 7.22. Time reconstruction of e_{10} packet from its SNR = 10dB noisy observation using a relative threshold of 0.1.

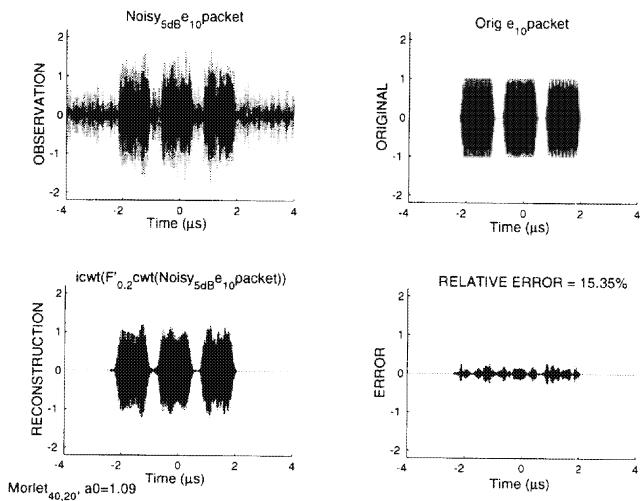


FIGURE 7.23. Time reconstruction of e_{10} packet from its SNR = 5dB noisy observation using a relative threshold of 0.2.

Although very limited in scope, the noise suppression experiment supports the following conclusions and observations.

- Redundant representations may offer greater performance benefits over orthonormal representations as evidenced by measured performance versus predicted bounds via Theorem 7.8.
- Slower chirps exhibit better noise suppression qualities than faster chirps. This observation is consistent with the fact that slower chirps are more coherent with respect to the wavelet frame of the experiment (c.f. Table 7.1).
- Large relative error as measured by the energy (L^2) difference may not be too bad. This is because such an error does not account for the fact that a well-localized wavelet representation will lead to errors that live in the same time–frequency region as the original signal. In other words, if the original signal is concentrated on the interval $[0, 1]$ the errors incurred by a wavelet-based thresholding will be concentrated on that interval as well. Similar remarks hold for the frequency domain. Thus, larger relative errors may be more tolerable than would be thought without this consideration.

7.1.8 Coefficient Noise and Overcompleteness

In this section, issues concerning noise in the coefficient domain are addressed. As always, suppose $f \in \mathcal{H}$ is a signal of interest with frame representation $Lf \in L(\mathcal{H})$.

There are various sources of perturbation that may occur in the coefficient domain. For instance, the representation Lf may be communicated, stored, or retrieved incorrectly. Moreover, any processing algorithms, such as thresholding, that operate in the coefficient domain will by design intentionally create a perturbation. For overcomplete signal representations there is no reason that general coefficient domain processing should result in a processed coefficient sequence that remains in the range $L(\mathcal{H})$. An additional practical constraint on the real- (or complex-) valued coefficients Lf is that they must necessarily be described by a finite number of bits; that is, they must be quantized. Quantization is not explicitly addressed, but is instead viewed as an additional source of perturbation in the coefficient domain.

For these reasons it is important to investigate the robustness of frame representations with respect to coefficient noise. In light of the preceding, it is clear that frame representations would be of little practical use if small perturbations in the coefficient domain destroyed large amounts of information in the signal domain. The following argument (cf. [DGM86] and [Dau92, Section 3.6]) asserts that frames which are far from exact,

that is, have a large degree of linear dependence, exhibit robustness to noise in the coefficient domain.

Suppose \tilde{c} is a noise-corrupted version of the coefficients Lf of a signal f . For an additive coefficient disturbance

$$\tilde{c} = Lf + d,$$

where d is random noise that has a known distribution. Recall the definitions of the frame correlation R (Definition 3.15) and pseudo-inverse R^\dagger . Motivated by the theory of frames, a natural reconstruction approximant of f from \tilde{c} is $f^\dagger \triangleq L^* R^\dagger \tilde{c}$ which gives

$$L^* R^\dagger \tilde{c} = L^* R^{-1} P_{L(\mathcal{H})}(Lf + d_{\parallel} + d_{\perp}) = L^* R^{-1}(Lf + d_{\parallel}),$$

where $d = d_{\parallel} + d_{\perp}$ is the orthogonal decomposition of d with respect to L ; that is, $d_{\parallel} \in L(\mathcal{H})$ and $d_{\perp} \in L(\mathcal{H})^\perp$. Thus, all the noise energy outside the range $L(\mathcal{H})$ is automatically nullified in the reconstruction process. For a uniformly distributed perturbation d , the larger the kernel of L^* the more energy in the noise will be nullified and the more noise tolerance will be achieved. Since the kernel of L^* being large is equivalent to the underlying frame having a large degree of linear dependence we may conclude that nonexactness yields robustness to coefficient imprecision. Note that \tilde{c} need not be in the range $L(\mathcal{H})$.

In Section 3.4.7 an iterative algorithm, Algorithm 3.24, was presented for the reconstruction of a signal from its *uncorrupted* frame representation. However, as was shown earlier, this algorithm will not converge for arbitrary initial data (i.e., for discrete representations in $\ell^2(\mathbb{Z})$ that are outside the range $L(\mathcal{H})$). Here, a second algorithm is provided that will converge on all of $\ell^2(\mathbb{Z})$. This algorithm is given as Algorithm 7.13.

The aim is to modify Algorithm 3.24 so as to alleviate the problem of initialization with a coefficient sequence outside the range of L . As a first step note that $Rc \in L(\mathcal{H})$ for all $c \in \ell^2(\mathbb{Z})$. Thus, if c_0 were initialized to $R\tilde{c}$ then all that would be required would be to provide an algorithm for the computation of R^{-2} on $L(\mathcal{H})$ instead of R^{-1} on $L(\mathcal{H})$. Before such an algorithm may be presented, it is necessary first to attend to some technical details.

Lemma 7.11 *If $A \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$ then*

$$\|I - A^* A\| < 1$$

if and only if

$$\forall x \in \mathcal{H}_1, \quad 0 < \inf_{\|x\|=1} \|Ax\| \leq \sup_{\|x\|=1} \|Ax\| = \|A\| < \sqrt{2},$$

where I is the identity operator on \mathcal{H}_1 .

Proof: Since $I - A^*A$ is self-adjoint

$$\begin{aligned}\|I - A^*A\| &= \sup_{\|x\|=1} |\langle x, (I - A^*A)x \rangle| \\ &= \sup_{\|x\|=1} |1 - \|Ax\|^2|.\end{aligned}$$

Therefore the condition that $\|I - A^*A\| < 1$ is equivalent to the two conditions

$$\sup_{\|x\|=1} (1 - \|Ax\|^2) = 1 - \inf_{\|x\|=1} \|Ax\|^2 < 1$$

and

$$\sup_{\|x\|=1} (\|Ax\|^2 - 1) = \sup_{\|x\|=1} \|Ax\|^2 - 1 < 1,$$

which further translate to

$$\inf_{\|x\|=1} \|Ax\|^2 > 0 \quad \text{and} \quad \sup_{\|x\|=1} \|Ax\|^2 < 2.$$

■

With the aid of Lemma 7.11 it is easily shown that the operator R^{-2} can be approximated as a Neumann series. To do this it is first shown that for a proper choice of relaxation parameter (e.g., $\lambda = \sqrt{2}/(A + B)$) the operator $I - (\lambda R)^2$ is a contraction on the range $L(\mathcal{H})$. This is the content of Proposition 7.12.

Proposition 7.12 *Let $\{\phi_n\}$ be a frame for \mathcal{H} with frame representation L , correlation R , and bounds A, B . If $\lambda = \sqrt{2}/(A+B)$ then $\|I - (\lambda R)^2\|_{L(\mathcal{H})} < 1$.*

Proof: Take $A = \lambda R$ and $\mathcal{H}_1 = \mathcal{H}_2 = L(\mathcal{H})$ in Lemma 7.11. Then

$$\|A\| = \lambda \|R\| = \frac{\sqrt{2}}{A + B} B < \sqrt{2}$$

and

$$\inf_{c \in L(\mathcal{H}), \|c\|=1} \|\lambda R c\| > 0$$

since by Proposition 3.17(e) R is 1-1 on $L(\mathcal{H})$ and $\|c\| = 1$. The result follows from application of Lemma 7.11. ■

Let $\lambda = \sqrt{2}/(A + B)$. Since $I - (\lambda R)^2$ is a contraction, the Neumann series

$$\lambda^2 \sum_{k=0}^{\infty} (I - (\lambda R)^2)^k$$

will converge to R^{-2} on $L(\mathcal{H})$. Formally, with \tilde{c} an arbitrary element from $\ell^2(\mathbb{Z})$ and $c_0 \triangleq R\tilde{c}$

$$\lambda^2 \sum_{k=0}^n (I - (\lambda R)^2)^k c_0,$$

will converge to $(R^{-2})R\tilde{c}$ since $c_0 \in L(\mathcal{H})$. In fact,

$$\lim_{n \rightarrow \infty} \lambda^2 \sum_{k=0}^n (I - (\lambda R)^2)^k R \quad (7.4)$$

is the pseudo-inverse R^\dagger of R .

Algorithm 7.13 Let $\{\phi_n\}$ be a frame for a Hilbert space \mathcal{H} with frame representation L , correlation R , and bounds A and B . Suppose $\tilde{c} \in \ell^2(\mathbb{Z})$ is the corrupted frame representation of a signal $f_* \in \mathcal{H}$. Set $c_0 = R\tilde{c}$ and $d_0 = 0$. If $\lambda = \sqrt{2}/(A+B)$ and d_n and f_n are defined as

$$\begin{aligned} d_{n+1} &\triangleq d_n + (I - (\lambda R)^2)^n c_0, \\ f_n &\triangleq \lambda^2 L^* d_n, \end{aligned}$$

then

- (a) $\lim f_n = f_\dagger \triangleq L^* R^\dagger \tilde{c}$ and
- (b) $\|f_\dagger - f_n\| < M\alpha^n$, where $M < \infty$ and $\alpha \triangleq \|I - (\lambda R)^2\| < 1$.

Proof:

- (a) As in Algorithm 3.24, an elementary induction argument shows that

$$\forall n, \quad f_{n+1} = \lambda^2 L^* \left(\sum_{j=0}^n (I - (\lambda R)^2)^j \right) c_0.$$

Consequently we have

$$\lim f_n = L^* R^\dagger \tilde{c}.$$

- (b) Write

$$\begin{aligned} \|f_{n+1} - f_n\| &= \|\lambda L^*(I - (\lambda R)^2)^n \tilde{c}\| \\ &\leq \lambda^2 \|L^*\| \left(\|I - (\lambda R)^2\| \right)^n \|R\tilde{c}\| \\ &< M'\alpha^n, \end{aligned}$$

where $M' = \lambda^2 B^{3/2} \|\tilde{c}\| < \infty$ since $\tilde{c} \in \ell^2(\mathbb{Z})$. Thus,

$$\|f_\dagger - f_n\| \leq \sum_{k \geq n} M' \alpha^k = M' \frac{\alpha^n}{1 - \alpha} = M\alpha^n.$$

■

Iterative processes for the construction of generalized inverses have been well investigated (e.g., [Sho67], [Pet67], [Alt60]; cf. [Gro77] for a broad overview).

7.1.9 Measuring Noise (SNR)

In numerical simulations of real systems the ability to corrupt signals with a given level of noise is very important. In this section, terminology and a method for generating desired levels of noise are reviewed.

A common relative measure of noise in a corrupted signal is the signal-to-noise ratio. The SNR is defined as the ratio of signal energy to noise energy; that is,

$$SNR \triangleq \frac{\|f_*\|^2}{\|\sigma \cdot w\|^2}.$$

It is customary to express signal-to-noise ratios in units of decibels as follows

$$SNR_{dB} = 10 \log_{10} \left(\frac{\|f_*\|^2}{\|\sigma \cdot w\|^2} \right).$$

Thus, if the signal and noise were known separately then the SNR could be computed exactly. This fact may be used in numerical simulations to achieve precise levels of noise and is the procedure used in the MATLAB method

`add_noise`

discussed on page 284.

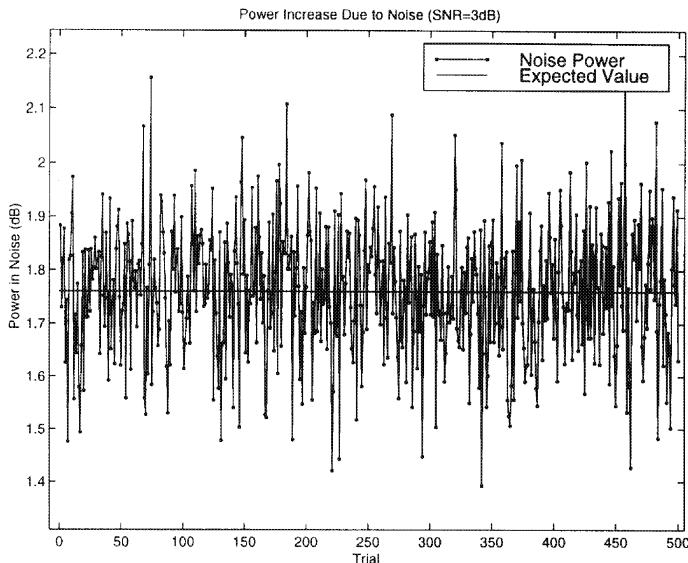


FIGURE 7.24. Computed power increase in signal due to additive noise with a SNR of 3dB.

Energy in the corrupted signal is related to the SNR as

$$\begin{aligned}\|f_* + \sigma \cdot w\|^2 &= \|f_*\|^2 + \|\sigma \cdot w\|^2 + 2 \operatorname{Re} \{\langle f_*, \sigma \cdot w \rangle\} \\ &= (1 + SNR^{-1}) \cdot \|f_*\|^2 + 2 \operatorname{Re} \{\langle f_*, \sigma \cdot w \rangle\}.\end{aligned}$$

Since the expected value of the last term is zero, the expected power (in dB) in a corrupted signal with signal-to-noise ratio of SNR is

$$\underbrace{10 \log_{10}(1 + SNR^{-1})}_{\text{noise power}} + \underbrace{10 \log_{10}(\|f_*\|^2)}_{\text{signal power}} \quad (7.5)$$

SNR	SNR (dB)	Power in Noise (dB)
inf	inf	0
10	10	0.414
2	3.01	1.76
1	0	3.01
0.5	-3.01	4.77
0.1	-10	10.4

TABLE 7.6. Expected power due to noise contributions.

Thus, the expected value of the increase in power due to the noise is clearly identified with the left term in (7.5). Expected values of the noise contribution to the power in a corrupted signal having a signal to noise ratio of SNR are enumerated in Table 7.6 for several values. The table shows, for example, that a signal additively combined with noise of sufficient power as to yield a signal-to-noise ratio of 3dB will exhibit (on average) an increase in power of 1.76dB. This fact is numerically illustrated in Figure 7.24 which depicts the measured power increase in a signal due to additive corruption at a signal-to-noise ratio of 3dB. Shown in the figure are the computed values of the power measured in the corrupted signal minus the power in the original signal for 500 trials. The expected value of 1.76 is also plotted as a horizontal line in the figure. As can be seen the sample values do indeed have this as their expected value (mean).

7.2 Compression

In this section, a general wavelet-based technique for the *lossy* compression of time–frequency coherent signals is described and numerically demonstrated. Both modified overcomplete and orthonormal wavelet representations are considered and compared. It is demonstrated that these modified overcomplete representations can provide superior compression performance as compared to related orthonormal representations.

7.2.1 Problem

Compression of signals is a main goal in applications that seek to minimize the storage or medium capacity needed to hold or convey the information contained in a signal. In this section, a general signal compression method is outlined for wavelet frames. Succinctly stated, the main objective addressed here is as follows.

To provide reduced storage representations of time–frequency coherent signals under the constraint that sufficiently high quality replicas of the originals are obtainable from the reduced representations.

7.2.2 Performance Measures

As a natural measure of compression, a “compression ratio” measures the relative decrease in complexity of data in a raw form as compared to the complexity of its new compressed form.

For speech signals, it is customary to deal directly with bit rates instead of compression ratios. This is because the bandwidth of speech is a fixed constant which, for some practical purposes, may be taken to be $\Omega = 4,000$ Hz. Consider the “raw” form of an analog speech signal f to be a sampled version with 8 bits per sample and a uniform sampling period of $T = 1/(2\Omega) = 1/8,000$ seconds. In this case, the required “raw” bit rate is 64Kbps. Since the hypotheses of the classical sampling theorem are satisfied, it is possible to reconstruct (modulo slight errors due to quantization) the original speech signal f . Any representation that allows for recovery of the original speech signal f and requires a bit rate less than 64Kbps is a compressed version of f .

A similar calculation can be made for signals from an arbitrary bandlimited space (with a bandlimit other than 4KHz). If a particular signal f has an effective bandlimit of Ω then the required bit rate using the classical sampling theorem with $b_c = 8$ bits per coefficient is

$$2\Omega \cdot b_c \quad (7.6)$$

with units of bits/sec (bps).

Bit Rate

Since, in practice, one may consider all signals of interest to be bandlimited, bit rates are a sufficient measure of compression. For numerical demonstration purposes, the FM data set of Section 6.4 is used in this section. For the FM signal data set, typical bandwidths are around 40MHz which leads to a classical bit rate of

$$2 \cdot 40 \times 10^6 \cdot 8 \text{ bps} = 640 \text{ bits}/\mu\text{s}.$$

Thus, for a 40MHz bandwidth signal, any representation of it that requires a bit rate of less than 640 bits/ μs is considered to be a *compressed* one. The smaller the bit rate is, the more compression there is.

Bit Rate Versus Error

Measuring compression alone is not a useful indicator of a compression scheme's performance since the obtainable quality of the reconstruction of the original signal from its compressed version must be sufficiently high. High fidelity reconstruction and low bit rate coding are competing objectives. One comes at the cost of the other. For a fixed compression scheme, a decrease in bit rate will, at best, cause the reconstruction error to remain the same, or more likely, to increase. For comparisons, a plot of reconstruction error (or distortion) versus the bit rate is an indicator of the quality of compression schemes (c.f. [Ber71]). Such a curve provides a tool for lossy compression to find the best compression available for a given level of tolerable distortion. For the FM data set, error versus bit rate curves are computed using the compression scheme presented in Section 7.2.3 and displayed in Section 7.2.4.

7.2.3 Approach

Compression problems also may be addressed using quantization in the wavelet domain. Since quantization may be viewed as an intentional coefficient disturbance, the robustness to coefficient noise that overcomplete representations offer plays an important role. The main idea is that for overcomplete representations, coefficients may be represented with fewer bits while not affecting reconstruction quality too much.

It is the wavelet frame coefficients that must be transmitted or stored. For representation in digital form, it is necessary that the frame coefficients be quantized. For simplicity, the quantization strategy⁷ employed here is one that maps values uniformly along some interval. This uniform mapping

⁷This is a major area for improvement of the compression scheme described. Nonuniform and/or vector quantization ([GG92], [Gra90]), of the wavelet coefficients should allow for substantially reduced bit rates with minimal loss of quality.

corresponds to specifying each coefficient with a fixed number of bits. The fixed number of bits that is allocated for the representation of each coefficient is denoted by b_c (bits/coef). Let the quantization level set $\{l_k\}_{k=1}^{2^{b_c-1}}$ be a sequence of increasing real numbers. The quantization function Q_{b_c} is defined in terms of the level set $\{l_k\}_{k=1}^{2^{b_c-1}}$ as

$$Q_{b_c}(x) = \begin{cases} 0, & |x| < l_1, \\ \text{sgn}(x) \cdot l_k, & |x| \in [l_k, l_{k+1}), \\ \text{sgn}(x) \cdot l_{2^{b_c-1}}, & |x| \geq l_{2^{b_c-1}}. \end{cases} \quad (7.7)$$

A *uniform* quantization is performed by the function Q_{b_c} if the level set $\{l_k\}_{k=1}^{2^{b_c-1}}$ has uniformly distributed elements; for example,

$$l_k = \left(\frac{2k}{2^{b_c} - 1} \right) \cdot M,$$

where

$$M \triangleq \|Lf\|_\infty,$$

and f is the signal to be quantized. Using the values $M = 1$ and $b_c = 3$, the uniform quantization of a straight-line function is plotted in Figure 7.25. Note that a representation Lf of a signal f is always reduced in magnitude by quantization; that is, $\|QLf\| \leq \|Lf\|$.

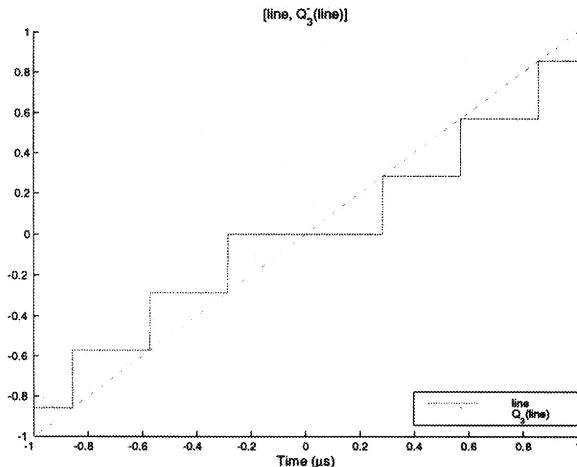


FIGURE 7.25. Uniform quantization of a line using three bits.

Figure 7.26 depicts the general scheme for compression of time–frequency coherent signals. Major elements and the processing flow of the model are explained in the following.

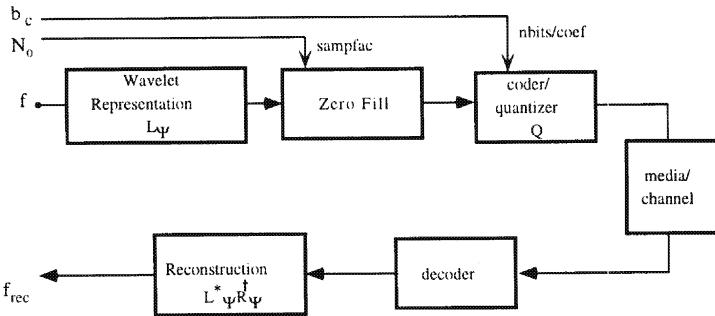


FIGURE 7.26. Overcomplete wavelet transform-based scheme for compression.

1. **(Signal f)** A time-frequency coherent signal $f \in \mathcal{H}$ with respect to the wavelet frame Ψ is input into the model.
2. **(Wavelet Representation L_Ψ)** The signal f is transformed to the wavelet domain to yield the wavelet coefficients $L_\Psi f$.
3. **(Zero Fill)** For a given sample rate factor $N_0 \in \mathbb{Z}$ the redundant representation $L_\Psi f$ is made less redundant by down-sampling by N_0 followed by up-sampling by N_0 . The net effect is to replace all but every N_0 th sample with zero. Explicitly, $Z : \ell^2(\mathbb{Z}) \mapsto \ell^2(\mathbb{Z})$ for a specific $c \in \ell^2(\mathbb{Z})$ as

$$(Z_{N_0} c)_n \triangleq \begin{cases} c_n, & n = kN_0, \text{ for some } k \in \mathbb{Z}, \\ 0, & \text{else.} \end{cases} \quad (7.8)$$

Zero filling clearly removes redundancy in time. One factor limiting the size of N_0 with respect to removing important signal information is the bandwidth of the wavelet. In general, small bandwidths should tolerate a higher level of zero filling.

4. **(Coding/Quantization Q)** To compress the wavelet coefficients further each must be assigned a number of bits to represent its value. Quantizing the coefficients performs this service. The specific quantization used here is uniform in nature, is described previously and given explicitly in Equation (7.7). Another issue that is not addressed here is the subsequent coding of the quantized bit stream in order to reduce even further the number of overall bits required to represent a given signal.
5. **(Media/Channel)** Presumably, the necessity for compression comes from the fact that the medium through which the information in a signal is to be conveyed has a limited capacity. This block in the diagram is dictated by the specific situation and translates into a compression requirement (e.g., a maximum sustainable bit rate).

6. **(Decoder)** Should the quantized coefficients be coded they must first be decoded before being presented to the reconstruction algorithm.
7. **(Reconstruction $L_\Psi^* R_\Psi^\dagger$)** Quantized and zeroed coefficients are used to reconstruct a version of the input via an appropriate reconstruction algorithm. Reconstruction algorithms are discussed in Section 7.1.3.

7.2.4 Numerical Experiment

Based on the compression scheme presented in Section 7.2.3, Experiment 7.14 details a numerical procedure for investigating the compression ability of general wavelet transform representations.

Numerical Experiment 7.14 (Compression) *Let $\{FB_m\}$ be a fixed set of filter banks with respect to the wavelet g_m . Fix a set of zero fill sample factors $\{N_{0k}\}$ and a set of fixed numbers of bits to assign to each representation coefficient $\{b_{cj}\}$.*

1. *Synthesize a set of time-frequency coherent signals $\{f_n\}$.*
2. *For each filter bank FB_m specified, each signal f_n , each number of bits b_{cj} , and each zero fill sample factor N_{0k} do the following.*
 - (a) *Compute the OCWT representation $W_g f_n$.*
 - (b) *Zero fill the representation using N_{0k} to yield*

$$Z_{N_{0k}} W_g f_n.$$

- (c) *Quantize the zero filled representation using b_{cj} bits per coefficient to yield the compress representation $W_Q f_n$:*

$$W_Q f_n \triangleq Q_{b_{cj}}(Z_{N_{0k}} W_g f_n).$$

- (d) *Compute the processed reconstruction f_{rec} :*

$$f_{\text{rec}} = W_g^{-1} W_Q f_n.$$

- (e) *Compute the effective bit rate bps:*

$$\text{bps} = \frac{b_c \cdot \text{card } \{|W_Q| > 0\}}{\text{supp } f_n},$$

the number of nonzero coefficients in the compressed representation times the number of bits per coefficient divided by the signal duration.

(f) Compute the relative reconstruction error:

$$ERR = \frac{\|f_n - f_{\text{rec}}\|}{\|f_n\|}.$$

For the data set of FM signals described in Section 6.4 whose frequency-modulations are depicted in Figure 6.6, the compression Experiment 7.14 has been run. Parameters associated with this run are listed in Table 7.7.

Parameter	Symbol	Values
#Bits/Coef	b_c	2, 3, 4, 5, 6
Sampfac	N_0	4 8 16 32
Dilation Const.	a_0	1.12, 1.08, 1.045
# Filters	n_f	20, 28, 48
Wavelet	g	$g_{\text{Morlet}}(CF = 40, BW = 5)$
Sample Period	Δ	1/128
Interval	T	4

TABLE 7.7. Parameters for Numerical Experiment 7.14.

Three different Morlet filter banks⁸ were used in the compression experiment with variations in only the number of filters n_f and dilation constant a_0 . These filter bank parameters were chosen such that the bank covered the frequency range of 5–40MHz. Figure 7.27 shows the filter bank support functions G for each of the three filter banks. Note that all the support functions cover the requisite area but that filter bank 1 (FB1) has larger variations than FB2, and FB2 has larger variations than FB3. Thus for the same time sampling geometry FB1 is much less tight than FB3. All parameters associated with the three filter banks are listed in Table 7.8.

Name	g	a_0	n_f
FB1	$g_{\text{Morlet}}(CF = 40, BW = 5)$	1.12	20
FB2	$g_{\text{Morlet}}(CF = 40, BW = 5)$	1.08	28
FB3	$g_{\text{Morlet}}(CF = 40, BW = 5)$	1.045	48

TABLE 7.8. Filter banks used in Numerical Experiment 7.14.

⁸Note that the filter banks used here are not L^2 normalized, but L^∞ -normalized in frequency; that is, each filter transfer function has maximum magnitude of 1.

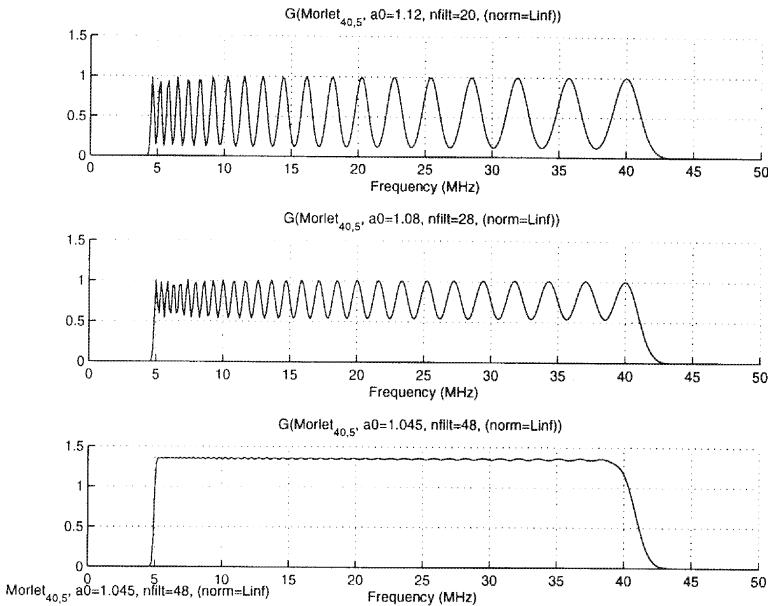


FIGURE 7.27. Filter bank support functions G of Equation (6.3) for the three filter banks used in Numerical Experiment 7.14.

For the case of filter bank FB1, Figures 7.28 through 7.33 show the results of Numerical Experiment 7.14 for each of the six FM signals, respectively. On the left side of each figure the relative error versus bit rate is plotted for the four values of sample rate factor N_0 given in Table 7.7. The best compression achieved for a level of error around 5% is indicated by a \star symbol. Stacked to the right of this plot are

1. the top plot which shows the original signal, and
2. the bottom plot which shows the reconstruction.

Reported in the title of the reconstruction plot is the associate rate and error values which should correspond to the \star point on the error versus rate curve.

Table 7.9 reports the error versus effective bit rates achieved with the three different Morlet OCWT filter banks as well as the orthonormal representation with respect to the Daubechies order 6 wavelet.

For comparison, the performance of the compression scheme using an orthonormal wavelet representation is shown in Figure 7.34. In a 3×2 grid of plots, all six error versus bit rate curves are displayed in the figure. There, the Daubechies wavelet of order 6 is used in Numerical Experiment 7.14 to create these plots where $N_0 = 1$ (no zero filling) and quantization

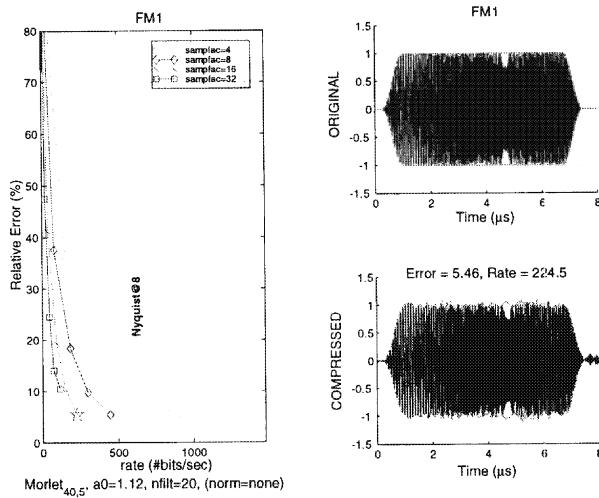


FIGURE 7.28. OCWT compression for signal FM1.

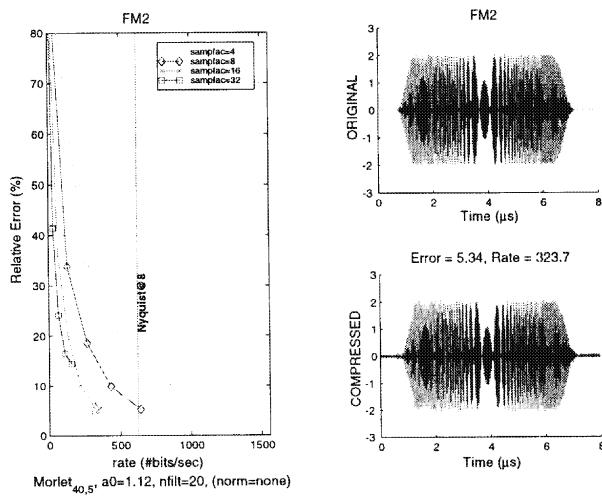


FIGURE 7.29. OCWT compression for signal FM2.

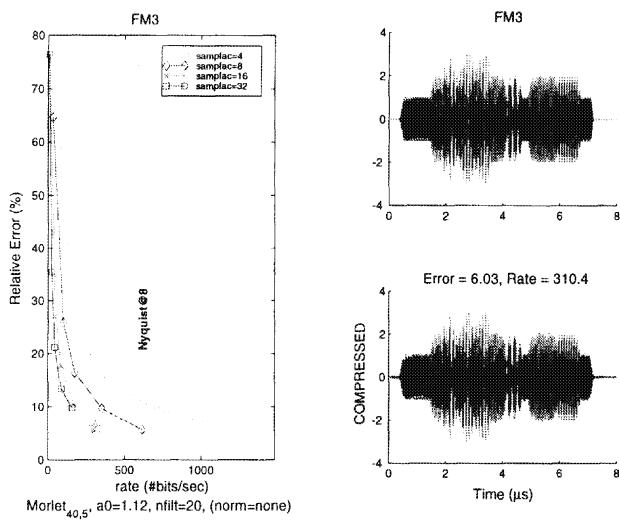


FIGURE 7.30. OCWT compression for signal FM3.

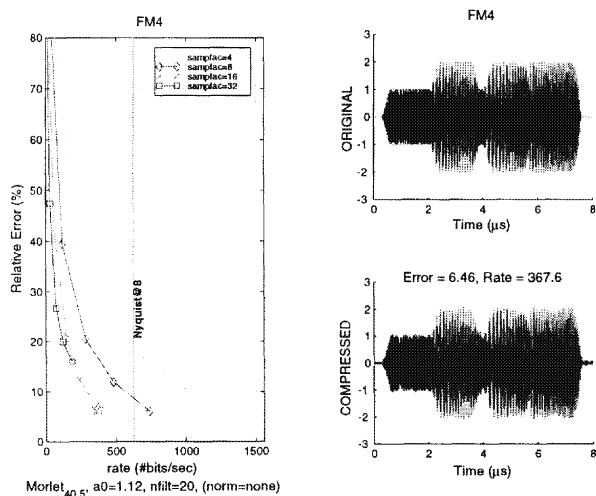


FIGURE 7.31. OCWT compression for signal FM4.

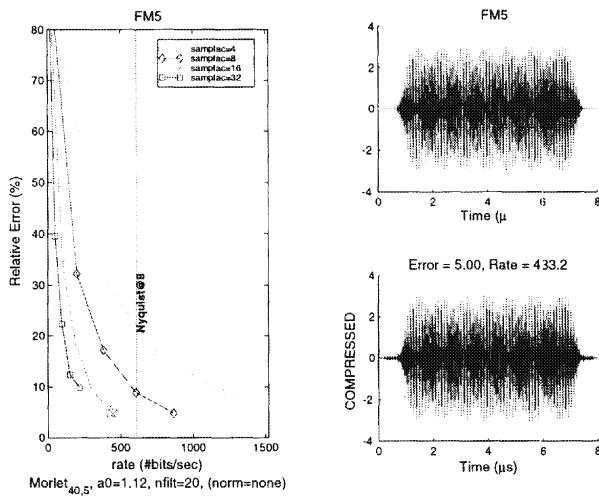


FIGURE 7.32. OCWT compression for signal FM5.

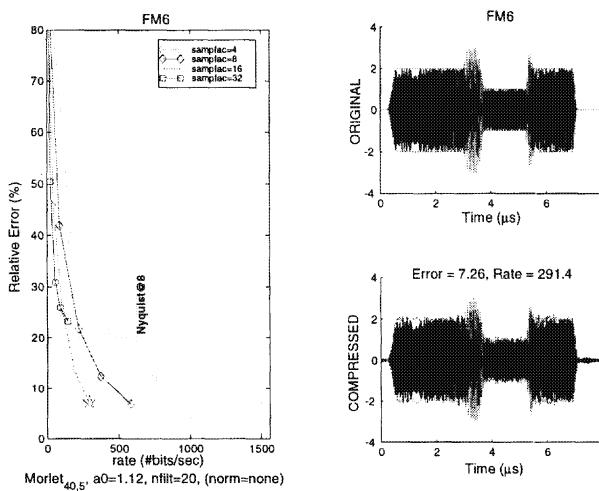


FIGURE 7.33. OCWT compression for signal FM6.

		FM1	FM2	FM3	FM4	FM5	FM6
FB1	Error	4.06	3.73	5.18	4.51	5.51	6.18
	Rate	326	469	486	551	308	409
FB2	Error	5.46	5.34	6.03	6.46	5.00	7.26
	Rate	225	324	310	368	433	291
FB3	Error	5.69	5.59	6.84	6.68	5.88	4.83
	Rate	278	567	413	628	385	737
Daub6	Error	4.83	5.71	6.39	5.37	5.09	5.47
	Rate	456	674	617	675	716	600

TABLE 7.9. Compression results for Numerical Experiment 7.14.

levels of 4 through 8 bits are used. As before, the approximate 5% error level on the curve is indicated by a star. From the figure it can be seen that for comparable error levels (around 5%), the required bit rate using this orthonormal wavelet is higher than that using overcomplete wavelets. In some cases the required rate is above that of the Nyquist limit of Equation (7.6). Actual values are listed in Table 7.7.

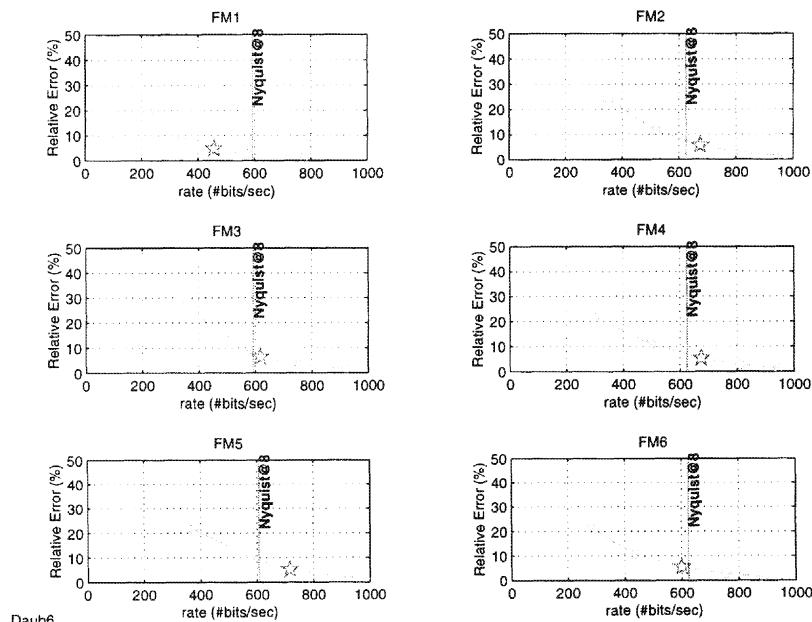


FIGURE 7.34. Compression for FM signal set using Daub6 orthonormal wavelet representation

7.2.5 Remarks

A wavelet-based scheme for the compression of time–frequency coherent signals has been proposed. The scheme hinges on the idea of representing signals in an overcomplete wavelet representation (expansion) and then severely reducing redundancy through zeroing. The principle demonstrated here is that in order to compress something a lot it may be useful to first expand it.

Examining the results of the compression experiment one may make the following anecdotal observations.

1. For a given level of error tolerance, overcomplete-based compression seems to yield better compression than orthonormal (Daub6) compression.
2. Overcomplete representation coefficient may be coded using only a few bits to achieve a good level of error.
3. Nontight and redundant frames *are* useful for compression.

There are other variables, tradeoffs, and issues involved evaluating the compression scheme. Some of them are listed in the following.

- (**Zero Coding**) The key element to achieving good compression performance here is the zero filling operation. Zero filling results in modified representations that are very sparse (i.e., having mostly zero coefficients). This leads to dramatically reduced bit rates if somehow only nonzero coefficients would contribute significantly to the number of bits necessary to code the resulting sequence. Presumably, the zeros will contribute some, although small, number of bits to the resulting coded representation. For this reason, bit rates reported here are slightly larger than may be expected to be realizable.
- (**Reconstruction**) Because of the nature of the semilog-regular sampling set associated with the OCWT representations used here, the direct reconstruction method of applying the OCWT inverse is valid. The application of the inverse directly to a zero filled representation yields a multiplicatively scaled version of the full reconstruction. Thus, reconstructions are scaled by a factor related to the sample rate factor N_0 used in the zero filling of Equation (7.8).
- (**Dilation Parameter**) In effect, the dilation parameter a_0 changes the frequency support of each filter in the filter bank. In particular, increasing the value of a_0 decreases the bandwidth of each filter in the bank. A decrease in bandwidth necessarily implies that a particular filter will respond to a smaller band in frequency. Thus, keeping f the same, an increase in the value of a_0 will cause the bands of activity in the original a_0 representation to become compressed along the

s -axis in the new increased a_0 representation. On the other hand, larger values of a_0 will cause the function G of (6.3) to have greater variation from constant value. This condition necessarily implies a spread between possible frame bounds, that is, movement away from tightness. This, in turn, can be related to a slowing of the rate of convergence of the reconstruction Algorithm 7.13. The OCWT direct implementation used here, however, does not have this drawback.

7.3 Digital Communication

In this section, an approach to the problem of secure digital communications is discussed. A key idea is to use overcomplete wavelets as building blocks or *atoms* for the construction of communications signals. The approach is based upon two major novel components:

- hybrid time–frequency allocation of communication channel resources, and
- overcomplete, that is, overlapped, multiplexed signals.

Together, these two components should allow for the secure transmission of a large number of multiple user messages within a given bandwidth. In addition, the design method yields multi-user messages that are robust to additive white Gaussian noise in the channel and allows error-free reconstruction in high noise levels having signal-to-noise ratios as low as 2dB.

7.3.1 Problem

The main interest within multiple access communication is directed towards the construction of low probability of intercept (secure) transmission signals that satisfy channel constraints, have a low probability of bit error, and exhibit high throughput.

In a multiple access setting, one may think of N users who each must transmit a single bit b_k , $k = 1, 2, \dots, N$ on a shared medium, for example, a channel. Alternatively, one may think more simply in terms of a raw number of bits $\{b_k\}_{k=1}^N$ that must be communicated through the channel. This view is slightly more general in that it allows for the raw bits to be allocated among users in a flexible way. Moreover, the channel constraints are thought of as a time–frequency region \mathcal{R} in which signals for transmission are forced to reside. For example, \mathcal{R} may be a rectangular region that describes both the channel bandwidth and the time window in which a communication signal must be transmitted and decoded.

The main question addressed is stated as follows.

How is the time–frequency region \mathcal{R} parsed into smaller *atomic* regions such that the objectives of low probability of error, high security, and high throughput are met?

Atomic regions that are defined by (near arbitrary) wavelet families are proposed to perform this partitioning. In particular, sets of nonorthogonal wavelet functions (i.e., functions that are related to each other by pure dilation and translation and have, in general, a nonzero correlation with each other) are used to encode and decode digital information for analog communication.

7.3.2 Objectives

The main objective is that of employing nonorthogonal wavelet functions as the basic building blocks for the construction of communication signals. To fulfill this objective, the following subobjectives have been developed and are shown to be satisfied by the proposed communication schemes. In particular, the subobjectives are to design an analog transmission signal construction procedure and an associated recovery procedure such that

- (**Resource Allocation**) medium (channel) constraints are satisfied,
- (**High Throughput**) throughput is maximized (number of users),
- (**Noise Robust**) recovery of user data is robust to noise, and
- (**Secure**) recovery of user data has low probability.

To demonstrate the feasibility of the proposed approach specific objectives are identified:

1. determine methods to design a best atomic set of nonorthogonal wavelet functions;
2. develop algorithms for user bit vector recovery;
3. formulate performance metrics that quantify throughput, noise robustness, and secure communication; and,
4. design and perform numerical experiments to demonstrate
 - (a) channel throughput,
 - (b) noise robustness, and
 - (c) security.

Section 7.3.3 presents the technical approach and supporting background material. Discussed there are mathematically precise methods to construct (modulate) and recover (demodulate) signals for the communication of digital information. Various quantitative tools to evaluate the performance

of the wavelet communication scheme are presented in Section 7.3.4. In Section 7.3.5 a specific implementation of the wavelet modulation and demodulation schemes is presented, as well as a look at some of the numerical tools that have been developed to demonstrate the communication scheme. Numerical results of simulation tests are given in Section 7.3.7.

7.3.3 Approach

In this section the details of the approach to the communication of digital information using a fundamentally analog medium are described.

As a solution to the communication problem, a natural and widely practiced approach follows the three-step model of encode, transmit, and decode. In succinct terms, the goal of a digital communication system is to convey a set of raw bits from one location to another. Typically such communication is facilitated by the three-step process:

1. encode,
2. transmit over a given channel, and
3. decode.

Consequently, the main areas of focus are

1. (**Encoding**) formulate a transmission signal construction procedure in which to encode the desired information (i.e., the user bit vector $b = \{b_k\}$); and
2. (**Decoding**) formulate a procedure to recover the transmitted information from the observed (received) signal.

The approach hinges on the idea of using nonorthogonal sets of wavelets as building blocks or *atoms* for the encoding of signals for transmission over the given channel.

Figure 7.35 illustrates the overall communication concept using digital wavelet processors. From the figure it is seen that both the encoder and decoder consist of the elements:

1. digital wavelet processing,
2. analog to/from digital conversion, and
3. transmitter/receiver.

Transmission Signal Construction

The general approach to constructing transmission signals consists of specifying a set of atoms $\Phi \stackrel{\Delta}{=} \{\phi_k\}$ from which to build a transmission signal

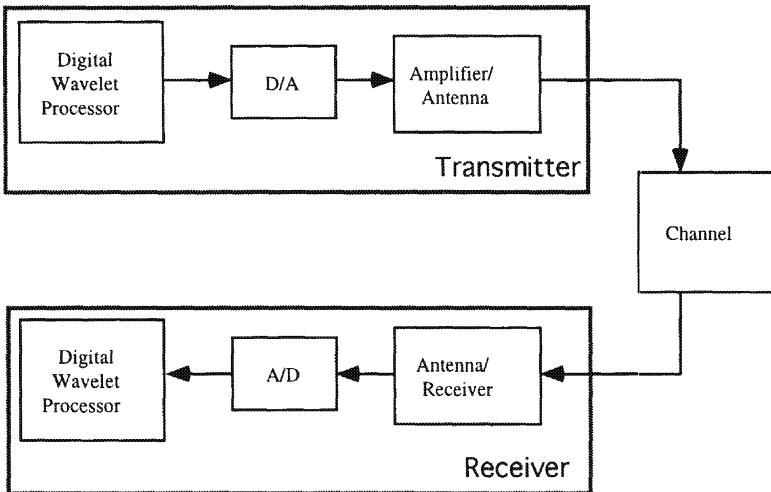


FIGURE 7.35. Digital communication system using a wavelet encoder in the digital transmitter and a wavelet decoder in the digital receiver.

f. A transmitted signal f has the form

$$f = \sum_{k=1}^N b_k \phi_k, \quad (7.9)$$

where N is the number of bits to be transmitted, $b = \{b_k\}$ is the bit vector, and the atomic functions $\{\phi_k\}$ are subject to design.

In conventional multiple access approaches, the atoms $\{\phi_k\}$ with which the transmission signal may be built are required to be orthogonal. For instance, in the common multiple access schemes of frequency division (FDMA) and time division (TDMA) the atoms are made orthogonal by ensuring that they do not overlap in frequency or time, respectively. In these two approaches the available time–frequency region \mathcal{R} of the channel is allocated in nonoverlapping strips parallel to one of either the time or frequency axes. Very recently, wavelet packet-based approaches ([LKC⁺94] and [CW92]) to the allocation problem have emerged, that is, wavelet packet multiple access (WPMA). In all of the proposed WPMA approaches, however, a constraint that the wavelet packet atomic functions be orthogonal is enforced.

In the proposed approach, the atomic set is chosen to consist of nonorthogonal wavelet functions. This allows for the time–frequency region \mathcal{R} to be partitioned in almost arbitrary and overlapping chunks. A schematic view of the various approaches to channel allocation is illustrated in Figure 7.36.

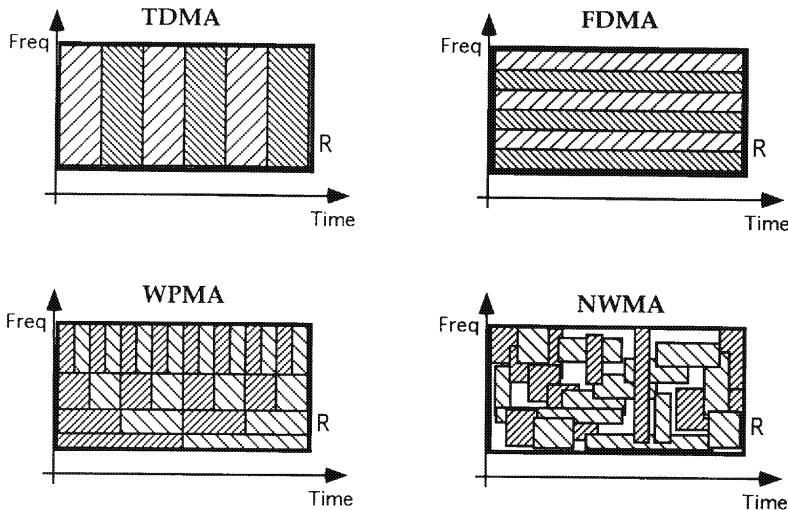


FIGURE 7.36. Various approaches to channel allocation: TDMA, FDMA, WPMA, and NWMA.

Recovery

The received signal f_r observed through noise is

$$f_r = \sum_{k=1}^N b_k \phi_k + \sigma \cdot w, \quad (7.10)$$

where w is a normalized noise process, and σ indicates the associated noise level. The objective is to robustly recover the b_k s from knowledge of only the received signal f_r .

Let L be the frame operator with respect to the functions $\{\phi_n\}$. Recall $L : L^2(\mathbb{R}) \mapsto \ell^2(\mathbb{Z})$ defined as

$$Lf = \{\langle f, \phi_n \rangle\}$$

where

$$\langle f, \phi_n \rangle \triangleq \int f(t) \bar{\phi}(t) dt$$

is the L^2 inner product.⁹ For a given n the inner product $\langle f, \phi_n \rangle$ gives a (complex) numerical value whose magnitude indicates the degree of correlation of the analyzed signal f with the atomic ϕ_n . In this way one might

⁹ $L^2(\mathbb{R})$ (the set of all finite energy signals on the real line) is the space in which the (analog) signals for transmission reside and $\ell^2(\mathbb{Z})$ (the set of all finite energy sequences) includes as a subset the users' bit vector.

think of L as a compilation of matched filters matched to the atomic set $\{\phi_n\}$. It is easy to compute the adjoint $L^* : \ell^2(\mathbb{Z}) \mapsto L^2(\mathbb{R})$ as

$$L^*b = \sum b_k \phi_n.$$

In terms of the adjoint operator L^* the transmitted signal of Equation (7.9) is

$$f = L^*b.$$

With $R \triangleq LL^*$, a procedure for recovering the user bit vector b from observation of the received signal f_r consists of two main components: application of the “inversion” operator $R^{-1}L$, and search for the closest match to the inversion in the bit vector space \mathcal{B} . Under the assumption that the atomic set Φ forms a frame for \mathcal{H} , R is invertible on the range $L(\mathcal{H})$ of L . This fact allows us to apply the operator $R^{-1}L$ on the received signal f_r to yield

$$\begin{aligned} R^{-1}Lf_r &= R^{-1}LL^*b + R^{-1}L(\sigma \cdot w) \\ &= P_{L(\mathcal{H})}b + \sigma \cdot R^{-1}Lw \\ &= P_{L(\mathcal{H})}b + \sigma d, \end{aligned}$$

where d is a bounded coefficient disturbance. For zero noise levels with optimally designed atomic sets Φ it is clear that $R^{-1}Lr$ is exactly $P_{L(\mathcal{H})}b$ in $\ell^2(\mathbb{Z})$. It is also clear that for modest noise levels $R^{-1}Lf_r$ will be closest to $P_{L(\mathcal{H})}b$ in $\ell^2(\mathbb{Z})$. Thus, the noise robust estimate \tilde{b} for b can be written as

$$\tilde{b} = \underset{b \in \mathcal{B}}{\operatorname{argmin}} \|R^{-1}Lr - P_{L(\mathcal{H})}b\|. \quad (7.11)$$

In Section 7.3.5 a simpler, but related, energy detector bit recovery procedure is presented. As an energy detector, it may be more amenable to specific types of hardware implementation.

7.3.4 Performance Measures

For the encoding of communication signals, the superiority of one particular wavelet method over another is, of course, determined by how well each fulfills the main objectives of secure, high throughput, and low bit error communication. In this section, several numerical and graphical *indicators* are presented that quantify the performance of a given method with respect to the given objectives. Not only does this allow us to give precise limits on the performance of the method, but it also facilitates quantitative comparisons between competing methods. Even more, it allows the formulation of an optimization approach that seeks to find the “best” method in a well-defined numerically precise manner.

Of the four objectives listed in the previous section the first, resource allocation, is inherently satisfied by the wavelet method since the underlying wavelet atomic set is constrained to reside within the channel's time-frequency domain of definition. Thus, there is no need to formulate a special indicator for this objective. For the remaining three, however, the numerical indicators described in the following sections are identified.

Throughput

One main interest in a digital communication system is the amount of information that can be transmitted from one location to another in a given amount of time over a given channel, that is, the rate at which information may be communicated. It is well known in information theory that there is a fixed upper bound on this rate which is dictated by the channel and called the channel capacity. For convenience, it is assumed that the channel is described by a given rectangle in the time-frequency plane.

Another issue of concern is the number of users that may be supported in a given multiple access scheme. This statistic is not explicitly addressed in the performance indicators since it can be equivalently viewed in terms of information carrying capacity, that is, throughput. This is easily seen by simply considering multiple users as contributing their bits to a common pool. Of course, the actual implementation of a multiple access scheme requires some additional overhead to recover just a specific user's bits.

Since digital information is measured in terms of bits, an obvious measure of information throughput is simply the rate measured in units of bits per second (b/s). For a fixed channel this sort of figure of merit is sufficient; however, comparison over different channels is not easily covered by this simple definition. To include the possibility of varying channels as well, it is convenient to introduce an additional parameter: the channel bandwidth (measured in Hertz). This leads to the idea of the spectral efficiency of a digital communication scheme.

Spectral Efficiency

The notion of *spectral efficiency* is presented in [Feh95] and is defined as

$$\nu_s = \frac{n_b}{\Omega T},$$

where

$$\begin{aligned} \nu_s &= \text{spectral efficiency (bits/sec/Hz)} \\ n_b &= \text{raw throughput (b/s)} \\ \Omega &= \text{bandwidth (Hz).} \end{aligned}$$

As its units reflect, the spectral efficiency of a communication scheme measures the achievable throughput per unit bandwidth (Hz) of a channel.

None of the discussion so far has addressed the issue of errors at the decoder. Yet it is a very important one since throughput alone, in terms of bits decoded at the receiver, is not a sufficient measure of overall system performance. It is also necessary to take into account the associated error rate. Ideally, one would like to maximize throughput with no errors at the decoder. In real systems, however, some level of error must be tolerated in order to achieve higher throughput rates.

For these reasons, a common performance specification is the curve specified by the bit error rate (BER) or probability of bit error as a function of throughput, or better still, the spectral efficiency. There should be a monotonic relationship between the two; that is, as throughput increases so does the BER.

Efficiency Versus BER

As discussed previously the goals of maximizing spectral efficiency and minimizing the probability of error (BER) are competing objectives. Since both quantities are of direct interest in terms of communication performance, a useful performance tool is a graph of the scheme's measured BER versus its spectral efficiency.

Noise Robustness

Another factor of importance to the operation of a digital communication system is the effect that channel noise has on the decoding process. For a good communication scheme, it is reasonable to expect that the BER will be optimally small in the case where there is no noise on the channel. This is an ideal situation that never occurs in practice. Instead there are unknown nondeterministic sources of perturbation in the channel that affect the communication signal in usually undesirable ways. Thus, it is imperative to have a good understanding of how a particular communication scheme operates under different types of noise and noisy conditions.

The most widely used noise model in communication systems is additive white Gaussian noise. Although there is noise in certain communication channels that is bursty or impulsive in nature, the AWGN channel has proven to be an adequate model for design even for systems meant to operate in impulsive conditions ([Gib93, Section 10.3]).

SNR Versus BER

To measure the noise robustness of a communication scheme the relationship of the BER for different levels of noise is a useful performance tool. The BER is computed under various signal-to-noise ratio conditions to yield a graph of SNR versus BER. Here the noise is assumed to be Gaussian and additive in nature (i.e., AWGN), with a variance set to reflect the desired SNR.

Security

A key idea behind the wavelet encoding and decoding schemes proposed here is that the ability to decode the received message is severely diminished when knowledge of the underlying wavelet atoms and their associated discrete sample set in the time–frequency plane is denied to the decoder.

Thus, one conservative method of measuring communication security is to decode incoming wavelet messages with respect to “guessed” wavelets and associated discrete sets and then compute the standard performance curves of Section 7.3.4. Poor performance in this case would indicate a high degree of security.

7.3.5 Computer Implementation

Wavelet Modulation

The fundamental problem of interest is to communicate a set of digital data (ones and zeros) or a bit stream through an analog communication channel. At a minimum, this requires a method to encode the digital data into some analog communication signal suitable for broadcast through the channel. Doing this in such a way so as to maximize throughput (bit rate), minimize errors (bit error rate), and minimize the probability of third party interception and decoding (security) is one of the main challenges in communication system design. It is a basic contention in this effort that the nonorthogonal wavelet modulation described here meets these challenges.

For this problem some assumptions are made:

1. the communication channel is characterized by a time–frequency rectangle; and
2. bits are encoded in bundles of *cells* each containing N -bits where N is determined by the channel and modulation scheme parameters.

Mathematically, the main tool employed for the encoding process is the overcomplete wavelet transform. Details concerning the OCWT are presented in Chapter 6 and not repeated here except to reiterate that the OCWT is determined by the specification of an analyzing wavelet g and a discrete sample set $\{t_{m,n}, s_m\}$ in the time–scale plane.

Suppose that $\{b_{m,n}\}$ is a set of N bits (doubly indexed) to be communicated. The fundamental idea of the wavelet encoding scheme is to assign to each element in the set of N bits $\{b_{m,n}\}$ the wavelet function

$$\tau_{t_{m,n}} D_{s_m} g, \quad (7.12)$$

that is, a translated by $t_{m,n}$ and dilated version of the analyzing wavelet g . Clearly, this results in a corresponding set of N wavelet atoms. The major design parameters in this scheme are clearly

1. the wavelet g , and
2. the discrete time-scale set $\{t_{m,n}, s_m\}$.

Optimal methods for their selection is one focus of this effort and is addressed numerically in Section 7.3.6. Within this framework it should be evident that the throughput (i.e., the number of bits encoded within the given time-frequency rectangle) is directly related to the density of sample points in the time-scale plane. In fact the number of bits transmitted and the number of sample points are exactly the same. The tradeoff in increased density is an increased bit error rate in the decoding. From this perspective, the main objective may be restated in general terms as:

to specify as dense as possible a sample set in the time-scale plane that has an associated bit error recovery rate which falls within acceptable limits.

Recovery of the bit data is discussed in Section 7.3.5.

By the preceding association an analog communication signal is constructed from the bit set using the wavelet atoms as building blocks and the values of the bits (0 or 1) as indicators of which atoms are included in the construction. Precisely, an analog communication signal f is constructed from the bit set $\{b_{m,n}\}$ as

$$f = L^* b = \sum_{m,n} b_{m,n} \cdot \tau_{t_{m,n}} D_{s_m} g, \quad (7.13)$$

where L is the representation operator associated with the wavelet atoms.

Provided that the discrete sample set and wavelet are chosen appropriately, the resulting communication signal f will be suitable for transmission via the channel. Namely, if all the atoms in the selected wavelet set essentially live within the time-frequency rectangle characterizing the channel then the composite signal f will be supported by the channel. In this regard, it is worth explicitly mentioning that this method results in a signal f that may be directly applied to the channel, and, in particular, no further conventional modulation is necessary. Explicit examples of this construction are presented in Section 7.3.6.

Wavelet Demodulation

Given a possibly corrupted version of the transmitted signal f of (7.13) it is the objective of the demodulator to recover the embedded bit set $\{b_{m,n}\}$. The procedure used to do this is fundamentally based on an analysis of the energy distribution of the OCWT of f with respect to the wavelet constructor atoms. Simply stated, the function of the wavelet demodulator is to perform the OCWT with respect to the wavelet atoms of (7.12) and subsequently measure the energy in the regions where the atoms live. Regions with a high enough measured energy are deemed as having an associated bit value of one and zero otherwise.

Time-Frequency Masks

Explicitly, the wavelet demodulator requires knowledge of

1. the generating wavelet g ,
2. the discrete time-scale set $\{t_{m,n}, s_m\}$, and
3. an essential support factor $R_{\text{fac}} \in [0, 1]$.

This, in turn, describes a set of time-frequency masks $\{M_{m,n}\}$ which are time-frequency indicator functions defined as

$$M_{m,n}(t, \gamma) = \begin{cases} 1, & |(\tau_{t_{m,n}} D_{s_m} g)(t)| > R_{\text{fac}} \cdot \|g\|_\infty \text{ and} \\ & |(\tau_{t_{m,n}} D_{s_m} g)^\sim(\gamma)| > R_{\text{fac}} \cdot \|\widehat{g}\|_\infty, \\ 0, & \text{else,} \end{cases}$$

where $\|f\|_\infty$ is the sup-norm (simple maximum for analytic functions). These masks define rectangles

$$R_{m,n} \triangleq \{(t, \gamma) : |(\tau_{t_{m,n}} D_{s_m} g)(t)| > R_{\text{fac}} \cdot \|g\|_\infty\} \cap \{(t, \gamma) : |(\tau_{t_{m,n}} D_{s_m} g)^\sim(\gamma)| > R_{\text{fac}} \cdot \|\widehat{g}\|_\infty\} \quad (7.14)$$

(for well-behaved single humped analyzing functions as considered here). Each wavelet atom essentially lives in one of these rectangles in the sense that R_{fac} percentage of their energy is concentrated in that rectangle. See Figures 7.38 and 7.39 in Section 7.3.6 for a numerical example.

The demodulator uses the further scalar thresholds

1. detection threshold $\delta_d \in [0, 1]$, and
2. a noise rejection threshold $\delta_n > 0$.

These are discussed in the following subsections.

Noise Rejection

The noise threshold δ_n is used as a hard threshold in the wavelet domain to yield a thresholded wavelet transform $W_{g,\delta_n} f$ of a signal f as

$$(W_{g,\delta_n} f)(t, s) = \begin{cases} (W_g f)(t, s), & |(W_g f)(t, s)| > \delta_n \cdot \|W_g f\|_\infty, \\ 0, & \text{else.} \end{cases}$$

It is well known that a thresholding operation of this sort in the wavelet domain performs noise suppression in the signal domain. To see this consider a noise-corrupted signal as consisting of a coherent portion (the signal) and an incoherent portion (the noise). Here coherence is defined with respect to a set of (wavelet) atomic functions: namely, a function is coherent with respect to a set of functions if the energy in the function is distributed among relatively few coefficients associated with the inner product representation

of the signal. In other words, the signal is well correlated with (a small number) of the underlying atomic functions. The incoherent portion of a corrupted signal f will necessarily have its energy spread throughout the time–frequency plane whereas the coherent portion (e.g., a wavelet atom itself) will have its energy concentrated in a small area in the time–frequency plane. Thus, the coefficients associated with the incoherent portion of a corrupted signal are necessarily small as compared to those associated with the coherent portion. Hence, a thresholding operation will naturally reject incoherence and preserve strong coherence.

Energy Threshold Detection and Bit Recovery

Given the rectangles $\{R_{m,n}\}$ of (7.14) used to construct the modulation signal of (7.13) a simple energy threshold detector may be employed to recover the embedded bit information. This bit recovery process measures the energy in each region $R_{m,n}$ of the thresholded wavelet transform $W_{g,\delta_n} f$ of the received and possibly noise-corrupted signal f . Denoting $e_{m,n}$ as this energy gives precisely that

$$e_{m,n} = \int_{R_{m,n}} |W_{g,\delta_n} f(t, s)|^2.$$

From this set of energies the embedded bits $\{b_{m,n}\}$ are estimated by

$$d_{m,n} = \begin{cases} 1, & e_{m,n} > \delta_d, \\ 0, & \text{else.} \end{cases} \quad (7.15)$$

Here, $d_{m,n}$ is a decision function for the (m, n) rectangle. A proper value of the detection threshold δ_d may be arrived at by consideration of the quantity

$$\int_{R_{m,n}} |W_{g,\delta_n} (D_{s_m} \tau_{t_{m,n}} g)|^2 = \int_{R_{m,n}} |W_{g,\delta_n} g|^2 = E = \text{const.}$$

The quantity E is independent of the values of s_m and $t_{m,n}$ because the dilation and translation operators are unitary; that is, the energy in each wavelet atom is the same. E is the amount of energy that is transferred to the time–frequency plane in the region $R_{m,n}$ by a single wavelet atom when included in a signal (i.e., $b_{m,n} = 1$). In noiseless and nonoverlapping conditions the energy $E_{m,n}$ is ideally zero when the corresponding atom is not included in the signal (i.e., $b_{m,n} = 0$). Therefore, the detection threshold δ_d should take some value between 0 and E .

Caveats

There are some caveats that must further be addressed if the wavelet communication scheme is to be successful. Some of these are discussed in the following paragraphs.

Synchronization

Because it is continuously receiving the channel signal, the receiver has no inherent knowledge of the start of a given bit packet. This points to a need to synchronize the masking process with the incoming signal. Lack of synchronization will likely result in unacceptably high bit error rates. One possible solution is to reserve one or two bits in the modulated signal as synchronization (or reference) bits. The idea here is that the position of the time-frequency rectangular bit masks $R_{m,n}$ will be relative to the reference rectangle(s).

Hardware Implementation

The wavelet processing described here is numerically computationally intensive. There is a need to examine possible hardware architectures for the implementation of the scheme and any associated intrinsic limitations.

7.3.6 Wavelet Communication Testbed

Clearly there is a fair amount of design freedom associated with the overall wavelet modulation and demodulation scheme discussed here. This freedom primarily lies in the choice of the specific values of the different parameters:

1. the wavelet g , and
2. the discrete time-scale set $\{t_{m,n}, s_m\}$,
3. an essential support factor $R_{\text{fac}} \in [0, 1]$ which together with the preceding specify the rectangles $R_{m,n}$ of (7.14),
4. detection threshold $\delta_d \in [0, 1]$, and
5. a noise rejection threshold $\delta_n > 0$.

Needless to say, there are tradeoffs and limitations associated with each choice. To help understand these different tradeoffs, a graphically oriented MATLAB-based numerical testbed for the wavelet encoding and decoding of digital communication signals has been developed.

In the next subsections both the wavelet communication scheme and the software testbed are numerically illustrated. There are two examples presented. The first in Section 7.3.6 is intended to illustrate the main concepts of the modulation and demodulation schemes proposed here; and the second, also presented in Section 7.3.6, is intended to illustrate the full potential of the method.

In each of the examples the wavelet g is fixed as a Morlet wavelet which is depicted in Figure 7.37. This figure shows the general wavelet interface used for specifying wavelets and filter banks. It shows the Morlet wavelet in the time (upper plot) and frequency (lower plot) domains. Note in particular that there is good localization in both domains.

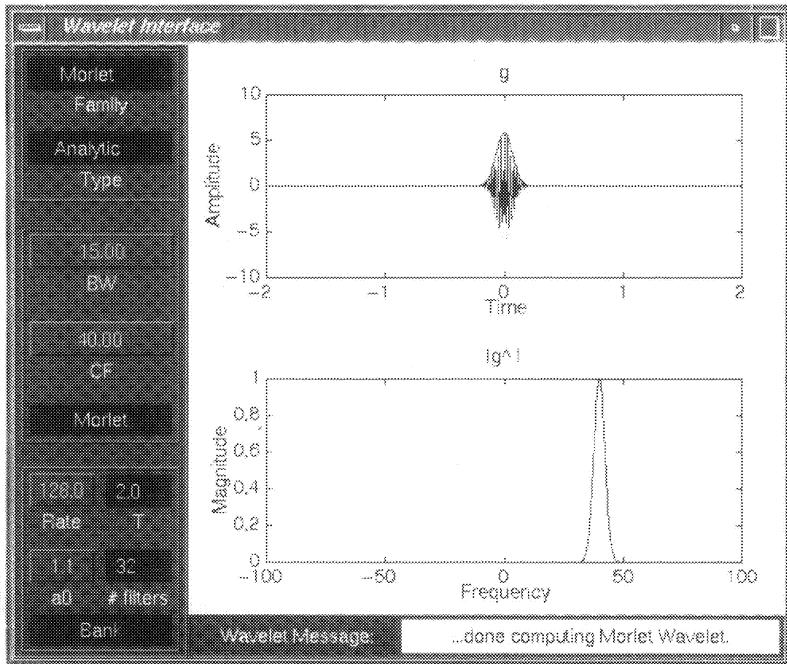


FIGURE 7.37. The Morlet wavelet used in the numerical illustrations of this section.

A Simple Numerical Example

In this example only four wavelet atoms are used. Figure 7.38 displays the main WavSynth interface including the main parameter controls along the left column and a time-frequency box representing the channel. Units of the time and frequency axis may be interpreted as either seconds/Hertz, milliseconds/kiloHertz, or microseconds/megaHertz. A sparse regular grid of points is overlaid in the time-frequency plane which may be used to mouse-select corresponding atoms for inclusion into the underlying atomic set. Once a point is selected the corresponding time-frequency rectangle $\{R_{m,n}\}$ (viz. Equation (7.14)) is displayed.

In the figure, four such points were selected and the resulting set of masks is displayed. Each mask may be subsequently assigned a one or zero by selecting the rectangle center point. This action changes the color of the rectangle to indicate its desired value. This figure shows dark (green) rectangles indicating values of one for all four regions. A light (purple) color indicates a value of zero.

The associated wavelet communication signal is synthesized by depressing the Synth button at the bottom left of the figure. It is subsequently

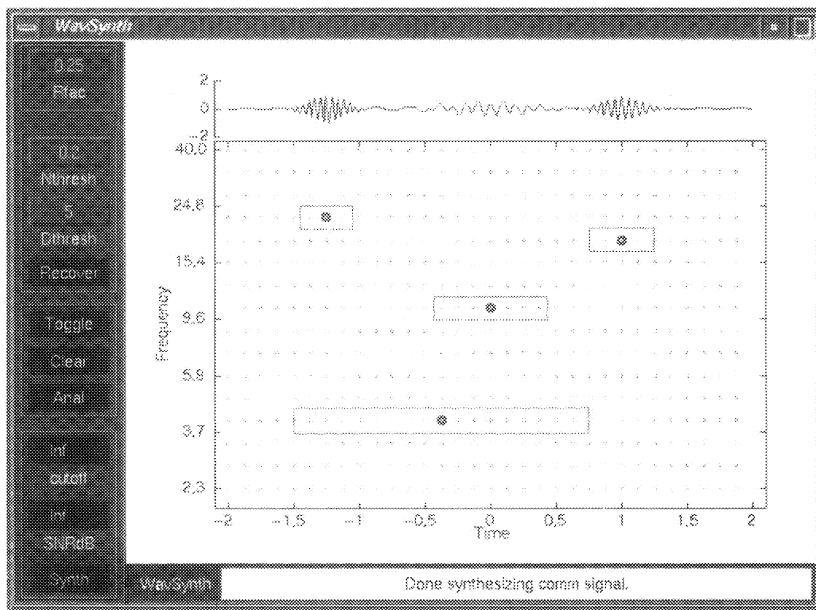


FIGURE 7.38. The wavelet modulation of a simple four-bit carrying communication signal.

displayed above the time–frequency plane. This is the analog signal that is to be transmitted on the channel.

Figure 7.39 illustrates the bit recovery process via Equation (7.15). Overlaid in the time–frequency plane are the wavelet masks together with a color-coded magnitude wavelet transform of the synthesized signal. Darker points indicate a larger wavelet transform magnitude. From the figure, it can easily be seen how the wavelet transform resolves the communication signal into its underlying wavelet components. Depressing the **Recover** button in the GUI causes the energy in each masked region of the wavelet to be computed and displayed in the right-hand side of the rectangle. In this ideal example all the energies are approximately constant at $E \approx 11.6$.

A Denser Numerical Example

This example follows the previous example conceptually quite closely except that the number of wavelet atoms used is 52 instead of 4 in the same time–frequency plane. The 52 time–frequency rectangles associated with the atoms are shown in Figure 7.40 with desired bit values color coded¹⁰

¹⁰Note that in a black and white printout the light (zero) values are only subtly lighter than the dark (one) values.

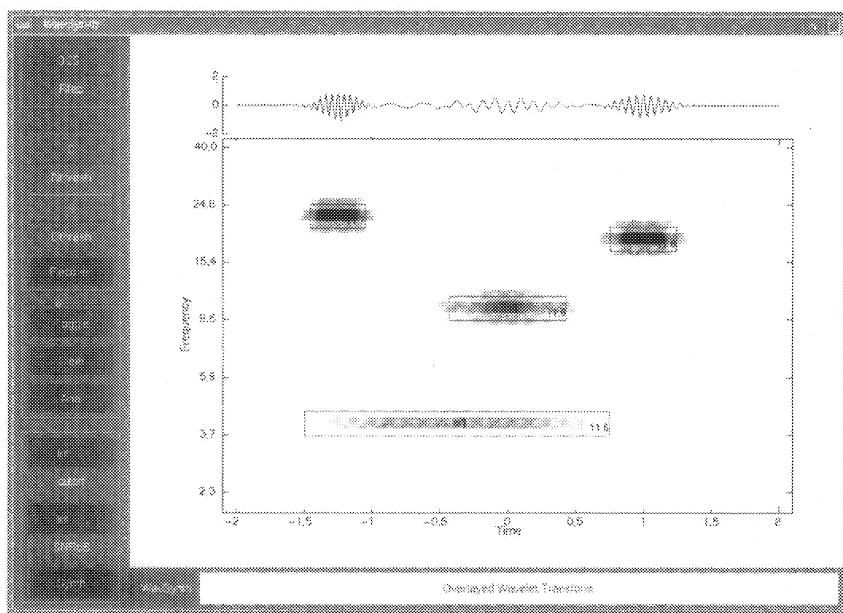


FIGURE 7.39. The corresponding magnitude wavelet transform of the communication signal.

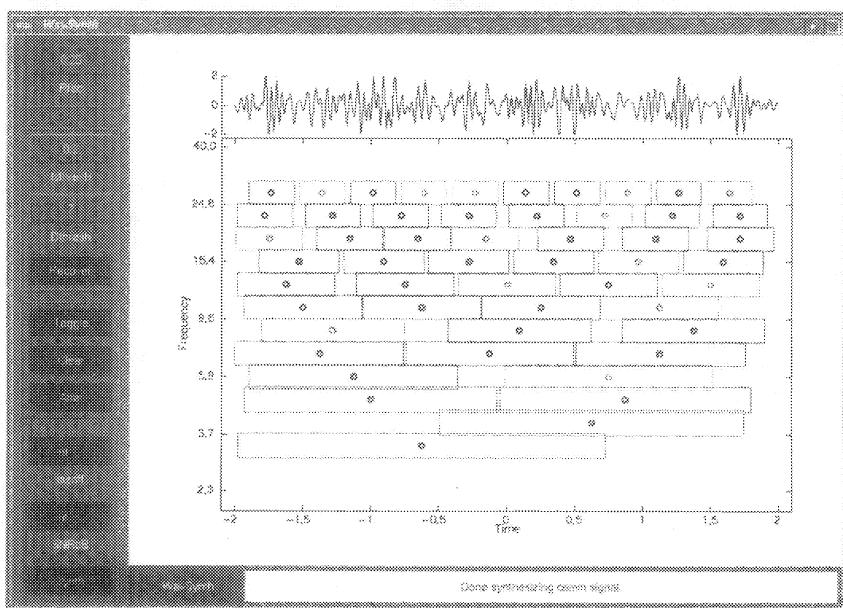


FIGURE 7.40. Modulation of a 52-bit carrying information signal.

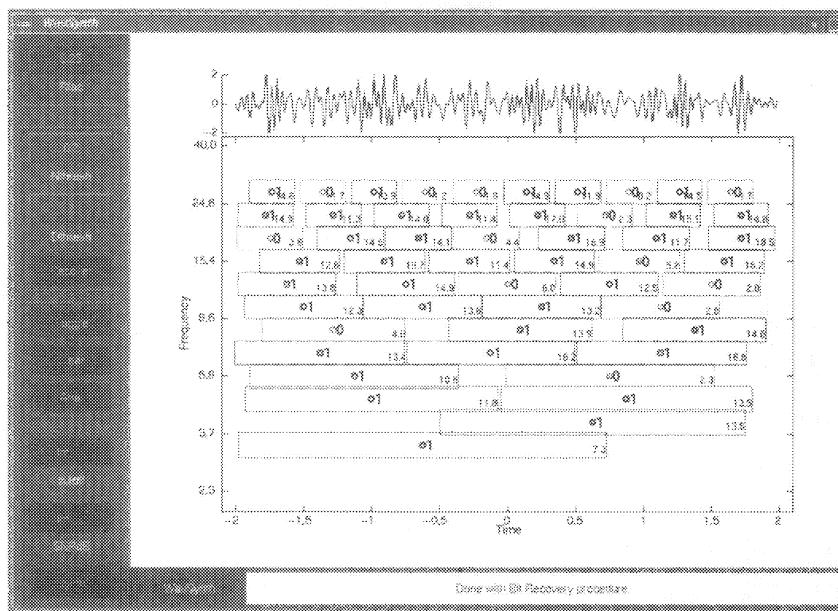


FIGURE 7.41. Bit recovery process for the 52-bit communication signal.

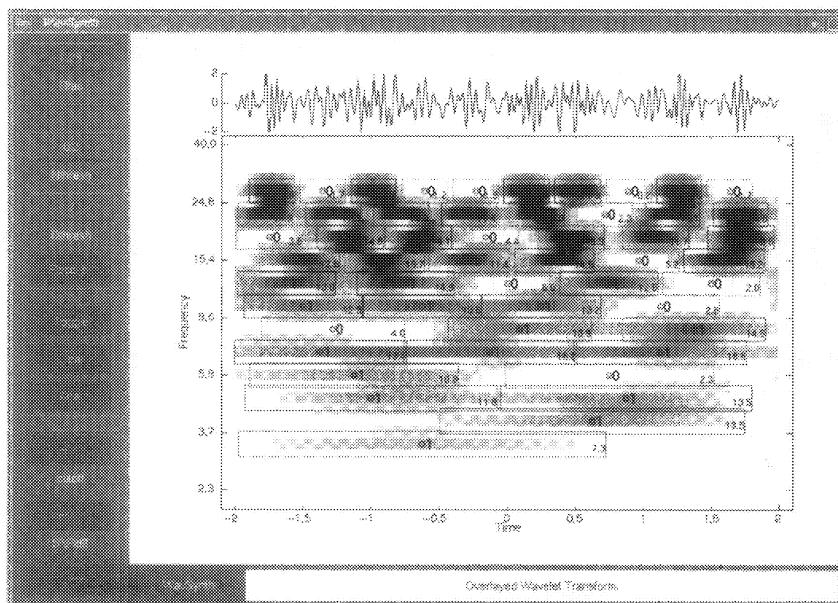


FIGURE 7.42. Magnitude wavelet transform associated with the 52-bit communication signal.

as described previously. In this way, the figure depicts the modulation of a specific bit sequence and its corresponding time domain form.

Figure 7.41 illustrates the bit recovery procedure for this example. Each rectangle is assigned a 0 or 1 as dictated by (7.15) and the specific parameter values as indicated in the figure. Note that in this case the underlying bit vector is recovered perfectly. Energy in the different rectangles is not nearly constant in this case since there is a large degree of overlapping ($R_{\text{fac}} = 0.25$).

For completeness Figure 7.42 shows the corresponding magnitude wavelet transform overlaid on the mask values.

7.3.7 Numerical Experiments

Using the performance measures developed in Section 7.3.4, several numerical experiments have been conducted to determine the overall performance of the wavelet communication scheme described here. In particular, results of numerical simulations to assess the levels of throughput, noise robustness, and security that the scheme possesses are presented.

Background

In order to compute different performance statistics of the proposed wavelet communication scheme, the following reasonable assumptions have been made.

1. Behavior of the scheme is fully determined locally in time and frequency. This allows the analysis to be confined to a simple cluster of cells in the time frequency plane. Subsequent overall performance in the entire time frequency plane may be directly inferred by this local analysis (viz. Figures 7.43 and 7.44).
2. The energy in a given bit cell is only affected by its immediate neighbors. As long as the wavelet atoms are localized enough in time and frequency then this assumption is reasonable. However, this assumption can never be strictly true, yet the effect of nonneighboring cells on a cell's energy can be made negligible.
3. Worst-case scenarios, in terms of bit error, occur when all neighbor cells have the opposite bit value from that of the center cell. In other words, if the center cell is to carry a bit value of zero then the worst situation that can occur with respect to decoding error is that every one of its neighbors has a bit value of one. In this case every neighbor makes some unwanted contribution to the energy of the center cell. A similar argument holds for the case when the center has a value of one and the adjacent cells all have value zero.

Figures 7.43 and 7.44 depict a schematic diagram of a cluster of seven

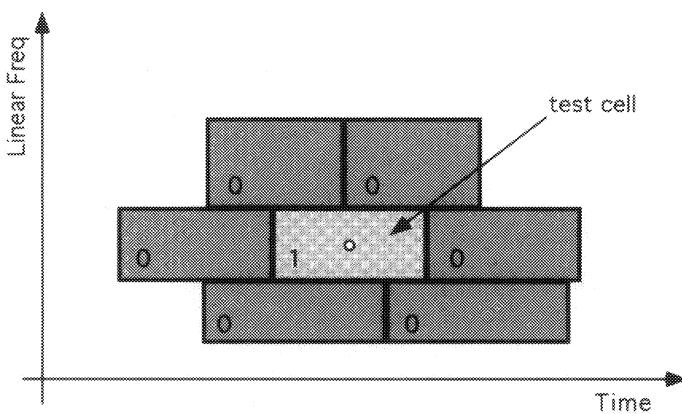


FIGURE 7.43. A 7-bit cell wavelet cluster in the time–frequency domain configured such that the center cell carries a value of 1 and adjacent cells carry values of 0 (i.e., the 1|0 configuration).

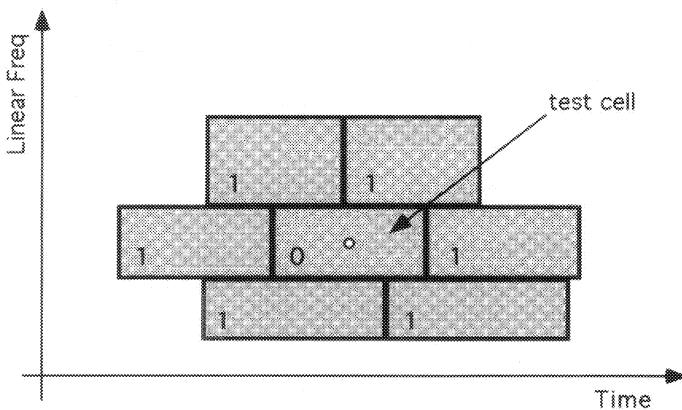


FIGURE 7.44. A 7-bit cell wavelet cluster in the time–frequency domain configured such that the center cell carries a value of 0 and adjacent cells carry values of 1 (i.e., the 0|1 configuration).

cells consisting of one center cell and its six neighbor cells. Respectively, they show the worst scenario configurations of

1. **(1|0)** a center cell of value 1 with adjacent cells of value 0, and
2. **(0|1)** a center cell of value 0 with adjacent cells of value 1.

These two cell geometries have been used in all the numerical simulations conducted. Note that the figure is drawn with a linear scaling on the frequency axis so that the wavelet rectangles have a larger height for higher frequencies. On a log scale, as used in the software (viz. Figure 7.45), the wavelet rectangles have a constant height.

Figure 7.45 shows an example of a larger collection of time–frequency rectangles. The reader may imagine this pattern being extended over the entire time–frequency plane. In such a configuration, it is easy to see that each cell is most closely surrounded by six others. The figure illustrates an instance of a 1|0 configuration. Overlaid on the time frequency plane is the wavelet transform of the corresponding synthesized communication signal (corrupted at the 3dB SNR level).

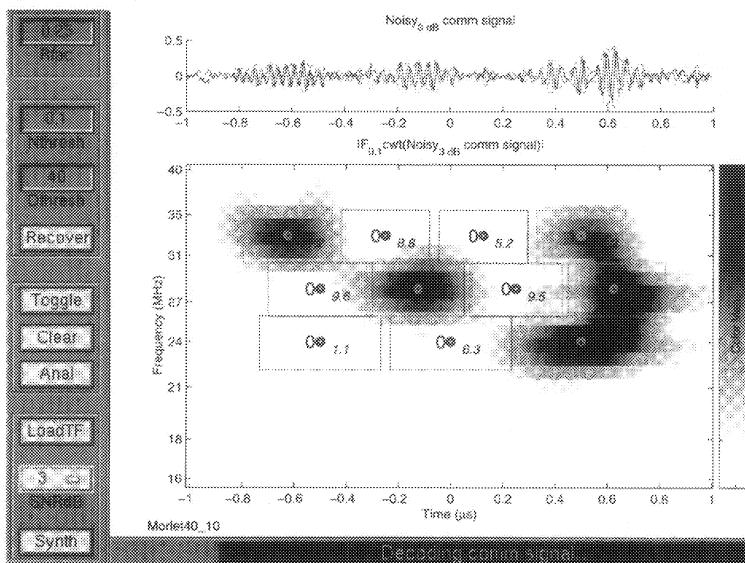


FIGURE 7.45. A more typical arrangement of time–frequency rectangles that contains an example of the 1|0 configuration.

Figures 7.46 through 7.48 show a 0|1 cluster in various lights. Respectively, these figures show

1. the cluster itself and the corresponding analog transmission signal;

2. the cluster with the magnitude of the wavelet transform of the communication signal overlaid; and, finally,
3. the cluster with magnitude of the wavelet transform of a AGWN (SNR = 0dB) corrupted version communication signal overlaid.

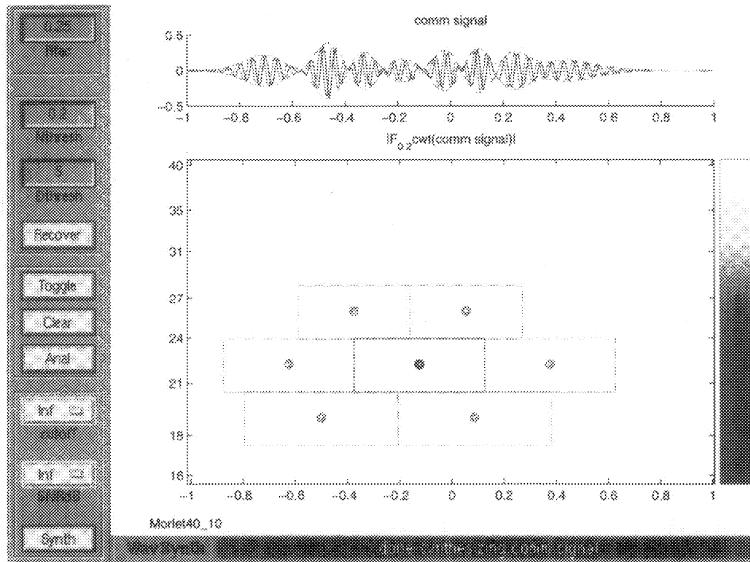


FIGURE 7.46. Wavelet communication interface showing a 0|1 cluster and its corresponding time domain signal.

Decision Function

In Section 7.3.5 the bit recovery procedure based on energy detection is outlined. There, a decision function $d_{\delta_d}(e)$ is formulated that depends on the quantities:

1. the wavelet domain energy e measured in the center cell, and
2. the choice of decision threshold δ_d .

In this setup, e is a random variable that inherits its randomness from the additive noise. The decision function is explicitly defined as

$$d_{\delta_d}(e) = \begin{cases} 1, & e > \delta_d, \\ 0, & \text{else.} \end{cases}, \quad (7.16)$$

where e is the energy measured in the cell of interest in the wavelet domain. Thus, the distribution of the random variable e becomes of great

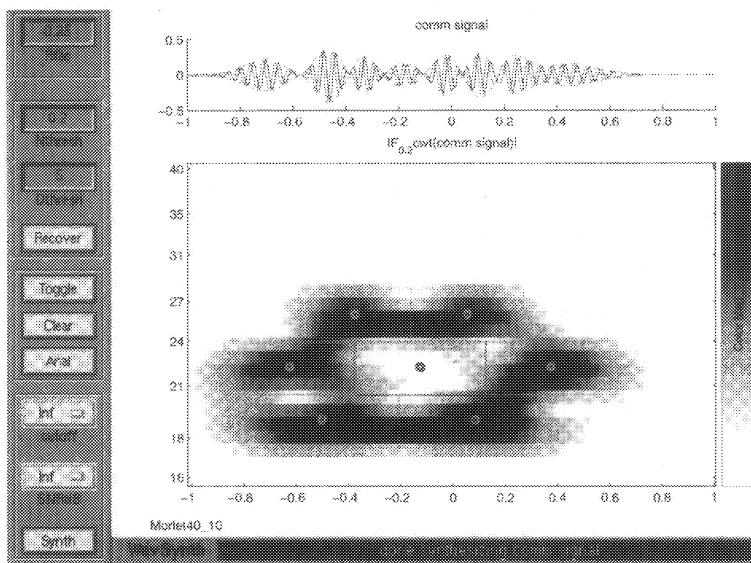


FIGURE 7.47. The 0|1 cluster with the wavelet transform magnitude overlaid.

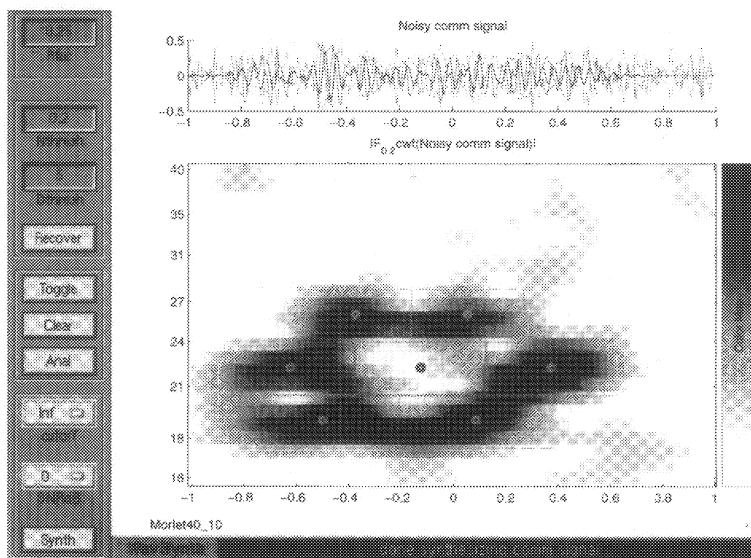


FIGURE 7.48. Noisy version, 0dB SNR, of the 0|1 cluster (overlaid with CWT).

importance with respect to detection performance in the face of noise. In the next subsections this relationship is explored in the context of the decoding scheme of Section 7.3.5.

Noise Robustness

As noise in a channel increases, decoding performance, as measured via the BER, will necessarily degrade. To what extent this degradation occurs is a robustness quality of the overall communication scheme to noise. In this section, two views of the noise robustness associated with the wavelet modulation and demodulation process proposed here are presented. First is the bit error performance versus the level of noise, and second is the measure *receiver operating characteristics* or ROC.

BER versus SNR

Bit error rates versus noise level (SNR, signal-to-noise ratio) are computed via numerical simulation on a large number N of trials. In the numerical experiments reported here $N = 2048$. For a given noise level α and noise detection threshold δ_d , the numerical simulation consists of the following steps.

Numerical Experiment 7.15 (BER versus SNR) Fix a number of trials at N .

1. Compute “analog” signals p_0 and p_1 for the cell arrangements $0|1$ and $1|0$.
2. For each of N trials do the following.
 - (a) Corrupt the signals p_X , $X = 0, 1$, by adding a σ scaled Gaussian distributed random signal w to yield the instance $p_X + \sigma \cdot w$. The value of σ is determined to yield the desired SNR level (viz. Section 7.1.9).
 - (b) Compute the energy in the center cells to yield e_0 and e_1 .
 - (c) Use the decision function of Equation (7.16) to assign a decoded bit value to the center cells based on the values of e_0 and e_1 .
3. Compute histograms and bit error rates.
 - (a) Estimate conditional probability distribution functions
 - i. $p_0(x) \stackrel{\Delta}{=} \Pr\{e_0 = x\}$, and
 - ii. $p_1(x) \stackrel{\Delta}{=} \Pr\{e_1 = x\}$,
 - via histogram binning of the corresponding collections $\{e_{0,n}\}_{n=1}^N$ and $\{e_{1,n}\}_{n=1}^N$ of computed energy values.

(b) Compute the BER based on decisions and true values; that is,

$$BER = Pr\{d_{\delta_n}(e) = 1|0\} \cdot Pr\{0\} + Pr\{d_{\delta_n}(e) = 0|1\} \cdot Pr\{1\} \quad (7.17)$$

Here, $Pr\{0\}$ is the probability that the true center cell carries a 0 and $Pr\{1\}$ is the probability that the true center cell carries a 1. For the preceding experiment these two probabilities are equal to 0.5. The results of this experiment over a range of SNRs and noise thresholds is shown in Figures 7.49 and 7.50. There, the BER is plotted versus 11 different SNRs over 6 different noise threshold values δ_n . The exact values used are listed in Table 7.10.

Parameter	Symbol	Values
SNR (dB)	α	-10, -8, -6, -4, -2, 0, 2, 4, 6, 12, 20
Noise Thresh	δ_n	0, 0.05, 0.10, 0.15, 0.2, 0.3

TABLE 7.10. Parameters for Numerical Experiment 7.15.

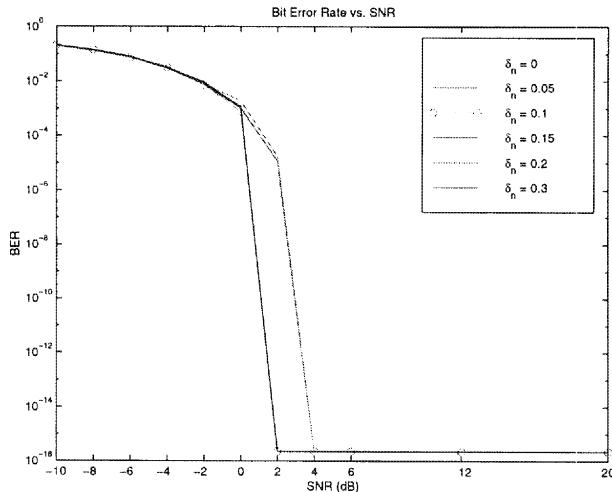


FIGURE 7.49. Full-scale view of BER versus SNR for different noise thresholds.

Figures 7.49 and 7.50 show the BER performance of the wavelet modulation and demodulation scheme for different noise levels and noise thresholds. From the figure, it can be seen that the best performance is given by a noise threshold value of $\delta_n = 0.1$. It shows that user bits are perfectly transmitted (no error) for signal-to-noise ratios as low as 2dB. For smaller

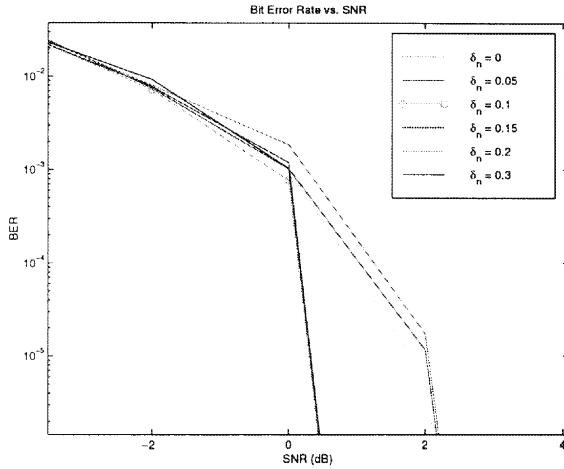


FIGURE 7.50. Zoomed to transition region view of BER versus SNR for different noise thresholds.

SNRs, BER increase from a level of 10^{-3} at 0dB SNR to a level of 10^{-1} at -10dB.

Receiver Operating Characteristics

Receiver operating characteristics (ROC) are a standard measure of detection performance ([Poo88]). Recall that the ROC is a graph of the probability of false alarm versus the probability of detection. A *false alarm* occurs in this case when a bit value of 1 is decoded even though the true value the cell carries is a 0. A *detection* occurs when a bit value of 1 is decoded and the cell is, in fact, carrying a 1. In this way, the ROC quantitatively describes the tradeoffs between the probability of correct decoding versus the probability of incorrect decoding. The associated probabilities of false alarm p_{FA} and detection p_D may be written as

$$p_{FA} = \Pr \{d_{\delta_d} = 1|0\} = \Pr \{d_{\delta_d}(e_0) = 1\} = \Pr \{e_0 \geq x\} = 1 - F_0(x),$$

and

$$p_D = \Pr \{d_{\delta_d} = 1|1\} = \Pr \{d_{\delta_d}(e_1) = 1\} = \Pr \{e_1 \geq x\} = 1 - F_1(x),$$

respectively. Here F_0 and F_1 are the cumulative distribution functions associated with the random variables e_0 and e_1 ; that is,

$$F_0(x) = \int_{-\infty}^x p_0(y)dy$$

and

$$F_1(x) = \int_{-\infty}^x p_1(y)dy.$$

Figure 7.51 depicts the ROC curves for SNRs less than or equal to 2dB. A

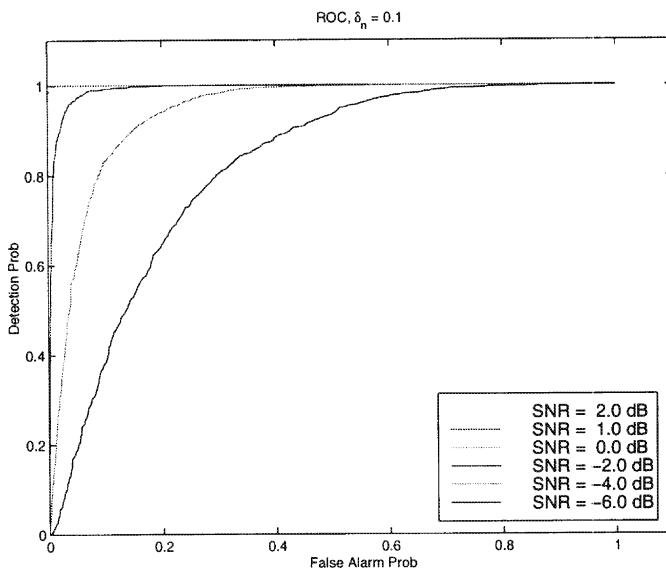


FIGURE 7.51. Receiver operating characteristics for the wavelet communication scheme.

ROC curve is generated by systematically varying the detection threshold δ_d in the decoding scheme and providing estimates for p_{FA} and p_D . As shown in the previous section, at an SNR greater than or equal to 2dB there are no errors; that is, the BER is 0. This is reflected in the ROC by attaining the point $(p_{FA}, p_D) = (0, 1)$. For SNRs less than 2dB, BERs are greater than zero and the corresponding ROCs do not pass through the optimal point $(0, 1)$. In light of this, the point on a ROC curve that is closest to $(0, 1)$ may be considered the optimal operating point and, in turn, this point gives an optimal value for the detection threshold δ_d .

Measured Energy Distributions

As the noise level varies, the distribution of energy in the center cell of a test cluster is necessarily affected. At very low levels, the energy distributions will have small variance and cluster around their mean. The separability between the energy associated with a 0|1 cluster and the energy associated with a 1|0 cluster is directly related to the BER. In the case where the two distributions are well separated and do not overlap, perfect decoding

performance is clearly possible simply by taking a detection threshold δ_d equal to any value in the gap. Once the distributions overlap some error in bit decoding is unavoidable.

For different values of SNR, Figures 7.52 through 7.54 display the energy distributions for the two cases of a 0|1 cluster and a 1|0 cluster. These correspond to histograms of the measured energy values e_0 and e_1 , respectively. Figure 7.52 shows the distributions in relatively low noise levels of SNRs of 12 (left) and 6dB (right). Here the two distributions are well separated. This is also the case in Figure 7.53 (left) where the SNR is 2dB; however, on the right side of the same figure, the distributions with a SNR of 0dB are seen to have some overlap. Hence there is a nonzero level of bit error at 0dB. Further overlap is shown in Figure 7.54 where SNRs of -2dB and -6dB are used.

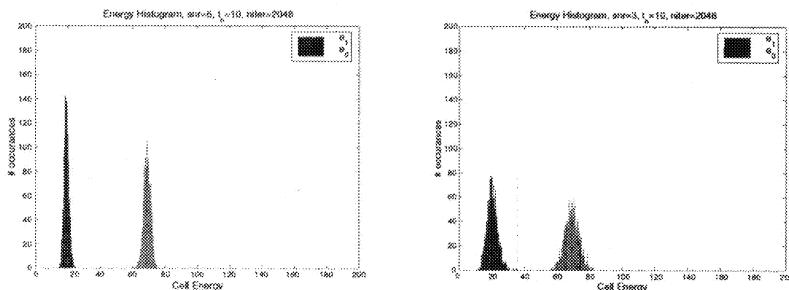


FIGURE 7.52. Histogram of energy distributions for the center test cell and the surrounding cells: (left) SNR = 12dB (right) SNR = 6dB.

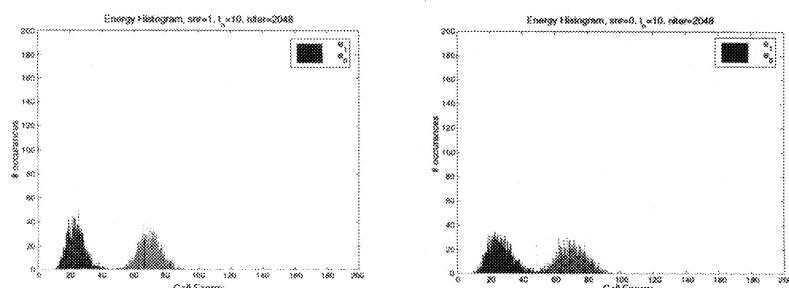


FIGURE 7.53. Histogram of energy distributions for the center test cell and the surrounding cells: (left) SNR = 2dB (right) SNR = 0dB.

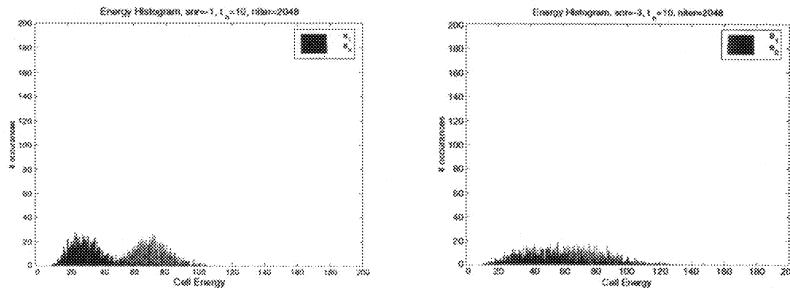


FIGURE 7.54. Histogram of energy distributions for the center test cell and the surrounding cells: (left) SNR = -2dB (right) SNR = -6dB.

Throughput

To yield higher throughput (i.e., a larger number of bits per second per unit bandwidth (spectral efficiency)), one may try to pack more atoms into the same time–frequency area defined by the channel. Higher throughput will most likely come at the cost of a higher BER. The pertinent measure of throughput here is the spectral efficiency discussed in Section 7.3.4.

The spectral efficiency of the wavelet communication scheme is controlled directly by the time–frequency extent of the analyzing wavelet chosen together with the choice of the essential support factor R_{fac} discussed in Section 7.3.5 on page 234. A larger value of R_{fac} leads to smaller effective rectangles and hence allows denser packing of the wavelet atoms.

To make the idea of the time–frequency extent more precise the notion of the ϵ -time-bandwidth product of a function is now introduced. To do this the auxiliary concept of ϵ -support is first defined. Let the ϵ -support of a function g be defined by

$$\text{supp}_\epsilon g \triangleq \{t : |g(t)| \geq \epsilon \cdot \|g\|_\infty\}.$$

The epsilon support is the set of domain values at which a function has significant magnitude. The level of significance is relative to the value of ϵ . Now, define the ϵ -time-bandwidth product $Q_\epsilon(g)$ of a function g as

$$Q_\epsilon(g) \triangleq \mu(\text{supp}_\epsilon g) \cdot \mu(\text{supp}_\epsilon \widehat{g}),$$

where μ is a measure of length.

For the Morlet wavelet used in most of the numerical simulations here the $\frac{1}{4}$ -time-bandwidth product is

$$Q_{1/4}(\text{Morlet40_10}) = 1.77.$$

The corresponding spectral efficiency is

$$\nu_s = 1/1.77 = 0.57.$$

For comparison, the spectral efficiencies for conventional techniques such as phase shift keying (PSK) and quadrature amplitude modulation (QAM) are in the range of 2 to 7 ([Gib93, p. 227]). Thus, in terms of throughput, the simple energy detection decoding discussed here performs poorly. This is not too surprising since all phase information in the scheme is essentially neglected. Taking into account phase information in the wavelet communication scheme should increase spectral efficiency into the competitive range.

Security

Following the approach outlined in Section 7.3.4, security of transmission may be evaluated by attempting to decode a transmission signal without explicit knowledge of the key decoder parameters. The decoder parameters are

1. the analyzing wavelet, and
2. the time frequency cell template, that is, the rectangles $\{R_{m,n}\}$ in Equation (7.14).

Without any knowledge of these parameters decoding of the transmission signal is very unlikely. To substantiate this claim the following numerical experiment has been conducted.

Consider a numerical simulation to evaluate the level of security inherent in the transmission. As a very conservative estimate of the security of the transmission scheme, it is assumed that

1. it is known that the signal is wavelet-modulated as given by Equation (7.9);
2. it is known that the decoder has the cell structure form of Equation (7.15); and
3. the frequency range of the wavelet filter bank is known.

In the numerical simulation the decoder uses various analyzing wavelets that vary in similarity with the actual synthesis wavelet. Even given the considerable knowledge of the decoding process listed previously, the decoding performance as measured via the BER is poor in all cases in which the decoding filter bank differs even slightly from the analysis bank. This is illustrated in Table 7.11. The first entry in this table is the synthesis wavelet which gives, as expected, zero error. As the analysis bank gets farther away from the synthesis bank the BER increases dramatically.

Table 7.11 enumerates the BER performance of the decoding process when the analysis filter bank is the Morlet ($a_0 = 1.04$, $BW = 10$, and $CF = 40$) filter bank and the analysis bank is varied as indicated. BER levels near 0.5 are indistinguishable from pure white noise and indicate

<i>Wavelet</i>	<i>a</i> ₀	<i>BW</i>	<i>CF</i>	<i>aux</i>	<i>BER@6dB SNR</i>
Morlet	1.04	10	40	-	0
	1.04	20	40	-	0.02
	1.04	30	40	-	0.06
	1.04	40	40	-	0.10
	1.06	10	40	-	0.00
	1.10	10	40	-	0.54
	1.04	10	30	-	0.02
	1.04	10	20	-	0.50
	1.04	10	30	2	0.50
BSpline	1.04	10	30	4	0.50
	1.04	10	30	6	0.50
	1.04	10	40	2	0.19
PBL	1.04	10	40	4	0.50
	1.04	10	40	6	0.50
	1.04	10	40	6	0.50

TABLE 7.11. Decoding stability with respect to uncertainty of synthesis filter bank and *full* knowledge of cell template.

a complete void of information. Even BER levels of 0.1 are considered very poor in any communication system. As indicated in this table, even with considerable knowledge of the encoding and decoding processes, the successful decoding of an intercepted message without explicit knowledge of the analysis filter bank seems highly unlikely.

7.3.8 Remarks

A wavelet-based scheme has been proposed for the coding and decoding of digital information for transmission through fundamentally analog channels described by a time–frequency rectangle. The scheme is novel in that it uses nonorthogonal wavelet functions as atoms to construct signals for transmission. It has been demonstrated that the scheme is highly robust to noise (AGWN) with undetectable (zero) error rates in noise levels as high as 2dB SNR. Moreover, it has been seen that the scheme has an inherent level of security that makes the decoding process ineffectual without knowledge of the time–frequency cell templates and/or knowledge of the synthesis wavelet.

Throughput

In the simple energy detection scheme for the bit decoding process (viz. Section 7.3.5) all phase information is essentially being neglected. By increasing the complexity of the detector to also consider phase information, the effective throughput of the overall scheme should be able to be markedly increased. Using this strategy, it is not unreasonable to expect an increase

in throughput of two to four times without significantly affecting the bit error rate.

Security

Although low probability of intercept has been demonstrated using the proposed scheme, it places all the energy due to a specific bit in a well-localized time-frequency rectangle due to the choice of underlying wavelet atoms. An alternative is to choose wavelet families that are *not* well localized in time and frequency, but rather distribute their energy throughout the time frequency plane. In this way the bits are *spread* through the entire spectrum adding an even greater degree of security to the overall communication scheme.

7.4 Identification

A signal's frequency-modulation may be estimated through simple coherent identification schemes in the OCWT domain, for example, thresholding. Identification may then be subsequently performed via a simple nearest neighbor thresholded classifier using a specified metric (notion of distance). This approach is applied to a small test set of monocomponent and multi-component synthetic FM signals and shown to yield 100% identification success at signal-to-noise ratios greater than -8dB using a Morlet-based OCWT. For comparison, the same data set yields 100% identification success for signal-to-noise ratios only as low as 0dB when comparing signals directly in the time domain, that is, via a matched filter technique. A time domain comparison of this sort is equivalent to comparisons with orthonormal representations in the L^2 -norm.

7.4.1 Approach

Figure 7.55 displays a block diagram of the identification process. It consists of two main operations of representation W , and matching (metric) $\rho : W(\mathcal{H}) \times W(\mathcal{H}) \mapsto (0, \infty)$. In the following discussion, only these distinct pairs (W, ρ) are considered:

1. **(Matched Filter)** $W = I$, the identity operator, and $\rho_{\text{MF}}(p, q) = \|p - q\|$, the Euclidean distance; and
2. **(OCWT Filter)** $W = W_g|_{\Gamma(\Delta, a_0)}$ (viz. (6.2) and (6.5)), and

$$\begin{aligned} \rho_{\text{OCWT}}(p, q) &= \|W(p - q) \cdot M_\sigma(Wp) \cdot M_\sigma(Wq)\|_W \\ &= \left(\int_{\mathbb{R}} \int_{s>0} |W(p - q) \cdot M_\sigma(Wp) \cdot M_\sigma(Wq)|^2 ds dt \right)^{1/2} \end{aligned} \quad (7.18)$$

Here M_σ is a wavelet domain threshold mask parameterized by a threshold σ defined as

$$(M_\sigma W p)(t, s) \triangleq \begin{cases} \alpha, & |W p(t, s)| > \sigma, \\ 0, & \text{else,} \end{cases}$$

where α is chosen so that $\|M_\sigma W p\| = 1$.

Let $\mathcal{P} = \{p_k\}$ be the set of analytic¹¹ FM signals to be identified. It is assumed that a database $\{W p_k\}_{k=1}^N$ of clean versions of signals to be identified is available. This deviates from a realistic scenario in at least two important respects: in a real system the size of the database would grow with observations deemed outside the database; and, elements in the database would not have an infinite signal-to-noise ratio.

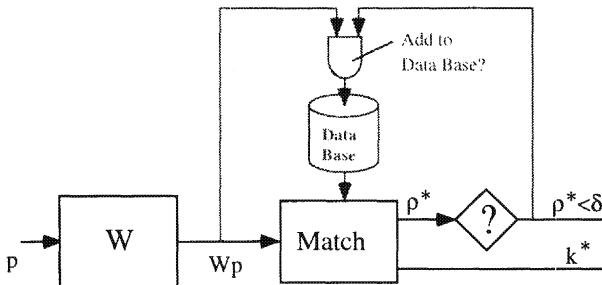


FIGURE 7.55. Block diagram of the identification process with a provision for determining whether an observed signal is outside the database.

Suppose the database currently contains J distinct classes. The database consists of a tagged collection of signal representations

$$\{W p_{j,k}\}, j = 1, 2, \dots, J, k = 1, 2, \dots, N_j,$$

where k is a class index, j is a signal index, and N_j is the number of signals in the j th class. Given an observed noisy signal p , the identification process is tasked to determine if the observation p is represented in the database, and if so to say which it is. In the case that it is not represented then the signal p is deemed to be of a *new* class. New classes may be flagged as such and subsequently added to the database as the $J + 1$ th class (provided the signal-to-noise ratio is sufficiently high).

To explain the identification process more precisely, let p be an observed noisy signal and (W, ρ) be a specified representation/metric pair. The closest candidate index k^* (i.e., the index of the representation in the database

¹¹The term analytic is used in the sense that the Fourier transform vanishes off positive frequencies.

that is closest to the observed representation in the sense of ρ) is

$$k^* \triangleq \underset{k}{\operatorname{argmin}} \rho(\operatorname{avg}(Wp_{k,\cdot}), Wp),$$

where avg is an averaging operator. To determine if this candidate is indeed the signal's class a threshold-based decision function may be formulated. Such a decision function is specified with a *closeness* threshold δ for which candidates with distances greater than the threshold are deemed outside the database. More precisely, we define a decision function d_δ as

$$d_\delta(Wp) = \begin{cases} k^*, & \rho(x_{k^*}, Wp) < \delta, \\ \text{NEW}, & \text{else.} \end{cases} \quad (7.19)$$

If all members from the same class generate sufficiently close representations and members from different classes generate sufficiently separated representations then this decision will provide perfect identification.

7.4.2 Performance Indicators

Given the preceding identification process it is desirable to formulate some useful performance indicators in order to compare the relative merits of different choices of representation W and matching metric ρ . The main interest here is the degradation of performance of the identification process as the signal-to-noise ratio of the observed signals falls toward $-\infty$.

To this end, two (database-dependent) conditional distance indicators have been developed. The first, ρ_{same} , measures the maximum distance between elements in a database and their noisy observations. The second, ρ_{diff} , measures the minimum distances between elements and the noisy observations of all others. These two indicators are given precisely as

$$\begin{aligned} \rho_{\text{same}} &\triangleq \max_j \max_{k, k'=1, 2, \dots, N_j} \rho(Wp_{k,j}, Wp_{k',j}) \\ \rho_{\text{diff}} &\triangleq \min_{j \neq j'} \min_{\substack{k=1, 2, \dots, N_j \\ k'=1, 2, \dots, N_{j'}}} \rho(Wp_{k,j}, Wp_{k',j'}) \end{aligned}$$

where indices are taken over the whole database. These indicators are formed with the idea that the threshold decision d_δ of Equation (7.19) will provide 100% correct decisions (on a given data set) should it be possible to choose a threshold δ such that $\rho_{\text{same}} < \delta < \rho_{\text{diff}}$.

7.4.3 Numerical Experiment

An experiment to assess the quality of the identification process is given as Numerical Experiment 7.16. For this purpose, the performance indicators of the previous section are used to provide a graph of identification performance as a function of the signal-to-noise ratio (SNR).

Numerical Experiment 7.16 (FM Identification) Specify a signal set $\mathcal{F} = \{f_k\}$ and a sequence $\{SNR_n\}$ of desired signal-to-noise ratios.

1. For each SNR, generate a set of noisy observations $\{f_k + \sigma \cdot w\}$ where w is a normalized Gaussian white noise process and σ is chosen so that the desired level of SNR is attained (viz. Section 7.1.9).
2. For each noisy data set, perform the identification and compute the indicators ρ_{same} and ρ_{diff} . This provides two curves as a function of SNR $\rho_{\text{same}}(\text{SNR})$ and $\rho_{\text{diff}}(\text{SNR})$.
3. Determine the smallest SNR and a value for the threshold δ that provides 100% identification via

$$\rho_{\text{same}}(\text{SNR}) < \delta < \rho_{\text{diff}}(\text{SNR}).$$

This process yields the best classifier threshold δ to use for a given set of data in the sense of maximum noise robustness. Figure 7.56 displays examples of these curves computed for the synthetic FM data set described in the next section using the parameters listed in Table 7.12.

Parameter	Symbol	Values
SNR (dB)	α	-16, -8, 0, 8
Dilation Const.	a_0	1.03
# Filters	n_f	64
Wavelet	g	$g_{\text{Morlet,CF}=50,\text{BW}=10}$
Sample Period	Δ	1/128
Interval	T	4

TABLE 7.12. Parameters for Numerical Experiment 7.16.

A set of six pulsed signals $\{p_k\}_{k=1}^6$ with various frequency-modulations

$$f_{k,l}, \quad k = 1, 2, \dots, 6, \quad l = 1, 2, \dots, L_k,$$

where L_k is the number of modulated components in the k th signal, has been synthesized to form a small data set for identification.

As a small data set for identification the synthetic FM data set discussed in Section 6.4 is used. The modulations associated with this data set are shown in Figure 6.6.

To demonstrate the approach only a single Morlet-based OCWT is considered. Figure 6.7 shows the associated filter bank used to compute this OCWT and its corresponding frequency support function G of Equation (6.3). Recall that this filter bank is described by Equation (4.1) with filter

parameters center frequency $\gamma_c = 40$, $\gamma_b = 5$, and filter bank parameters $s_m = 1.03^m$, $m = 1, 2, \dots, 64$.

Figure 7.56 displays the performance indicators as a function of signal-to-noise ratio for ρ_{same} and ρ_{diff} for the matched filter and OCWT identification schemes described here. Also displayed in the figure as a horizontal line is the choice of threshold which yields the optimal noise robust identification performance. The figure clearly shows that in the case of the OCWT identification scheme, choosing a classification threshold of $\delta \approx 2.1$ will yield 100% identification performance on the synthetic test data set for SNRs as low as -8dB. In contrast, the figure also shows that in order for the matched filter identification scheme to yield 100% identification accuracy the SNR must be 0dB or greater.

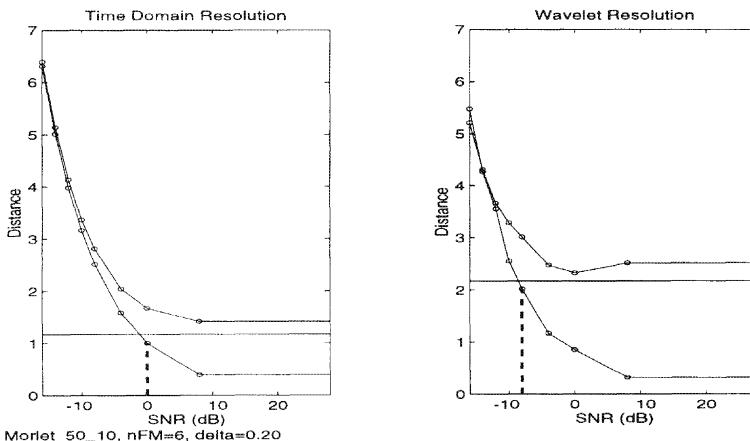


FIGURE 7.56. Performance indicators for (left) the matched filter and (right) the OCWT identification schemes.

Although this numerical experiment is far from comprehensive, it hints strongly at the potential benefits that overcomplete wavelet representations have to offer in the realm of identification. To give the reader a feel for the relative distances involved using the different representation/metrics, Tables 7.15 and 7.16 show the $\text{SNR} = -8\text{dB}$ match matrices for the OCWT and matched filter identification schemes, respectively. The same is shown in Tables 7.13 and 7.14 for the case of $\text{SNR} = 0\text{dB}$. Note that since the distance matrices are symmetric only values in the upper triangular portion of the matrix are computed.

	FM1	FM2	FM3	FM4	FM5	FM6
FM1	0.853	3.095	2.969	3.160	3.294	3.159
FM2	-	0.799	3.254	2.321	2.617	2.743
FM3	-	-	0.724	2.800	2.541	2.768
FM4	-	-	-	0.690	2.533	2.498
FM5	-	-	-	-	0.746	2.778
FM6	-	-	-	-	-	0.702

TABLE 7.13. Morlet OCWT match distances for SNR = 0dB.

	FM1	FM2	FM3	FM4	FM5	FM6
FM1	1.000	1.672	1.718	1.739	1.718	1.719
FM2	-	1.000	1.819	1.695	1.722	1.697
FM3	-	-	1.000	1.716	1.669	1.694
FM4	-	-	-	1.000	1.693	1.695
FM5	-	-	-	-	1.000	1.736
FM6	-	-	-	-	-	1.000

TABLE 7.14. Time domain match distances for SNR = 0dB.

	FM1	FM2	FM3	FM4	FM5	FM6
FM1	1.784	3.374	3.319	3.403	3.845	3.878
FM2	-	1.734	3.443	3.501	3.118	3.106
FM3	-	-	1.787	3.264	3.278	3.287
FM4	-	-	-	1.693	3.107	3.301
FM5	-	-	-	-	1.896	3.012
FM6	-	-	-	-	-	2.017

TABLE 7.15. Morlet OCWT match distances for SNR = -8dB.

	FM1	FM2	FM3	FM4	FM5	FM6
FM1	2.512	2.809	2.839	2.857	2.923	2.903
FM2	-	2.512	2.929	2.920	2.870	2.836
FM3	-	-	2.512	2.898	2.881	2.849
FM4	-	-	-	2.512	2.862	2.879
FM5	-	-	-	-	2.512	2.885
FM6	-	-	-	-	-	2.512

TABLE 7.16. Matched filter match distances for SNR = -8dB.

7.4.4 Remarks

A general framework for the identification of FM signals has been presented that consists of the three main elements of representation, metric (distance) specification, and nearest neighbor thresholded classifier. In addition, we have specified two key performance indicators that yield best values of classifier threshold and noise tolerance for a given data set. This framework is applied to two distinct cases in which the representation is the signal itself with a Euclidean metric; and the representation is a Morlet-based OCWT on a dense regular lattice in the time–frequency plane with a doubly masked wavelet domain distance metric (Equation (7.18)).

When applied to a small test set of single component and multiple component synthetic FM signals the OCWT-based representation has been shown to provide a -8dB SNR performance benefit over the direct (matched filter). Specifically, the OCWT-based identification scheme is shown to yield 100% identification success at signal-to-noise ratios greater than -8dB whereas the direct scheme yields 100% identification success for signal-to-noise ratios only as low as 0dB.

7.5 Conclusion

Several different applications of overcomplete wavelet transforms have been discussed in this chapter:

1. noise suppression,
2. signal compression,
3. digital communication, and
4. signal identification.

Although diverse in scope, all of these applications are linked via the fundamental concept of signal coherence with respect to an atomic set, and specifically, to wavelet atomic sets. A quantitative theory of coherence is developed in Section 7.1.4, and used explicitly in applications of noise suppression and compression, and implicitly in applications of identification and digital communications. To various extents, each application demonstrates the idea that there are performance benefits (c.f., Section 6.1.4), to be gained from the use of overcomplete wavelet representations.

Because the numerical experiments outlined here have a large number of free parameters, the limited choices demonstrated can serve only as a small indication of potential. The schemes outlined here are a first attempt toward that end. In and of themselves, the particular approaches outlined require further development to be truly useful. If nothing else, the main point of this chapter has been to illustrate the idea that overcompleteness is *not* necessarily a bad thing.

Problems

- 7.1 Let F_{δ, c_*} be the threshold operator of Equation (7.2) for some $c_* \in \ell^2(\mathbb{Z})$. If L is an overcomplete frame representation for \mathcal{H} show that even if $c \in \mathcal{H}$, $F_\delta c$ is not, in general, in $L(\mathcal{H})$. (Let $\mathcal{H} = \mathbb{R}^2$.)
- 7.2 Let $\{\phi_n\}$ be a frame for \mathcal{H} with frame correlation R which is diagonal. Show that if $f \in \mathcal{H}$ is coherent with respect to $\{\phi_n\}$ then f may be constructed from relatively few members of $\{\phi_n\}$.
- 7.3 Repeat Problem 7.2 in the case where R is block diagonal.
- 7.4 Prove Fact 7.3.
- 7.5 For a fixed value of δ and a given frame for \mathcal{H} , show that the coherence distribution of Definition 7.2 (page 179) satisfies

$$\text{Coh}_\delta(af) = \text{Coh}_{\frac{\delta}{|a|}} f$$

but NOT

$$\text{Coh}_\delta(f + g) = \text{Coh}_\delta f + \text{Coh}_\delta g$$

for all $f, g \in \mathcal{H}$ and $a \in \mathbb{C}$.

- 7.6 For a given frame for \mathcal{H} with representation L , show that the coherence measure of Definition 7.4 (page 181) satisfies

$$\text{COH}(af) = |a|\text{COH}(f)$$

and

$$\text{COH}(f) \leq \|Lf\|$$

for all $f \in \mathcal{H}$ and $a \in \mathbb{C}$.

- 7.7 Prove Corollary 7.9.

8

Object-Oriented Wavelet Analysis with MATLAB 5

All of the numerical processing and examples presented in this book have been computed using a suite of object-oriented tools developed in MATLAB 5. This set of tools has been called collectively the Wavelet Signal Processing Workstation (WSPW) and is described generally in Section 8.1. Up-to-date code and detailed documentation is available online at

www.birkhauser.com/book/ISBN/0-8176-3909-8.

The Wavelet Signal Processing Workstation facilitates the processing of signals using overcomplete and discrete wavelet transformations. It runs under MATLAB versions 5.0 and later on all MATLAB supported platforms and operating systems. This includes PCs running Windows95 and almost all UNIX based machines including LINUX.

MATLAB is a powerful and well-supported processing software package for technical computing that runs on a wide variety of computational platforms and operating systems. In its latest release, MATLAB 5.X offers significantly advanced features with respect to both functionality and coding efficiency. One major feature, in particular, is the support of object-oriented programming. Object-oriented code has an inherent level of extensibility and structure previously unattainable in earlier versions of the MATLAB language. These new features have been exploited to develop powerful object oriented tools for wavelet signal processing.

This chapter describes both the high-level user interfaces that provide a front end to the main wavelet processing routines and many of the key routines themselves. Together, these routines form the *Wavelet Signal Processing Workstation*. Since they are likely to change in future versions, the fine details of operation of the workstation are available via the World Wide Web.

There are two main object classes that have been developed for wavelet signal processing: a fundamental `sampled_signal` class and a `wavelet` class. The `sampled_signal` class provides a common setting for the processing of uniformly sampled signals with varying sample rates over varying interval lengths. Using this class of signals as a starting point, the `wavelet` class facilitates the wavelet filter bank processing of sampled signals; and,

hence, facilitates the implementation of the OCWT discussed in Chapter 6.

A good knowledge of basic MATLAB functionality is assumed in the presentation. In addition, some basic familiarity with MATLAB 5 advanced features including structures, cell arrays, objects, and methods, is helpful. For a discussion of the advanced features of version 5 the reader is referred to [Mat97].

8.1 Wavelet Signal Processing Workstation

The Wavelet Signal Processing Workstation is a set of MATLAB files for the analysis of 1- and 2-dimensional signals using overcomplete ("continuous") and discrete wavelet transformations. Running in MATLAB versions 5.0 and later, the WSPW comes with comprehensive graphical interfaces.

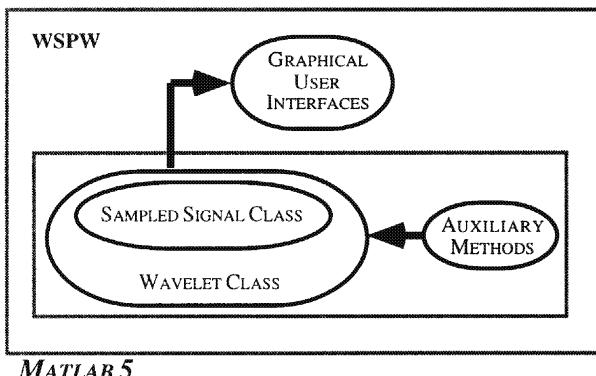


FIGURE 8.1. Architecture of the Wavelet Signal Processing Workstation.

Figure 8.1 depicts the high-level anatomy of the WSPW. The WSPW may be utilized on different levels:

1. graphical user interface, and
2. MATLAB file or command line.

Of course, flexibility in functionality (or speed) and ease of use are traded off between these two levels. At the highest level, the graphical user interface level, the user is constrained¹ by the functionality of the interface whereas

¹In the free demo distribution the user receives MATLAB preparsed binary p-code with a limited functionality. This code is unreadable to the user. A fully functional ASCII version, allowing the user to read the processing code, is also available for purchase.

at the lower MATLAB level users are free to use the workstation methods in any desirable manner.

Typing `WSPW`² at the MATLAB prompt invokes³ the main workstation graphical interface and displays the messages shown in the following:

```
>> WSPW
Switching to version 4 color scheme

Welcome to the 'Wavelet Signal Processing Workstation Version 1.0'

(c) Copyright 1997
      AIMS, Inc.

Inquiries and/or bug reports may be sent to:

tonyt@palindrome.nrl.navy.mil
```

Figure 8.2 shows the initial WSPW graphical user interface. It consists of a large white area for graphical output and different mouse-driven controls. Along the bottom of the window is a message box where processing actions and errors of the WSPW are reported. Active controls are placed along the left edge of the window in three main panels that control the following items (from bottom to top).

1. **Signal Specification:** This panel is dependent on the data type specified in the Data Type menu. For the `sig` type, as specified in the figure, it consists of the controls:
 - (a) **Make** button: synthesizes and displays the specified signal.;
 - (b) **Signal name** menu: a menu of predefined signals (viz. the MATLAB method `sampled_signal` for an overview of signal synthesis); and
 - (c) **SNRdB** menu: specifies a desired noise level with which to corrupt a synthesized signal (viz. the MATLAB method `add_noise`).
2. **Transform Specification:** This panel selects a transformation process and initiates the transform process. It consists of the controls:
 - (a) **Xform** button: transforms and displays the transform of the specified signal in a new window;

²This assumes that the WSPW software is resident on the machine that is running MATLAB and that the path is set properly.

³Note that the first action of the workstation is to revert the color scheme back to the original version 4 default scheme. However, the workstation routines will operate perfectly well using the version 5 color scheme but the resulting plots are not nearly as aesthetically pleasing in the author's opinion.

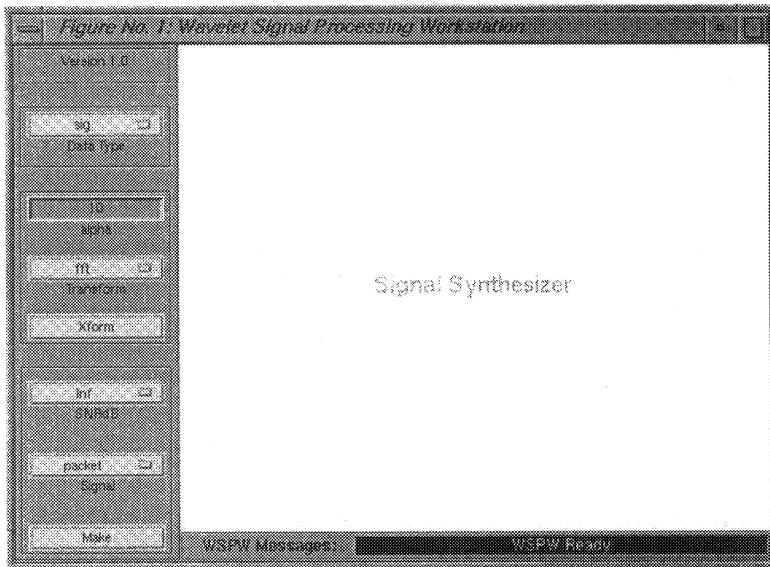


FIGURE 8.2. WSPW startup screen.

- (b) **transform name** menu: a menu of predefined transforms. Possible transforms include
 - i. **fft**: fast Fourier transform;
 - ii. **fwt**: fast (discrete) wavelet transform; and,
 - iii. **cwt**: overcomplete wavelet transform; (selection of this transform spawns the wavelet filter bank specification interface if one is not already present); and
 - (c) **SNRdB** menu: specifies a desired noise level with which to corrupt a synthesized signal (viz. the MATLAB method `add_noise`).
3. **Data Type** menu: this shows the possible classes of data types. Currently these include
- (a) **sig**: a one-dimensional predefined signal;
 - (b) **image**: a two-dimensional predefined image; and
 - (c) **FM**: an FM signal with a frequency modulation that can either be loaded from a file or drawn by the user.

Figure 8.3 depicts the WSPW after selecting the **FM** data type, typing FM3 in the Name field of the signal specification panel, and then clicking the *Load* button.

Further details of the WSPW operation are left to the online documentation.

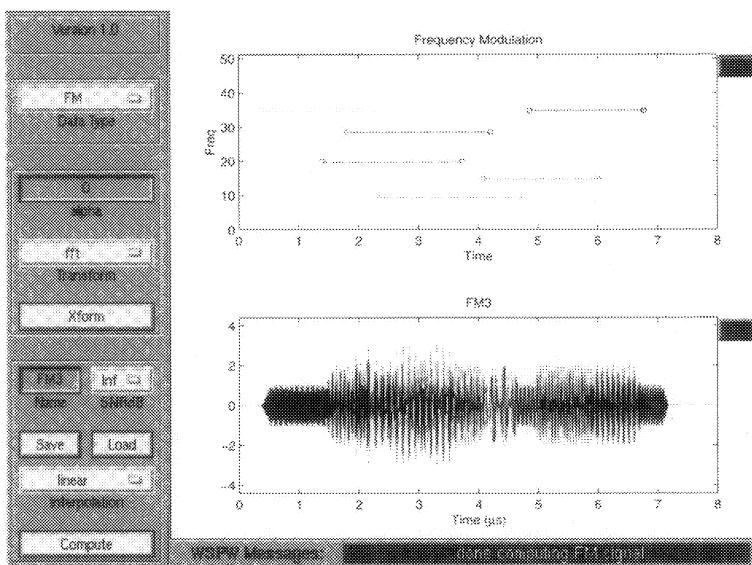


FIGURE 8.3. Loading a FM signal into the WSPW.

8.2 MATLAB Coding

In this section, the underlying design and functions of the WSPW are described. The presentation is tutorial in nature and may be followed at the computer on a step-by-step basis by the reader.

8.2.1 Object-Oriented Programming

In object-oriented programming, one thinks of *classes* of *objects* and associated *methods* that operate on objects of a given class. A class is a concept from which specific instances may be generated. Practically speaking, classes are implemented by specifying a (MATLAB) structure that consists of fields of variables. Associating specific values with these fields yields an instance of the class.

Methods are processes (m-files) that operate on and manipulate the fields of an instance. For example, consider the idea of a “signal” as a class. Fields of such a class would include at a minimum one to hold the data associated with a particular signal and one to hold information about how it was collected, for example, sample rates, intervals, time of day, and so on. A signal instance would be one having specific values for all of these fields. Associated methods that operate on a signal or group of signals might include “powerdB” which computes the power in the signal(s) or “abs” which computes the magnitude of the signal.

8.2.2 Designed Classes

The major classes of objects that have been implemented are

1. `sampled_signal`, and
2. `wavelet`.

The `sampled_signal` class is the parent class from which the `wavelet` is derived. In other words, the `wavelet` class contains a `sampled_signal` as part of its structure.

8.3 The `sampled_signal` Object

Sampled signals are the fundamental objects of digital signal processing. A common type of digital data that comes from real world devices consists of the sample values of an underlying analog signal that has been sampled at a uniform sample rate⁴ above the Nyquist frequency over some time interval. Depending on the signal and collection system specifics, sampling rates and intervals can vary significantly even for the collection of the same underlying analog signal. Together the sampling rate and interval are referred to as the *sample structure*.

This situation has led to development of the `sampled_signal` class. The main motivating factor behind the construction of the sampled signal class is the idea of abstracting, and thereby, unifying the processing of digital signals with distinct sample structures.

8.3.1 Class Construction

Let n be an integer and $\Delta > 0$ be a sample period. As a concept, a `sampled_signal` (one-dimensional) consists of the uniform samples

$$\{p(n\Delta)\}$$

of an underlying continuous-time signal $p(t)$; that is, p is sampled over the discrete set $\Gamma = \{n\Delta\}$. The sampled signal class must then contain a minimum of two elements that describe first, the discrete sample set and second, the values of the function at those points.

In addition, there are some practical restrictions placed on the discrete sampling set Γ ; namely,

- it must be finite,

⁴Of course, there are systems and devices that do *not* operate at uniform sample rates. Nonuniform sampling is not explicitly addressed in this implementation although it is a natural next step.

- it must contain 0, and
- it must have an even number of elements (preferably a power of two).

Thus, a uniform sampling set Γ may be specified by a sample period Δ and a sequential range of indices $n = 0, \dots, 2N - 1$ or $n = -N, \dots, N - 1$. Both associated intervals have a length of $2T = 2N\Delta$. In the former case, the interval contains only nonnegative values whereas in the latter case the interval is symmetric about zero. In the following, discussion is confined to the case in which the sample set is symmetric about 0, although all processing supports both types of intervals.

Some of the main fields associated with a `sampled_signal` object are shown in the following.

```
p <<sampled_signal>> =
  sig: [1024x1 double]
    d: [0.0078 4]
    d0: 0

  interp: '*cubic'

  xlabel: 'Time (\mu s)'
  ylabel: []

  buttondown: 'mouse_limits'
    grid: 'off'
      box: 'off'
    colormap: 'hot'

    name: 'chirp'
  nametrack: 'on'
```

Of these fields, the first three specify the sample signal and the sample values as described in the following. Remaining fields are used primarily for plotting purposes including labeling axes and controlling their appearance.

sig

Signal sample values

$$\text{sig}_n = p(n\Delta), \quad n = -N, -N + 1, \dots, N - 1$$

are stored in the `sig` field of the class as a column vector of doubles. Multiple signals (with the same sample structure) are stored as a matrix where each column of the matrix is a different signal.

Sample interval information is contained in `d` and `d0`.

d

Because the sample set may be described by the two values Δ and $T = N\Delta$, these two values are treated as a unit in the class and referred to as a “sample structure.” The sampled structure is stored in the field *d*. Figure 8.4 shows the sampling set Γ associated with a sample structure $\mathbf{d} = \{\Delta, T\}$. The sampling lattice is uniformly distributed over the interval $[-T, T]$ as

$$\Gamma = \Gamma(\Delta, T) \triangleq \{n\Delta\}_{n=-N}^{N-1},$$

where $N = T/\Delta$ (required to be an integer).

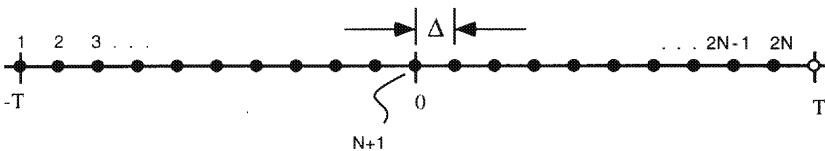


FIGURE 8.4. A symmetric about 0 uniform sample set

d0

The interval type is recorded in the class by the Boolean field *d0*. A value of *d0* = 1 indicates that the interval starts at 0 and a zero value indicates that the interval is symmetric about 0. Figure 8.4 displays a sample structure having a value of *d0* = 0.

interp

The *interp* field is a string that indicates the type of interpolation which is used during any resampling operations. Valid interpolation types are (as used in the MATLAB function *interp1*):

```
'*nearest' - nearest neighbor interpolation
'*linear' - linear interpolation
'*spline' - cubic spline interpolation
'*cubic' - cubic interpolation
```

Of these ‘*cubic’ is the default interpolation method used in resampling. Resampling automatically occurs when performing computations (e.g., addition and multiplication) between sampled signals with different sampling structures. The “*” prefix indicates that an interpolation procedure optimized for uniformly sampled data is to be used.

buttondown

The field `buttondown` is used to associate a procedure to be called each time a mouse button is pressed in a sampled signal plot. This field is set by default to the process `mouse_limits` which allows the user to control the axes limits using the mouse. Its functionality is described in the next section in detail.

name

By default the name of a signal is simply “NoName.” The name of a signal may be changed or viewed by using the sampled signal method `name`.

nametrack

When the `nametrack` field of a signal is set to “on” (the default value) any processing performed on a signal is reflected in its name. For example, if `p` and `q` are two signals named “`p`” and “`q`” then the signal generated by `p+q` would be named “`p+q`.” Turning name tracking off may save significant processing time in repeated call situations.

8.3.2 `sampled_signal` Methods

There are over 100 methods that have been coded for manipulating and viewing sampled signals. Table 8.1 summarizes some of the main `sampled_signal` methods and their functionality. A complete listing is available online. One powerful aspect of the object-oriented programming approach is the ability to *overload* operators. Typical overloaded operators include algebraic operations such as addition “`+`” and multiplication “`*`”, although any function may be overloaded. Overloading a function allows the ordinary operation associated with a function to be superseded by an alternative class appropriate operation. In particular, this feature has been exploited in the redefining of arithmetic operations between sampled signals of differing sample structures.

In each of the following subsections, the reader is presented with a method and associated MATLAB commands that if executed in order (following the subsection sequence) illustrate the method’s functionality. MATLAB commands are prefixed with the MATLAB prompt “`>>`.”

Constructor (`sampled_signal`)

Perhaps the most important method for a class is its constructor. The `sampled_signal` constructor returns a `sampled_signal` object and takes many different sets of arguments. These are all shown in the following.

<i>Method</i>	<i>Description</i>
sampled_signal	Constructor.
plot	Display signal(s).
fft	Fast Fourier transform.
resample	Interpolate to different sample structure.
conv	Convolve.
translate	Translate.
dilate	Dilate.
involute	Involute.
threshold	Apply threshold to signal.
sigcat	Concatenation.
add_noise	Perturb signal with white noise.
join	Combine multiple objects.
split	Decompose a complex object.
align	Align starts of signals.
bandpass	Bell-shaped bandpass filter.
der	Compute digital derivative.
int	Compute digital integral.
mean	Compute mean.
upshift	Translate frequency spectrum.
powerdB	Compute signal power (dB).
normalize	Rescale to unit norm.
support	Determine the signal support interval.
name	Sets or gets a signal's name.
+ , - , .* , ./	Arithmetic operators.
==, !=, <, <=, >, >=	Relational operators.

TABLE 8.1. Summary of some `sampled_signal` methods.

```
% Constructor for sampled signal class
%
% sampled_signal(<sample_structure>)
% sampled_signal(<sampled_signal>)
% sampled_signal(struct(<sampled_signal>))
% sampled_signal(<matrix>,<sample_rate>)
%
% sampled_signal(<name>,<sample_structure>,varargin)
% sampled_signal(<name>,<sampled_signal>, varargin)
%
% Possible names/varargs:
%
%-----%
%   name      % optional argument(s)    % argument default
%-----%
%           %                         %
%
```

```
% 'sinc'          %
% 'impulse'      %
% 'box'           % half width           % 1/2
% 'sine'          %
% 'exp'           %
% 'indicator'    % interval end points (a,b) % (0,1)
% 'packet'        %
% 'opacket'       %
% '2packet'       %
% 'chirp'         % chirp rate, upshift   % 1/2, none
% 'chirp2'        % chirp rates          % [1 3]/2
% 'chirp10'       % chirp rate (upshift by 10) % 1/2
% 'noise'         % randn distributed signal %
%                   %
%-----%-----%
```

One fast and easy way to create a predefined sampled signal is to call the constructor with a name argument. For example, the command

```
>> p=sampled_signal('chirp',[1/128 4]);
```

typed at the MATLAB prompt sets `p` to the chirped signal sampled at a 128MHz rate over the interval $[-4, 4]$ with the fields shown in Section 8.3.1 and plotted in Figure 8.5a.

plot

`Plot` displays a sampled signal, a matrix of sampled signals, or multiple sampled signals. If the input is an $m \times n$ matrix of sampled signals `plot` creates an $m \times n$ grid of axes and displays each signal in the corresponding axes.

For complex signals `plot` displays the real, imaginary, and magnitude envelope of the signal on a single axis. These three items are colored according to the color order property of the axes.

```
% function hout=plot(p,varargin)
%
% plot sampled signal matrix
%
% hout = matrix of line handles,           if p is a
%                                1x1 sampled_signal
%
% hout = a cell array of line handles of size(p), ELSE
%
% plot(<sampled_signal>)
% plot(<sampled_signal>,<sampled_signal of same dimension>,...)
```

Figure 8.5 illustrates the results of plotting the chirp sampled signal p and a 2×1 sampled signal consisting of the preceding chirp signal and the signal packet constructed as

```
>>p(2)=sampled_signal('packet',[1/200 3]);
>>p=p(:); % make p a column vector
```

This makes p a 2×1 sampled signal matrix. Note that the sample structures of $p(1)$ and $p(2)$ of the two signals are different.

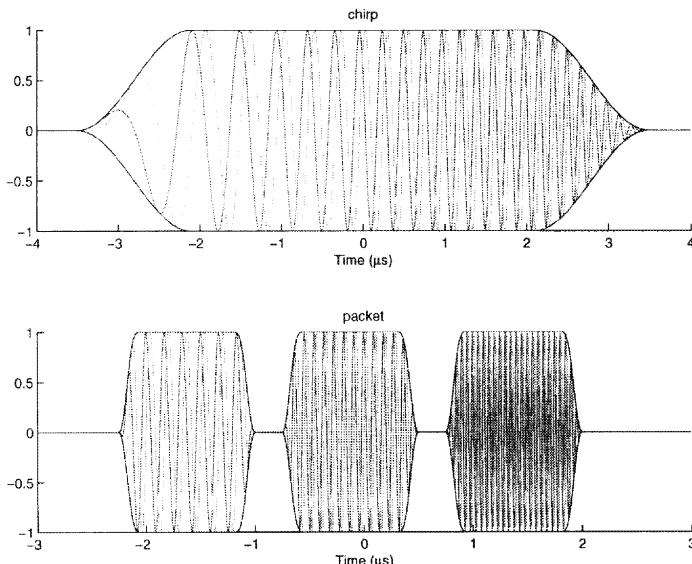


FIGURE 8.5. Displaying using plot: A 2×1 sampled signal consisting of the chirp and packet signals.

Figure 8.5 is generated by the MATLAB command:

```
>> plot(p) % plots both the chirp and packet
```

To plot just one of these functions, one can use the MATLAB command `plot(p(1))` or `plot(p(2))`.

Arithmetic Operators +, -, ., ./*

Most of the arithmetic operators have been overloaded with sampled signal appropriate methods. Essentially, algebraic operations between sampled signals with different sampling structures are facilitated by resampling the signal operands to a common structure and then applying the operation. Thus, the resulting sampled signal may have a unique sample structure associated with it. Shown in the following is the functionality of the sampled signal `plus` operator. The remaining operators, `-`, `.*`, and `./`, are all similarly defined. Note in particular that the operators function with hybrid arguments of scalar double, vector doubles, and sampled signals.

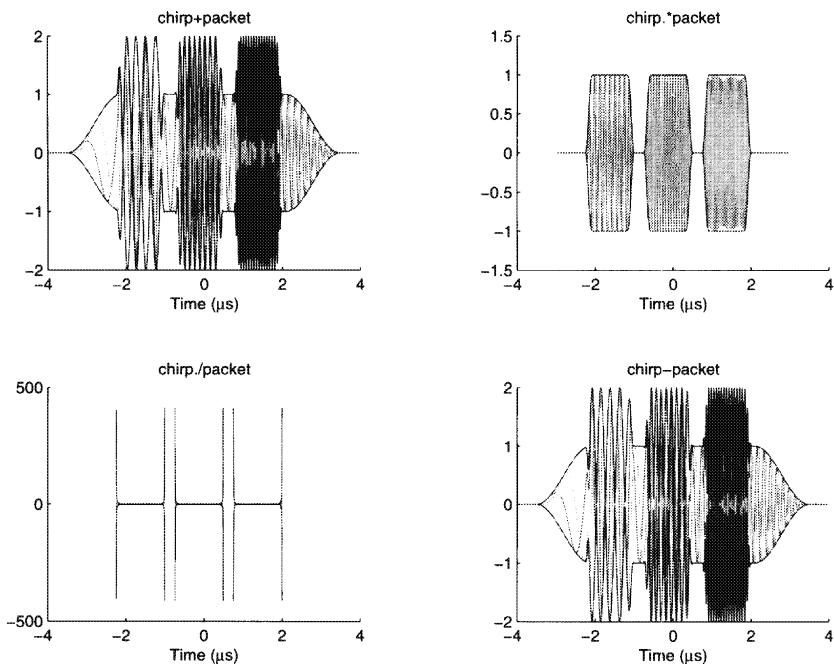


FIGURE 8.6. Arithmetic combinations of members of `p`.

```
% function r = plus(p,q)
%
% plus(<1x1 SS>,<const>)
% plus(<1x1 SS>,<arb. dimension SS>)
%
% plus(<arb. dimension SS>,<same dimension SS>)
%
% FOR 1x1 SS
```

```
%<scalar> + p      % pure scalar add
%    p + <scalar>
%
% <1 x n> + p      % each column signal of p is scalar added
%    p + <1 x n>
%
%    p + q          % sampled signals are added with
%                      % resampling if necessary
```

The following MATLAB code creates a 2×2 `sampled_signal` array `q` from arithmetic combinations of the chirp and packet signals and then plots the results.

```
>> q(1,1) = p(1) + p(2);
>> q(1,2) = p(1) .* p(2);
>> q(2,1) = p(1) ./ p(2); % this generates divide by zero warning

>> q(2,2) = p(1) - p(2);
>>
>> plot(q)
```

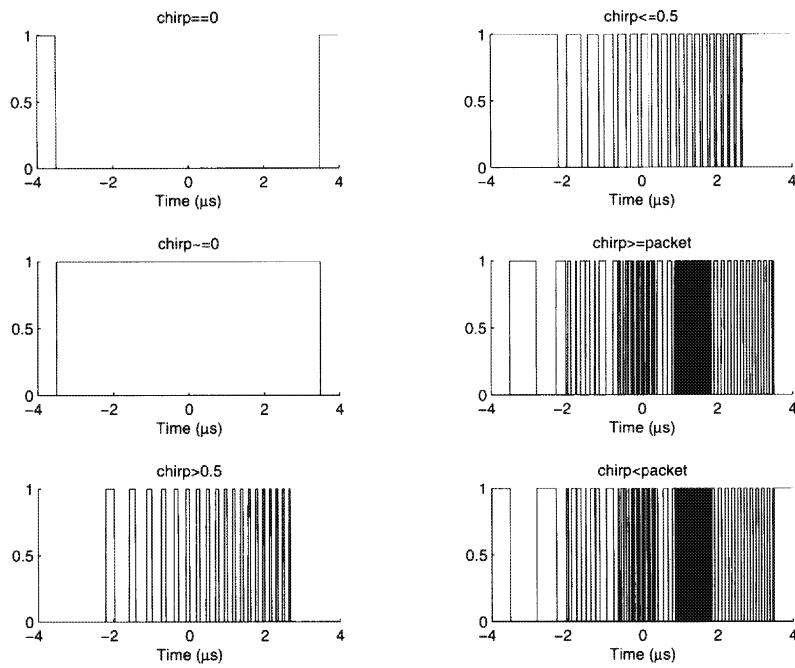
Figure 8.6 shows the different arithmetic combinations of the chirp and packet signals placed into `q`.

Relational Operators `==`, `≠`, `<`, `<=`, `>`, `>=`

In addition to the arithmetic operators, the relational operators also have been overloaded to behave in a `sampled_signal` class appropriate manner. The following MATLAB code segment illustrates the functionality of some of the relational operators.

```
>> q(1,1)=p(1)==0;
>> q(1,2)=p(1)^=0;
>> q(1,3)=p(1)>0.5;
>>
>> q(2,1)=p(1)<=0.5;
>> q(2,2)=p(1)>=p(2);
>> q(2,3)=p(1)<p(2);
>>
>> plot(q)
```

Generated by the plot command, Figure 8.7 depicts the resulting Boolean sampled signal `q`.

FIGURE 8.7. Relational operations on members of `p`.*name*

This method sets or gets a signal's name.

```
% function out = name(p,str)
%
%     p      = name(p,<string>)  % sets p's name to string
%     string = name(p);           % gets p's name
```

For example, `name` applied to the signal `p` acts as follows.

```
>> name(p)
```

```
ans =
```

```
'chirp'  
'packet'
```

fft

The method *fft* overloads the normal MATLAB *fft* and performs a one-dimensional *sampled_signal* appropriate fast Fourier transform.

```
% function p=fft(p)
%
% fast Fourier transform for sampled signal
```

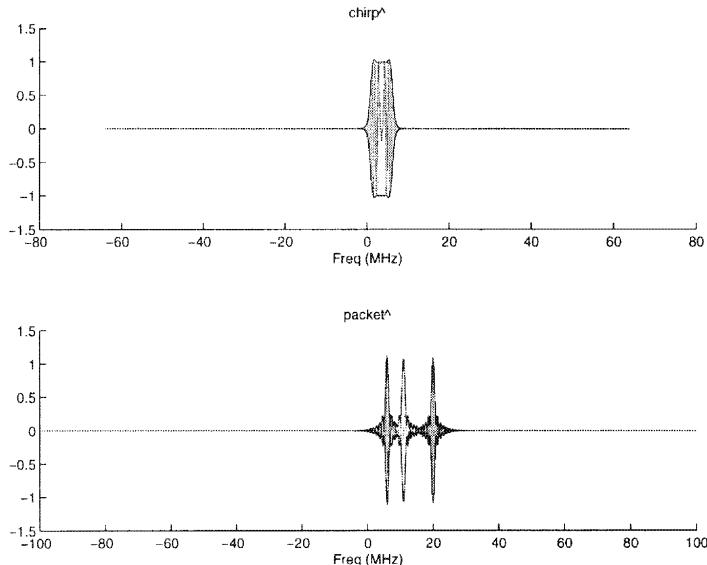


FIGURE 8.8. FFT of the 2×1 sampled signal matrix.

One benefit of the sampled signal class is that the computation and display of a Fourier transform of sampled signal *p* may be simply specified in MATLAB as follows.

```
>> plot(fft(p)) % compute and show the fft of p
```

The result of this command is displayed in Figure 8.8. Note that the frequency interval over which each signal is displayed corresponds to the Nyquist bandwidth associated with each, that is, 128MHz and 200MHz, respectively.

resample

`Resample` uses interpolation to estimate the values of sampled signals on different sample lattices described by different sample structures. The type of resampling done is governed by the `interp` field of the `sampled_signal` argument.

```
% function p=resample(p,pd_new,t0)
%
% resample the function p with sample structure pd
%   with respect to the new sample structure pd_new
%   and possible offset t0
%
%   method of resampling is controlled by p.interp
%
%   zero values in pd_new are substituted with the
%   corresponding value in p.d
```

Resampling provides the means for the algebraic combination of sampled signals with different sampling structures. Given two sampled signals with differing sampling structures operations such as addition, subtraction, and multiplication cannot be performed directly. A basic approach to performing operations on signals with mismatched sample structures is to first resample and then directly combine the resampled signals on the common sample set.

Figures 8.9 and 8.10 show the resampling of the chirp and packet signals for two different sample structures. Figure 8.9 illustrates severe undersampling (200MHz and 128MHz rates to 5MHz) and Figure 8.10 illustrates cubic interpolation to a 500MHz rate. These figures are generated by the commands

```
>> plot(resample(p,[1/5 5]))    % severe under sample
Warning: resfac=-4.6781, resampling accuracy may be inadequate
Warning: resfac=-5.3219, resampling accuracy may be inadequate
>> plot(resample(p,[1/500 5]))
```

Note that the severe undersampling requested yields a warning in the sampling process. A warning is generated whenever the resolution factor $\text{resfac} = \log_2(\Delta/\Delta_{\text{new}})$ exceeds the value 2. In the figure, the left plot shows the results of this undersampling, and the right shows the results of resampling at the 500MHz rate.

conv

This method performs convolution between sampled signals. Convolutions are computed in the Fourier domain via `fft` and `ifft` operations.

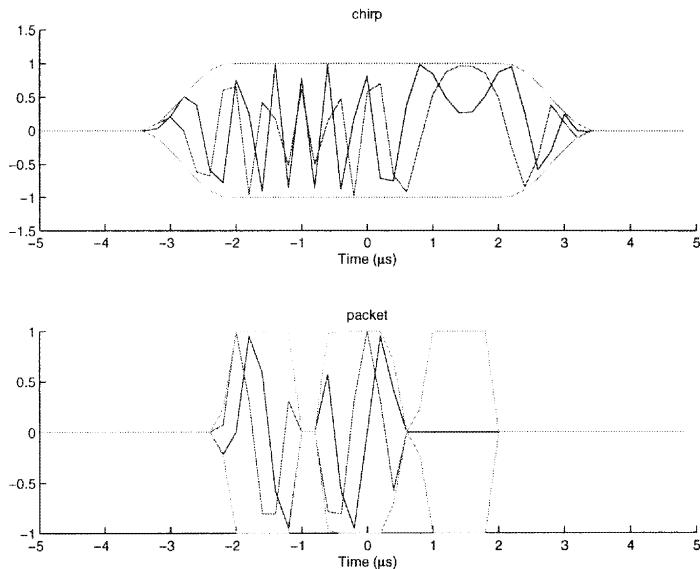


FIGURE 8.9. Resampling the chirp and packet signals with the sample structure [1/5 5] (severely undersampled).

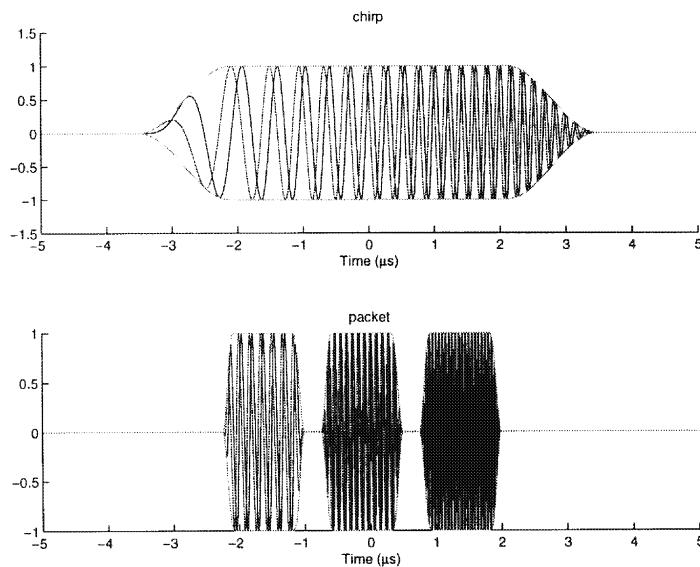


FIGURE 8.10. Resampling the chirp and packet signals with the sample structures [1/500 5].

```
% function r=conv(p,q)
%
% Convolution (performed in Fourier domain)
%
% conv(<sampled_signal>,<sampled_signal>)
% conv(<sampled_signal>,<scalar>) % convolve p with itself q times
```

Figure 8.11 displays the results of convolving the chirp signal with itself (top), the packet signal with itself (middle), and finally the chirp with the packet (bottom). This figure may be reproduced using the following MATLAB code.

```
>> q(1)=conv(p(1),p(1));
>> q(2)=conv(p(2),p(2));
>> q(3)=conv(p(1),p(2));
>> plot(q)
```

As seen in the figure, the convolution operator is denoted by a double asterisk '**' in order to distinguish it from the multiplication single asterisk operator '*'.

translate

```
% function p = translate(p,t0)
%
% Translate sampled signal by t0
%
% Signals are cyclically translated in the interval p.d
% with resampling if necessary
```

Translation moves a sampled signal along its axis by an amount t_0 . The translation is performed cyclically over the sample interval and requires resampling in the event that t_0 is not an integer multiple of the sample period (class field `d(1)`). Figure 8.12 shows the result of translating the sampled signal matrix `p` by $2\mu s$:

```
>> plot(translate(p,2))
```

To avoid unwanted wraparound effects the user may first zero pad the sampled signal (viz. `zeropad.m`).

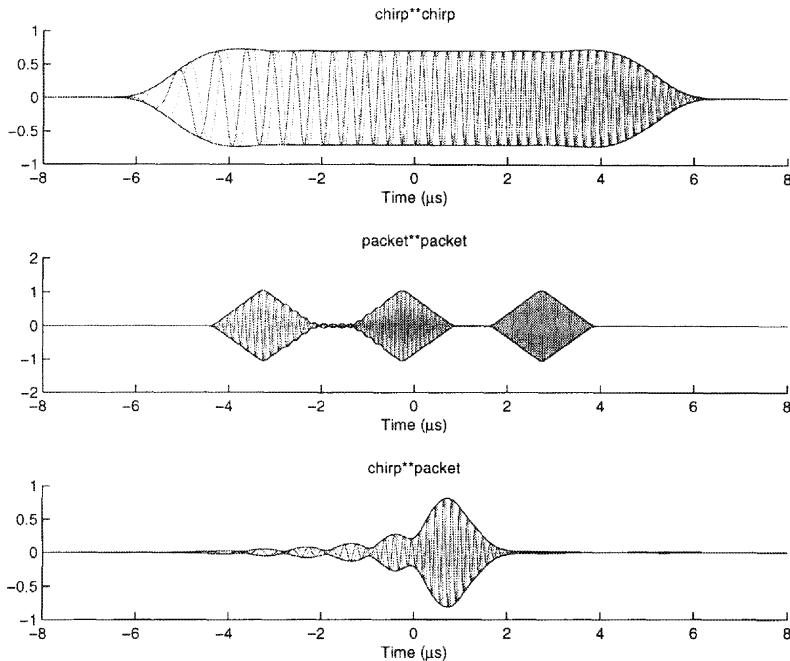


FIGURE 8.11. Convolution of the packet and chirp signals.

dilate

The L^n -norm preserving dilation by a value s of a function $p(t)$ results in a new function $s^{-n}p(st)$. This function is digitally emulated on sampled signals by the method *dilate*.

```
% function p = dilate(p,scale,n)
%
% L^n dilation of p.sig by the value scale
```

involute

This method performs the involution operation $\tilde{p}(t) \stackrel{\triangle}{=} \bar{p}(-t)$.

```
% function p = involute(p)
%
% Involute (flipud and conj) the signals in p
%
```

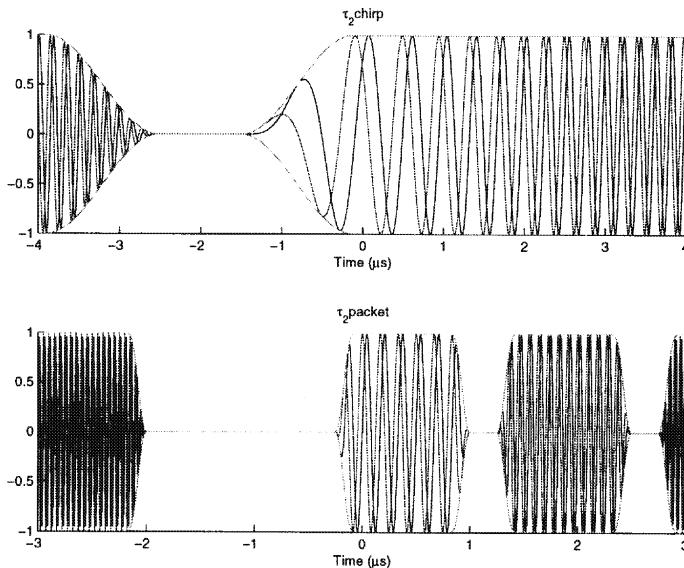


FIGURE 8.12. Translated by 2 version of p

threshold

This method thresholds a signal based on a δ -percentage of its maximum magnitude.

```
% function p = threshold(p,delta,how)
%
% Threshold signal based on delta percentage
% of the max magnitude of sig
%
% The optional input argument, how, controls how
% the threshold is interpreted
%
%     how = 'absolute' or ['relative']
%
% A value of 'relative' interprets the threshold relative
% to the maximum
```

sigcat

The method `sigcat` concatenates signals in a `sampled_signal` object; that is, `p.sig = p.sig(:)`. It is especially useful when plotting a large number of signals in a sampled signal object.

```
% function p=sigcat(p,int_index)
%
% concatenate signal portion of p
%
% set int_index=1 to make the x-axis
% be labeled with signal indices (defaults to 0)
```

add_noise

This method perturbs a signal with white (possibly analytic) noise to yield noisy versions with given signal-to-noise ratios.

```
% function [pnoisy,noise] = add_noise(p,SNRs,analytic)
%
% <sampled_signal> =
%     add_noise(<sampled_signal>,<double>,<scalar boolean>)
%
% Add White noise, w, to a signal, p, such that the resulting
% signal, p+w, has a signal to noise ratio of SNR dB, i.e.,
%
%     SNRdB = 20 log10( norm(p)/norm(w) ) .
%
%
% analytic =1 (default) creates analytic noise
%             (no negative freq. content)
%
% SNRs may be a vector in which case each SNR is applied
% to p and returned in a concatenated sampled_signal
%
% See Also: powerdB.m
```

Consider adding noise to the chirp and packet signals in the 2×1 sampled signal matrix *p*. The following MATLAB code fragment creates a new sampled signal object *pnoisy* that contains noisy versions (signal-to-noise ratios of SNR = ∞ , 6, and 0dB) of the signals in *p*.

```
>> [pnoisy,noise]=add_noise(p,[inf 6 0]);
>>
>> plot(sigcat(pnoisy))
```

The result of the plot command is displayed in Figure 8.13.

In the preceding, the noise component is returned in the variable *noise*. To verify the operation of *add_noise* the method *powerdB* can be employed in the following way.

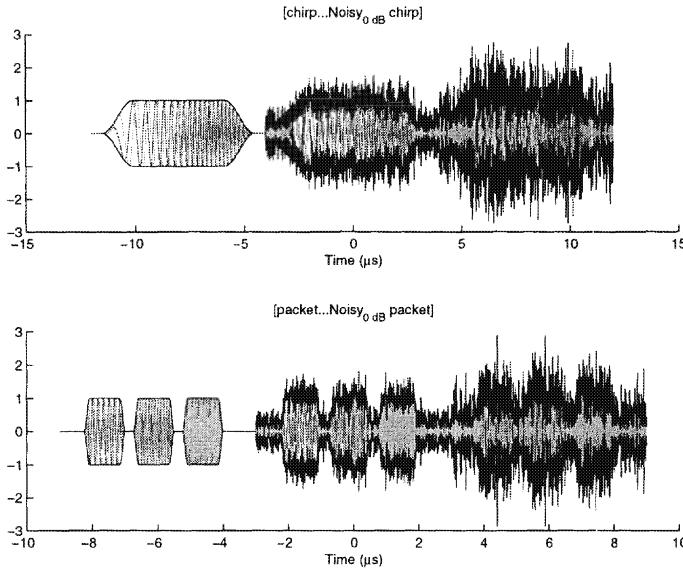


FIGURE 8.13. Noisy versions of the sampled signal object `p`, SNRs = ∞ , 6, and 0.

```
>> [pnoisy,noise]=add_noise(p,3); % create a 2x1 corrupted
>> % sampled signal
>>
>> SNRdB = powerdB(p)-powerdB(noise)

SNRdB =
3.0000
3.0000
```

join

This method is useful for combining multiple sampled signal objects into a single sampled signal object. When combining sampled signals with different sample structures, all signals are resampled to a new sample structure given by the minimum sample period and maximum interval over all sampled signals to be combined.

```
% function r=join(p,q)
%
% Concatenate matrices of sampled signals to a
```

```
% single sampled signal with resampling if necessary
%
% r.sig is a length(p(:))+length(q(:)) signal matrix
%
% See Also split.m
```

For example, joining the 2×1 sampled signal object *p* yields a new 1×1 sampled signal object as follows.

```
>> pj=join(p)

pj <<sampled_signal>> =

    sig: [1600x2 double]
        d: [0.0050 4]
    d0: 0
    dy: []
    dy0: []
    SNR: Inf
    interp: '*cubic'
    xlabel: 'Time (\mu s)'
    ylabel: []
    buttondown: 'mouse_limits'
    grid: 'off'
    box: 'off'
    name: '[chirp, packet]'
    colormap: 'hot'
    nametrack: 'on'
```

split

This method converts a single sampled signal object that contains *n* signals into a $n \times 1$ sampled signal object array of each with a single signal. In this way, it performs a function that is inverse to the method *join*, although it is not a strict inverse due to the possible resampling that may be required.

```
% function r=split(p)
%
% Split a 1x1 sampled signal object into a
% multi-dimensional sampled signal object of the
% same dimensions as the .sig element of input
%
% See Also join.m
```

When applied to the joined chirp and packet signals *pj* illustrated pre-

viously, the split method recovers a (more dense) resampled version of the original p . In fact, in this case, the energy in the error between the original and resampled versions is zero. This can be verified with the MATLAB code segment that follows.

```
>> ps=split(pj); % split the joined version of p
>> norm(ps-p)    % look at the energy in the differences

ans =
[0]
[0]
```

align

This method aligns starts of pulse-like signals to a specific domain value x_0 . Starts are defined by the level crossing of the magnitude which is at a specified percentage of its maximum.

```
% function p=align(p,x0,how)
%
% Align signals in p to the value x0
%
% how =
%     'lead'    % leading edge          % sets how=0.25
%     'max'     % maximums            % sets how=0.99
%
%     double   % delta percentage % defaults to 0.25
%
% See Also: time_align.m
```

The following MATLAB code segment aligns the points at which the magnitudes of the chirp and packet signals cross a value of 0.25 of their maximum to $-2.5\mu s$. Depicted in Figure 8.14 is the resulting alignment as generated in the following.

```
>> pa=align(p,-2.5);
>> plot(pa)
>>
>> % set x-axis to same scale using mouse
```

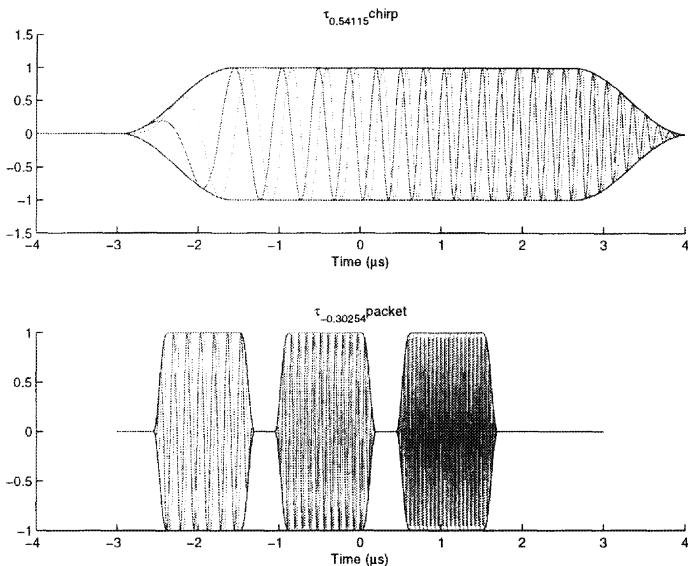


FIGURE 8.14. Time domain alignment of the chirp and packet signals.

bandpass

This method applies a bell-shaped bandpass filter with a specified bandwidth and center frequency to sampled signals.

```
% function p=bandpass(p,BW,CF);
%
% Bell-shaped bandpass filter with specified
% bandwidth BW and center frequency CF
```

Figure 8.15 shows the results of passing the chirp and packet signals through a bandpass filter with a 4MHz bandwidth and a center frequency of 6MHz. This can be reproduced in MATLAB using the following commands.

```
>> pb=bandpass(p,4,6); % bandpass with BW=4MHz and CF=6MHz
>>
>> plot([pb fft(pb)]) % show time and frequency domain of result
>> % zoom in on frequency region using mouse
```

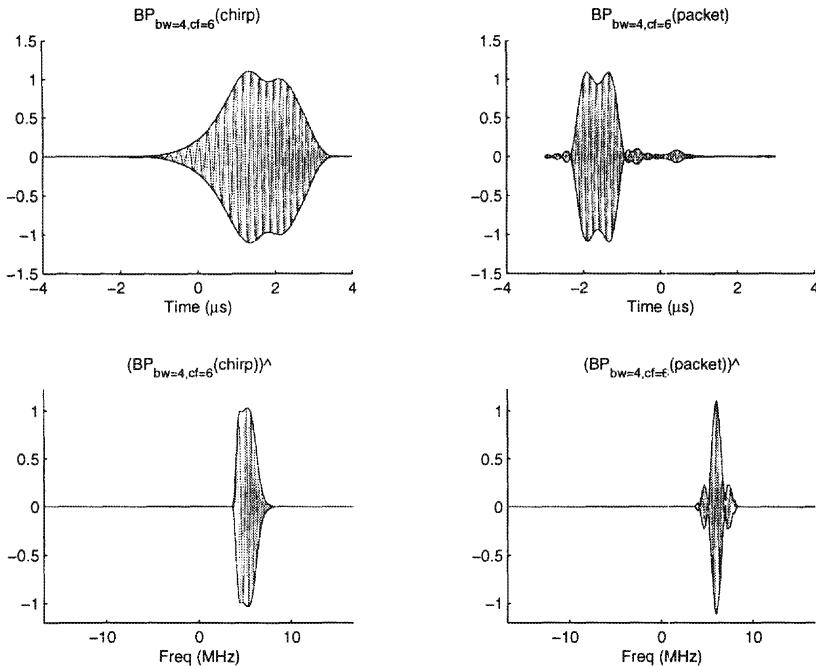


FIGURE 8.15. Bandpassed versions of the chirp and packet signals.

der

This method computes a coarse digital “derivative” of the input signal argument; it essentially performs a digital `diff` operation.

```
% function p = der(p)
%
% provides approximate derivative(s)
```

Figure 8.16 shows the result of performing the `der` method on the chirp and packet signals. It may be generated by the simple MATLAB command:

```
>> plot(der(p))
```

int

This method computes the running “integral” of the input signal argument. Essentially it performs a digital `cumsum` operation.

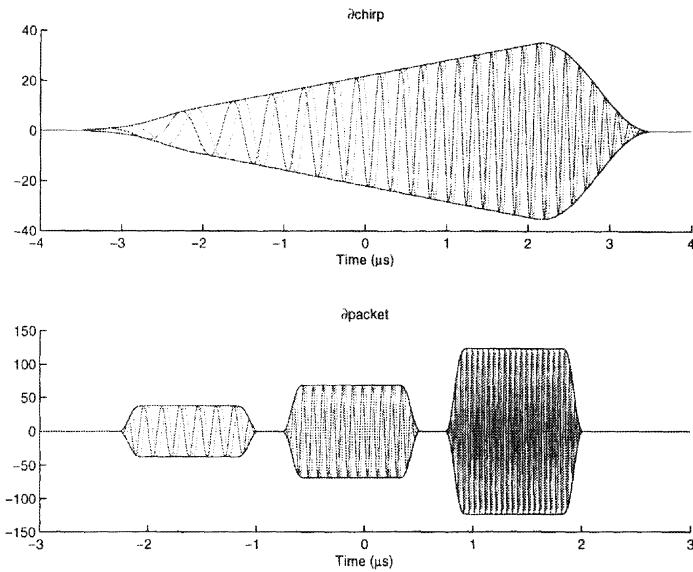


FIGURE 8.16. Derivative of the chirp and packet signals.

```
% function p = int(p)
%
% provides approximate integration
% using Simpson's method
```

Figure 8.17 shows the result of performing the `int` method on the real parts of the chirp and packet signals. It may be generated by the simple MATLAB command:

```
>> plot(int(real(p)))
```

mean

This method computes sampled signal means (scalars) or mean sampled signals (signals).

```
% function mm = mean(p,dim)
%
% dim=1: Compute the mean of each signal in p
%
```

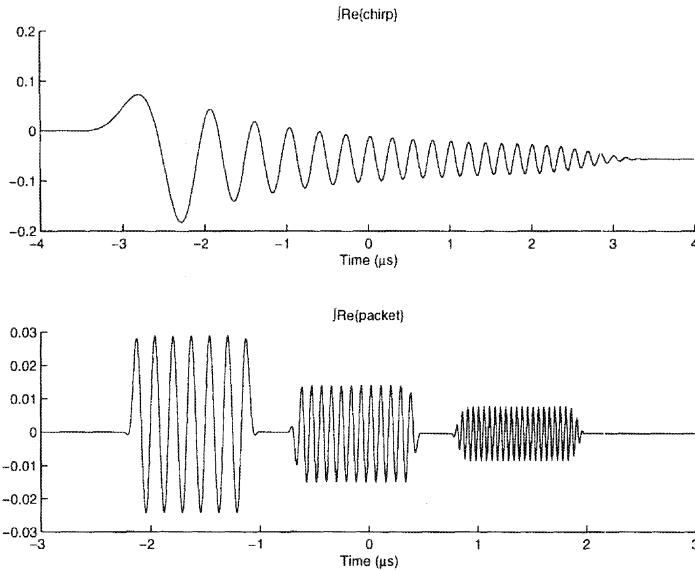


FIGURE 8.17. Integral of the real parts of the chirp and packet signals.

```
% dim=2: Compute the mean signal
%
```

As computed in MATLAB, the chirp and packet signal means are

```
>> mean(p)

ans =
[-0.0071+ 0.0100i]
[-0.0001+ 0.0031i]
```

whereas the mean sampled signals are themselves since there is only one signal to average.

upshift

```
% function p = upshift(p,df)
%
% Upshifts p by df frequency units
```

The following MATLAB command shifts the chirp and packet signals in

p by 20MHz and then plots the Fourier transform of the result as shown in Figure 8.18.

```
>> plot(fft(upshift(p,20)))
```

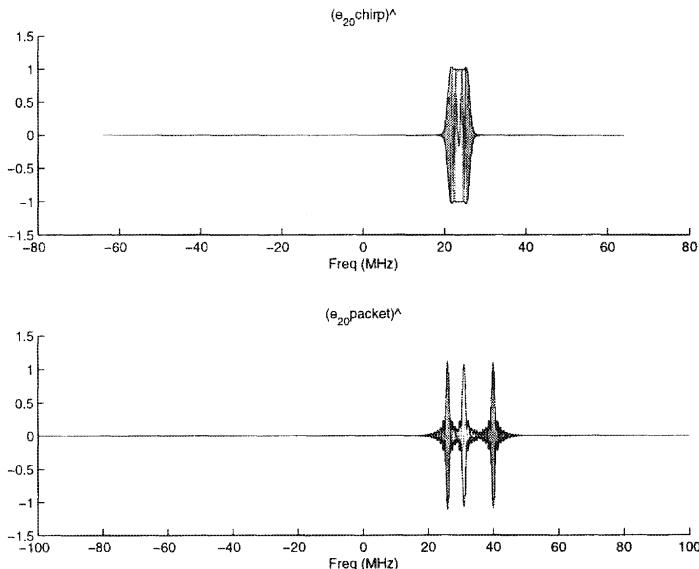


FIGURE 8.18. Fourier transform of the 20MHz upshifted chirp and packet signals.

powerdB

For `sampled_signal` objects containing a single signal, `powerdB` returns a scalar indicating the power of the signal in units of decibels (dB). For `sampled_signal` objects containing multiple signals a new `sampled_signal` object is returned that contains a single signal representing the power versus signal index.

```
% function p = powerdB(p)
%
% Computes power in dB of signals in p
```

For example, one may compute and display the power in the noisy signals `pnoisy` using the commands:

```

>> pow=powerdB(pnoisy)
pow <<sampled_signal>> =
2x1 struct array with fields:
    ...fields deleted...
>>plot(pow)

```

Displayed in Figure 8.19 is the result of the plot command. It consists of two axes corresponding to the chirp and packet signals. The first point in each plot corresponds to the no-noise added case and so gives the power in the clean signal. Note that index 1 corresponds to the 1dB SNR case and that the power calculation shows an increase of 1dB there and similarly index 2 has a 3dB increase in power. This provides a check on the proper operation of the noise perturbation method `add_noise`.

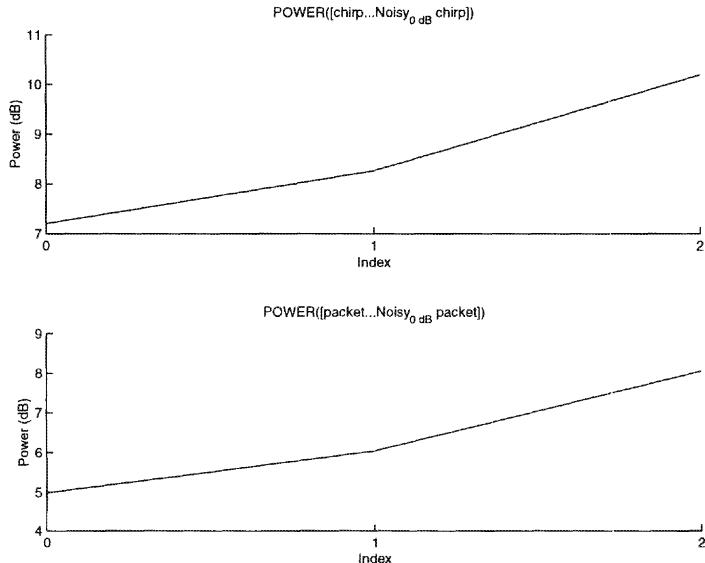


FIGURE 8.19. Power as a function of index (0–2) for the perturbed signals chirp (top) and packet (bottom).

normalize

This method normalizes sampled signals so that the L^n is one. It digitally emulates the L^n normalization of $p(t)$

$$p_{\text{norm}}(t) = \frac{p(t)}{\|p\|_n} = \frac{p(t)}{\left(\int |p(t)|^n dt\right)^{1/n}}.$$

```
% function p = normalize(p,n)
%
% Normalize the sampled signal p using the L^n norm
%
% n=inf means max normalization
```

With $n = 2$ an energy normalization is performed and with $n = \infty$ a maximum (digital sup) normalization is performed so that the maximum magnitude of the signal is 1.

support

This method finds a domain interval over which a signal has significantly nonzero magnitude.

Often it is useful to determine the interval over which a function has a magnitude that is significantly above zero. This function is performed by the method `support` which takes as an argument a threshold value $\delta \in (0, 1)$. The threshold value quantifies the significance relative to the max-normalized version of the signal. For example, a value of $\delta = 0.25$ means that a value is significant only if it is at least a quarter of the value of the maximum. The smallest and largest time indices at which the signal crosses the significance value is what is returned in the first two arguments by the method.

Determines the delta-support of sampled signals.

```
% function [t0,t1,n0,n1] = support(p,delta)
%
% Determines start and end of a sampled signal p according
% to threshold parameter delta
%
% t0,t1 are the the start and stop times with respect to p.d
% n0,n1 are the indices associated with the start and stop
%
% Synopsis:
%
%      [t0,t1,n0,n1] = support(p,delta)
%      [t0,t1] = support(p,delta)
```

For example,

```
>> [a,b] = support(p(1),0.25);
```

yields values of $a = -3.0411$ and $b = 3.0255$.

8.4 Wavelet Transform Implementation

The wavelet transforms discussed here are a special type of discretized continuous wavelet transform that may be implemented using filter banks. Because such a transform is, in general, overcomplete it is called an *overcomplete wavelet transform*. A review of the OCWT and the MATLAB specification of the associated filter banks is presented in the following subsections.

8.4.1 Overcomplete Wavelet Transform (OCWT)

This section briefly reviews the CWT and OCWT which are discussed in detail in Chapters 4 and 6, respectively.

Recall that the continuous wavelet transform) converts a one-dimensional signal $f(t)$ to a signal of two dimensions, time t and scale s (or equivalently frequency), and is given by

$$W_g f(t, s) = \langle f, \tau_t D_s g \rangle = (f * D_s \tilde{g})(t).$$

Because it may be written as a convolution, the wavelet transform may be interpreted in terms of a filter bank with a special structure. Namely, the impulse responses of a wavelet filter bank are all related to each other by dilation. Explicitly, the bank is given by

$$\{D_s \tilde{g}\},$$

where s ranges continuously over the interval $(0, \infty)$. This corresponds to an infinite number of filters. In a practical implementation, of course, it is necessary to have a finite number of filters in the filter bank. As a consequence, it is not possible to compute the full continuous wavelet transform using a finite precision digital machine; however, it is possible to obtain fine approximations to it. This entails the fine sampling (in the time-scale plane) of an underlying continuous wavelet transform.

Conceptually, the OCWT is defined by a CWT, W_g , evaluated at specific time-scale points given by a sampling grid Γ . Specifying a finite wavelet filter bank corresponds to sampling the scale axis in a specific way.

For “fine enough” samplings in the time-scale plane, the overall transform becomes overcomplete. In other words, the resulting coefficients are *not* linearly independent; that is, they are redundant. This type of discretized wavelet transform has been called an overcomplete wavelet transform.

Sampling sets used in the OCWT filter bank implementation are given by two sampling constants; Δ (specifies the spacing on the time axis) and a_0 (specifies the spacing on the scale axis). These constants yield a *semilog regular* sampling grid $\Gamma(\Delta, a_0)$. Recall that a *semilog regular* sample set has the form

$$\Gamma(\Delta, a_0) \triangleq \{n\Delta\} \times \{a_0^m\}$$

(viz. Definition 6.2 on page 131).

Further details of the filter bank implementation are discussed in Section 6.2. The reader may wish to examine Figure 6.2 on page 138 which depicts a typical OCWT analysis filter bank.

8.4.2 GUI for Filter Bank Specification

This section describes a graphical user interface for the construction of OCWTs. Since a forward OCWT is defined by an *analysis* filter bank and a uniform time sampling, a way of specifying wavelet filter banks together with time sampling is needed.

Because of its wavelet structure, an OCWT filter bank is further fully determined by the specification of the analyzing wavelet g and scale samples $\{a_0^m\}_{m=0}^{n_f-1}$, where n_f is the number of filters in the bank. In accordance with the semilog regular sampling, time sampling is accomplished via a common *sample structure* for each filter in the filter bank. Recall that a sample structure consists of a sampling period Δ and an interval $[-T, T]$ that defines a corresponding time sampling set.

Thus, the main items needed to implement a wavelet filter bank are:

1. the analyzing wavelet g , and
2. the time-scale sampling set Γ .

In the OCWT implementation these break down further into several other parameters. These are discussed in the following paragraphs.

Analyzing wavelet

In the MATLAB GUI of Figure 8.20 there are several families of wavelets that may be specified. These include the

- a Morlet (shown),
- b Frequency B-spline,
- c Parametric bandlimited (PBL), and

d Daubechies

wavelets. All of these families of wavelets have a set of parameters associated with them that may be set using the interface.

Time-scale sampling set

To define a semilog regular time-scale sampling set there are four values that must be specified:

- a time sampling rate Δ^{-1} ,
- b time interval bound T ,
- c filter bank exponential spacing constant a_0 , and
- d the number of filters n_f .

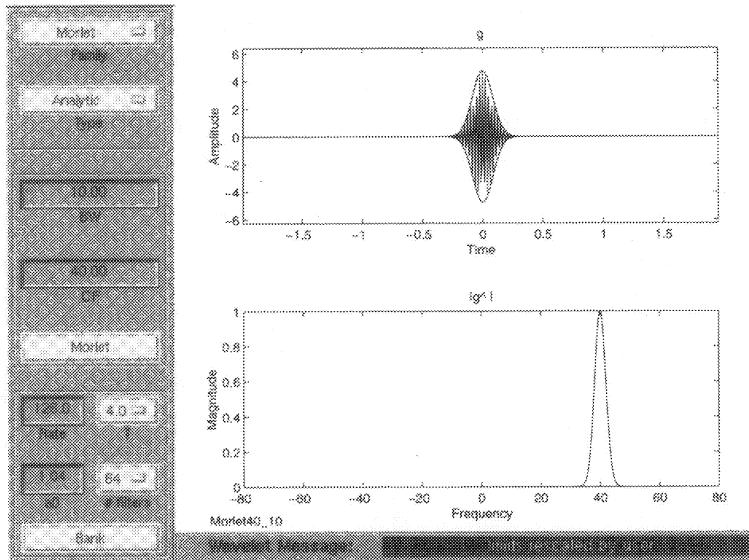


FIGURE 8.20. Graphical user interface for specifying wavelet filter banks.

Invoked by the MATLAB command

```
>> uibank
```

the MATLAB interface for specifying wavelets and wavelet filter banks is shown in Figure 8.20. Along the left side of the window are placed the main user interface controls. These are grouped into three subpanels of controls: top, middle, and bottom. These are now described from top to bottom.

Top

The top panel of controls contains two menus that specify the wavelet family name and the type of family: one of either **Analytic** (wavelets supported only on positive frequencies) or **Real** (wavelets symmetric about 0 frequency).

Middle

The middle panel of controls is dependent on the wavelet family chosen in the upper panel and contains controls to adjust parameters specific to the family. For example, as pictured in the figure, the Morlet wavelet has the two parameters of center frequency and bandwidth (variance) associated with it. The very bottom control in this group is a button that initiates the construction and plotting of the wavelet (time domain and frequency domain plots) using the parameters selected. In Figure 8.20 the window state after depressing the “Morlet” button is displayed. The top plot shows the time domain impulse response of the wavelet filter $n_f - 1$ and the bottom plot shows the corresponding magnitude Fourier transform. Note that only the last filter, the one with the largest center frequency and bandwidth, is displayed. To see the entire filter bank the “Bank” button at the bottom of the next panel may be used.

Bottom

The bottom panel contains the time-scale plane sampling parameters: sample rate Δ^{-1} , sample interval T , dilation constant a_0 , and the number of filters in the bank. At the very bottom of the panel is the “Bank” button which displays the wavelet filter transfer functions for the entire filter bank in an overlaid fashion. In addition the filter bank support function G_{part} is displayed in a separate plot. Here the function is defined as

$$G_{\text{part}}(\gamma) = \sum_{m=0}^{n_f - 1} |\hat{g}(a_0^{-m}\gamma)|^2.$$

8.5 The `wavelet` Object

8.5.1 Class Construction

The main fields associated with a `wavelet` object are

1. a filter bank field `FB` which is a structure describing the underlying wavelet filter bank;
2. a signal field `p` which is a `sampled_signal` which contains the function on which the wavelet transform is taken; and

3. a signal field `sampled_signal` which is an inherited `sampled_signal` object containing the wavelet transform itself.

To see the wavelet object structure in MATLAB, one can use the constructor `wavelet`. For example, after a filter bank has been chosen using the interface (Figure 8.20), typing

```
W = wavelet(GetFB)
```

at the MATLAB prompt calls the wavelet constructor and returns the wavelet object:

```
W <<wavelet>> =
FB:
  ghatbank: [1x1 sampled_signal]
    scales: [1x64 double      ]
      name: 'Morlet40_10'

p:
  [1x1 sampled_signal]

sampled_signal:
  [1x1 sampled_signal]
```

Note that this is for illustrative purposes only and that the

`p` and `sampled_signal` portions

of the structure contain empty signals when the constructor is called with a single input argument.

FB

`FB` is a simple MATLAB structure (as opposed to an object) that describes an OCWT filter bank using three fields: `ghatbank`, `scales`, and `name`. The first field, `ghatbank`, is a `sampled_signal` object that contains the frequency responses of the entire filter bank. The field `scales` is a vector of doubles that contains the explicit scale sample values and `name` is a string that provides a label for the wavelet family. Typically the label contains some information regarding parameters used in the construction of the wavelet function. For example, the label “Morlet40_10” indicates a Morlet wavelet with center frequency 40 and variance 10.

p

It is convenient to include a copy of the original (time-domain) signal to be wavelet-transformed as a field of the wavelet object. The field `p` in the `wavelet` object provides this accommodation and is useful in plotting and reconstruction comparisons.

sampled_signal

The raw wavelet transform values are stored in the `sampled_signal` field of the `wavelet` object.

8.5.2 `wavelet` Methods

Presented in this section are the important `wavelet` and supporting functions for the `wavelet` class. A `wavelet` method is one that takes either a `wavelet` object as an input argument or returns a `wavelet` object as an output argument. Supporting functions do not operate directly on `wavelet` objects but are related to them. Table 8.2 summarizes the main `wavelet` methods.

<i>Method</i>	<i>Description</i>
<code>wavelet</code>	Constructor.
<code>cwt</code>	Overcomplete wavelet transform.
<code>icwt</code>	Inverse overcomplete wavelet transform.
<code>plot</code>	View wavelet transform.
<code>TF_bandwidth</code>	Compute filter bank bandwidths.
<code>center_freqs</code>	Compute filter bank center frequencies.

TABLE 8.2. Summary of the main `wavelet` methods.

wavelet

The `wavelet` constructor takes several different possible sets of input arguments and returns a `wavelet` object with the fields of the object set appropriately. These input choices are shown in the following.

```
% function W = wavelet(in1,in2,in3)
%
% Constructor for wavelet class
%
% wavelet()
% wavelet(W)
% wavelet(FB)
% wavelet(FB,Wp)
% wavelet(FB,Wp,p)
```

cwt

The method `cwt` is the principal wavelet method since it performs the over-complete wavelet transformation. It takes two arguments, a `sampled_signal`

and a filter bank, and returns its overcomplete wavelet transform in a `wavelet` object.

Filter banks are most easily specified by using the “uibank” user interface in conjunction with “GetFB” for retrieving it.

```
% function W = cwt(p,FB)
%
% "Continuous" or Overcomplete Wavelet Transform
%
% <wavelet> = icwt(<sampled_signal>,[<struct filter_bank>])
%
% INPUT:
%     p    = inverse cwt of W <sampled_signal>
%     FB   = filter bank structure with fields
%
%         ghatbank = wavelet filter bank <sampled_signal>
%         scales   = associated scale (dilation) values <double>
%         name     = name of wavelet family
%
% OUTPUT:
%     W    = wavelet object
%
% See Also:
%   icwt, uibank, GetFB, plot
```

Applying this method to an upshifted by 10MHz version `pu` of the chirp and packet sampled signals in `p` of the previous section yields a 2×1 `wavelet` object as indicated in the following. The actual transform may be seen using the `plot` method discussed in the next section and is plotted in Figure 8.21.

```
>> pu=upshift(p,10); % upshift signals by 10MHz so their frequency
>>                                % support is covered by the filter bank
>>
>> W=cwt(p,GetFB);    % perform the OCWT
Computing cwt...
...signal 1 of 2
...signal 2 of 2
...done computing cwt.
>>
>> plot(abs(W))      % display the result
```

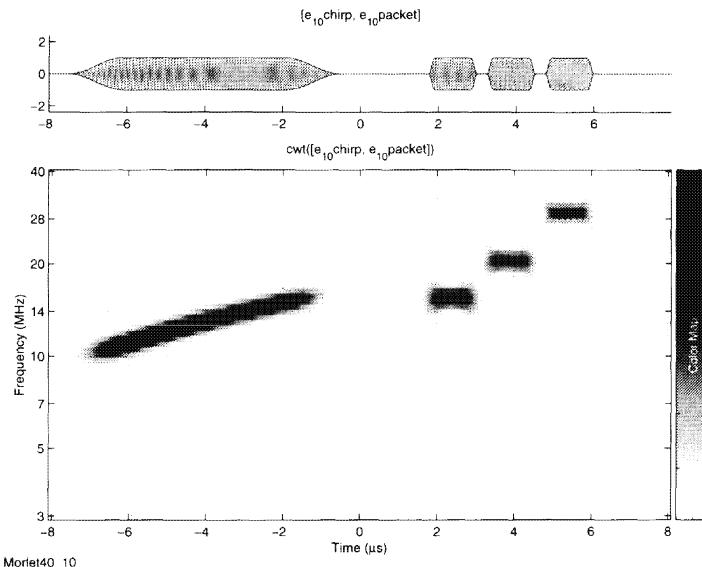


FIGURE 8.21. Wavelet transform magnitude of the chirp and packet signals.

plot

This method is used to view different aspects of a wavelet object and is dependent on the options with which it is called. By itself `plot(W)` (where `W` is a wavelet object) plots the magnitude wavelet transform⁵ and its time domain signal in the current figure (see Figure 8.24 on page 307).

```
% function plot(W,opt)
%
% graphically display a wavelet object
%
% opt:
%
%   'wt'    wavelet transform default
%
%   'ui'    plot with ui controls and menus
%
%   'menu'  plot with only menus
%
%   'bank' show the magnitude Fourier transforms of the
```

⁵Since the wavelet transform is in general a complex-valued function `plot(W)` will give a warning that indicates it is displaying the magnitude only. To avoid the warning use `plot(abs(W))`.

```
%      entire wavelet filter bank (WFB)
%
% 'G'    show the support function of the WFB
%
```

The command

```
>> plot(W,'bank')
```

plots the magnitude of the frequency transfer functions of the filter bank. This is shown in Figure 8.22.

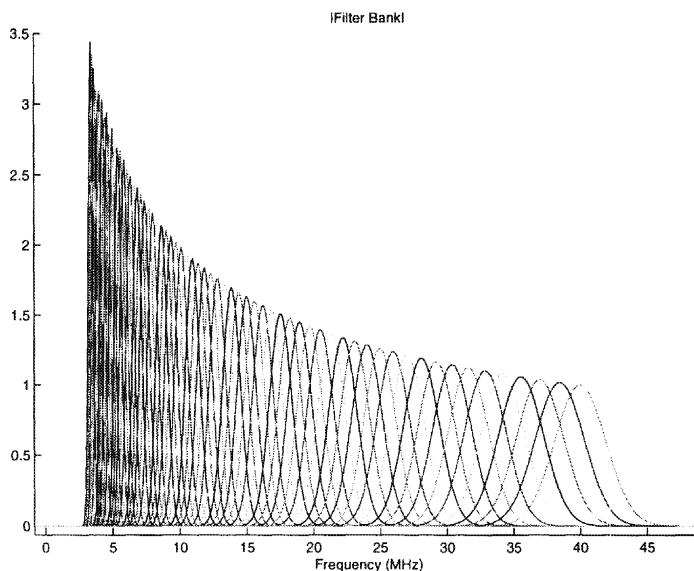


FIGURE 8.22. Magnitude of the Morlet filter bank transfer functions.

TF_bandwidth

This method returns two vectors containing the time and frequency ϵ -bandwidths, respectively, for each filter in the bank of the wavelet object *W*.

```
% function [TW,BW]=TF_bandwidth(W,epsilon)
%
% [<double>,<double>] = TF_bandwidth(<wavelet>,<scalar>);
```

```
%  
% compute the epsilon time bandwidth and  
% frequency bandwidth of wavelet filter bank W.FB
```

center_freqs

This method returns a vector of the “center” frequencies of each frequency transfer function in the wavelet filter bank of the wavelet object *W*.

```
% function CFs=center_freqs(W)  
%  
% compute wavelet filter bank center (really max) frequencies  
%  
% uses simple argmax estimate on frequency  
% domain transfer functions
```

icwt

This method performs the inverse OCWT. It takes a **wavelet** object as its sole argument and returns a **sampled_signal** object.

```
% function f = icwt(W)  
%  
% Inverse "Continuous" or Overcomplete Wavelet Transform  
%  
% <sampled_signal> = icwt(<wavelet>)  
%  
% See Also:  
% cwt, uibank, GetFB
```

The next section illustrates the use of **icwt**.

8.6 Processing Example

In this section, wavelet processing using the methods discussed previously is illustrated.

8.6.1 Forward OCWT

First, it is necessary to obtain a signal (**sampled_signal** object) to process. Fortunately, the constructor for the **sampled_signal** class supports a number of predefined signals of which the signal “chirp” is one. The following MATLAB session creates a chirp signal *p*, plots it, starts the wavelet

interface, creates a Morlet filter bank (with parameters rate = 128, $T = 4$, $a_0 = 1.04$, #filters = 64), computes the wavelet transform, and displays it.

```
>> p = sampled_signal('chirp',[1/128 4]) % make a chirp
p <<sampled_signal>> =

    sig: [1024x1 double]
    d: [0.0078 4]
    d0: 0
    dy: []
    dy0: []
    SNR: Inf
    interp: '*cubic'
    xlabel: 'Time (\mu s)'
    ylabel: []
    buttondown: 'mouse_limits'
    grid: 'off'
    box: 'off'
    colormap: 'hot'
    name: 'chirp'
    nametrack: 'on'

>> plot(p) % see the chirp
>>
>> uibank % run the wavelet interface and build a wavelet
>> % Select the Morlet family
>> % Hit the Morlet button in the interface
>>
>> GetFB % examine the filter bank just made

ans =

    ghatbank: [1x1 sampled_signal]
    scales: [1x64 double      ]
    name: 'Morlet40_10'

>> W = cwt(p,GetFB); % compute the OCWT
Computing cwt...
...signal 1 of 1
...done computing cwt.
>>
>> pop 2          % pop open a new figure
>>
>> plot(abs(W)) % display the OCWT of the chirp signal
```

Note that the analysis filter bank covers only the frequency range of

about 3 to 40MHz. The chirp starts at zero frequency so that a good portion of the leading edge of the chirp signal is cut off in the wavelet domain. To see the entire chirp, the signal may first be upshifted in frequency before applying the wavelet transform. An upshifted wavelet transform is computed next.

```
>> p = upshift(p,10); % upshift p by 10MHz in the frequency domain
>>
>> W = cwt(p,FB); % compute the upshifted OCWT
Computing cwt...
...signal 1 of 1
...done computing cwt.
>>
>> pop 3           % pop open a third figure
>>
>> plot(abs(W))   % display the OCWT of the upshifted chirp signal
```

Figures 8.23 and 8.24 display the results of the plotting commands in the preceding MATLAB session.

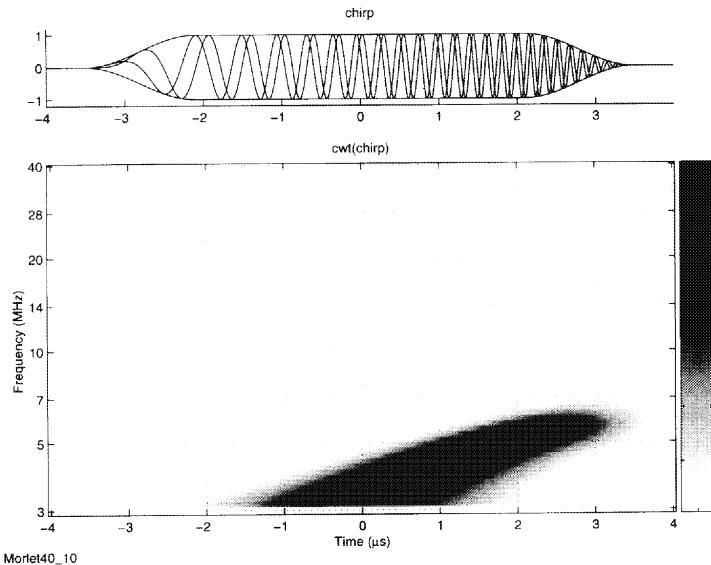


FIGURE 8.23. Wavelet transform of the chirp signal p .

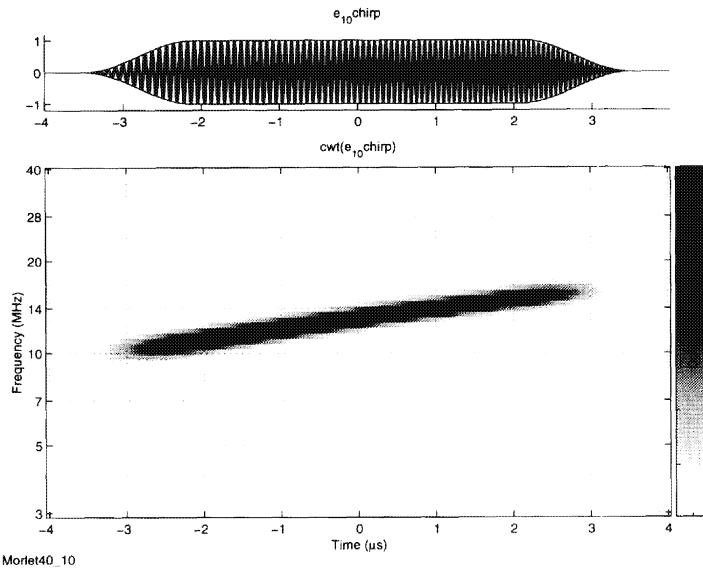


FIGURE 8.24. Wavelet transform of upshifted by 10 chirp signal $e_{10,p}$.

8.6.2 Inverse OCWT

Now consider the inverse (OCWT) wavelet transform. If the inverse is applied to the preceding chirp signal p , all that can be recovered are those frequencies over which the filter bank is supported. In the case of the Morlet bank under consideration, that range is roughly 3 to 40MHz. Note that the chirp signal, due the shape of its envelope, does not have all its energy contained in this frequency range. Thus, reconstruction of the chirp signal from the Morlet OCWT filter bank will not be able to reproduce those frequencies outside the support of the filter bank. Such a reconstruction cannot be perfect and will necessarily have some significant error.

This situation is depicted in Figure 8.25 where three axes are plotted (from top to bottom): the original chirp signal, the reconstructed chirp signal, and the error signal. Note that the error signal has a magnitude on the order of 10^{-4} . This relatively large error (as compared to machine precision of $\epsilon = 2.2 \times 10^{-16}$) is due to the energy in the analyzed signal outside the support of the filter bank.

To illustrate this, one may first force the analyzed signal to share the support of the filter bank by applying an appropriate bandpass filter. Taking this tack, the wavelet transform of a bandpassed (over the range 3 to 40MHz) version of the chirp is inverse wavelet-transformed. As shown in Figure 8.26 this leads to an instantaneous error between the original signal and its reconstruction that is well below 10^{-13} .

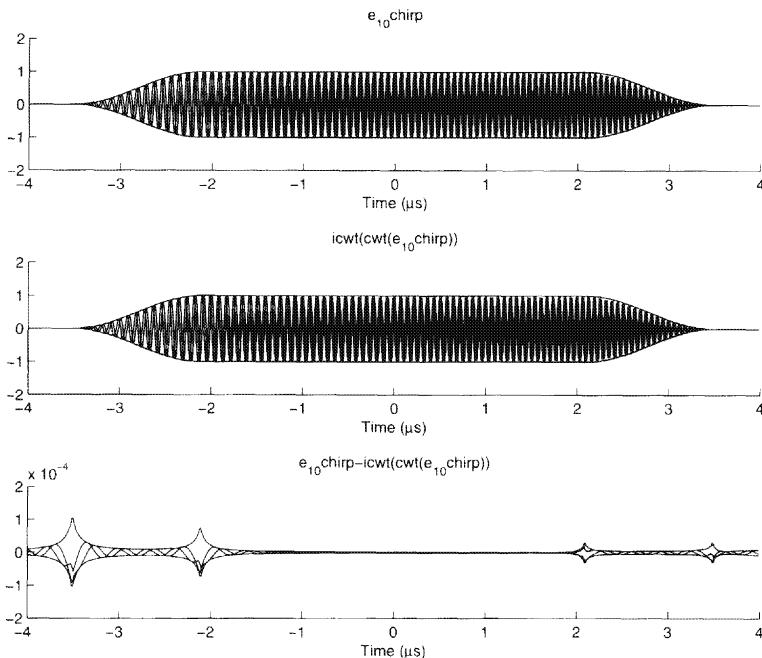


FIGURE 8.25. Reconstruction of chirp signal having significant energy outside the filter bank frequency support; error is on the order of 10^{-4} .

8.7 Supporting Functions and Globals

8.7.1 Frequency and Time Units

Although the basic units of time are seconds (s) and the basic units of frequency are Hertz (Hz), it is convenient to use scaled versions of these units that depend on a particular application. For example, in speech processing applications it is more convenient to use milliseconds (ms) and kiloHertz (KHz) as units.

These unit labels are stored in the global variables

`TimeLabel` and `FreqLabel`.

To change the value of the variables, the user can utilize commands such as:

```
SetGlobal('TimeLabel','Time (\mu s)')
SetGlobal('FreqLabel','Frequency (MHz)')
```

To change them automatically at startup, the user can place these com-

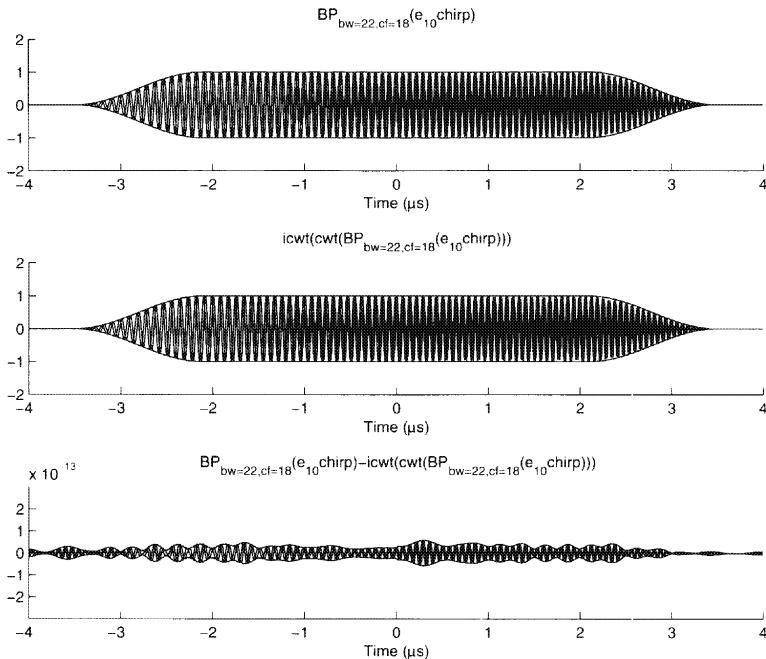


FIGURE 8.26. Reconstruction of filtered chirp signal having almost all of its energy inside the filter bank frequency support; error drops to below 10^{-13} .

mands in the file `startup.m`.

8.7.2 Graphical

`mouse_limits`

This function is the default *buttondown* field for sampled signals and allows the user to control the signal's axis limits and properties using the mouse.

```
% function mouse_limits
%
% control axis limits via mouse
%
% NO PLOTS SELECTED:
%
% WITHIN PLOT RECTANGLE:
%
% LEFT button drag zooms axes to rectangle
%
% LEFT button click selects the plot
```

```
% MIDDLE button click enlarges the plot limits
% RIGHT button click auto-sizes the plot limits
%
% OUTSIDE PLOT RECTANGLE (along x or y axis):
%
% LEFT button click x or y axis
% MIDDLE toggles x or y axis scale between log and linear
% RIGHT toggles x or y axis scale between log and linear
%
% PLOT SELECTED
%
% ANY BUTTON causes next plot selection to take the
% x and/or y limits of the selected plot
```

setplot

This function is used to create an axis on a grid in an analogous fashion to the MATLAB function `subplot`. The difference is that `setplot` takes into account the location of any uicontrols that may be present in a figure. In the case where uicontrols are present, `setplot` uses the largest available rectangle in which to make axes.

```
% function ret=setplot(in1,in2,in3,in4)
%
% Create an axis at the given subplot address
% within the largest rectangle not containing a
% a ui control in the current figure
%
% To create an axis at the (i,j) location in an
% m x n grid of subplots use:
%
% setplot(<subplot_address>
%
% <subplot_address> = [ m n i j ]      = m,n,i,j
%                      = [ m n (i-1)*n+j ] = m,n,k
%
% Overlapping axis are deleted
%
% Examples
%
% setplot([2 3 2 1]) % creates axis in lower left hand
%                           corner of 2x3 grid
% setplot(2,3,2,1)    % same as above
% setplot(2,3,4)      % same as above
%
% See Also: subplot
```

8.7.3 Wavelet

uibank

This function creates a graphical user interface for the specification of an analyzing wavelet and wavelet filter bank. Its functionality is discussed in Section 8.4.2.

```
% function fig=uibank
%
% Starts the wavelet filter bank interface
%
% Interface supports multiple families of
% wavelet functions:
%
%      Morlet
%      Parameteric Band Limited (PBL)
%      B-Spline
%      Daub
```

GetFB

```
% function FB=GetFB
%
% Gets the filter bank specified in the wavelet
% interface and returns it in the structure FB:
%
%      FB.ghatbank [sampled_signal (FB transfer functions)]
%      FB.scales    [double vector (scales sample values)]
%      FB.name      [char id]
%
% See Also uibank.m
```

MakeFB

```
% function FB = makeFB(function,params, a0,nfilters, ...
%                               gd, normalization)
%
% Makes a filter bank for performing the OCWT
%
% INPUT:
%
%      function = string containing name of 1-D function to compute g,
%                  e.g., 'Morlet' or 'BSpline'
```

```
% params = vector of parameters for the above function;
%           function call has the form "function(tt,params)"
%           where tt are the time samples at which to evaluate
%
% a0      = dilation constant for exponential scale (frequency)
%           sampling
%
% nfilters = number of filters to make
%
% gd       = sample structure for the impulse response functions
%           in the filter bank [ [1/128 4] ]
%
% normalization = {'L2','L1','none'}
%
% OUTPUT
%
% FB = filter bank structure with fields
%
%     ghatbank = wavelet filter bank transfer functions
%                 (frequency domain) <sampled_signal>
%     scales   = associated scale (dilation) values <double>
%     norm     = {'L2','L1','none'}
%     name     = name of wavelet family <char>
%
% Example:
%
% FB = makeFB('Morlet',[40 10], 1.06,32, [1/128 4]);
% plot(wavelet(FB),'bank') % display the filter bank
```

References

- [AA95] P. Abry and A. Aldroubi, Designing multiresolution analysis-type wavelets and their fast algorithms, *J. Fourier Anal. Appl.*, 2:135–159, 1995.
- [AF97] A. Aldroubi and H. Feichtinger, Complete iterative construction algorithms for irregularly sampled data in spline-like spaces, Proc. IEEE ICASSP, 1997.
- [Ald96] A. Aldroubi, The wavelet transform: A surfing guide, In A. Aldroubi and M. Unser, editors, *Wavelets in Medicine and Biology*, pages 3–36, CRC Press, Boca Raton, FL, 1996.
- [Ald97] A. Aldroubi, Oblique and hierarchical multiwavelet bases, *To appear in Applied and Comp. Harmonic Analysis*, 1997.
- [All85] J. Allen, Cochlear modeling, *IEEE ASSP Magazine*, 2:3–29, 1985.
- [Alt60] M. Altman, An optimum cubically convergent iterative method of inverting a bounded linear operator in Hilbert space, *Pacific J. Math.*, 10:1107–1114, 1960.
- [AS96] A.N. Akansu and M.J.T. Smith, *Subband and Wavelet Transforms*, Kluwer Academic Publishers, Boston, 1996.
- [AU93] A. Aldroubi and M. Unser, Families of multiresolution and wavelet spaces with optimal properties, *Numer. Funct. Anal. Optimiz.*, 14:417–446, 1993.
- [AU94] A. Aldroubi and M. Unser, Sampling procedures in function spaces and asymptotic equivalence with Shannon’s sampling theory, *Numer. Funct. Anal. Optimiz.*, 15:1–21, 1994.
- [AU96] A. Aldroubi and M. Unser, *Wavelets in Medicine and Biology*, CRC Press, Boca Raton, FL, 1996.

- [Ben76] J. J. Benedetto, *Real Variable and Integration*, Teubner, Stuttgart, 1976.
- [Ben92] J. J. Benedetto, Irregular sampling and frames, In C. Chui, editor, *Wavelets: A Tutorial in Theory and Applications*, pages 445–507, Academic Press, Boston, MA, 1992.
- [Ben93] J. J. Benedetto, Frame decompositions, sampling and uncertainty principle inequalities, In J. J. Benedetto and M. Frazier, editors, *Wavelets: Mathematics and Applications*, CRC Press, Boca Raton, FL, 1993.
- [Ben96] J. J. Benedetto, *Harmonic Analysis and Applications*, CRC Press, Boca Raton, FL, 1996.
- [Ber71] T. Berger, *Rate Distortion Theory, A Mathematical Basis for Data Compression*, Prentice-Hall, Englewood Cliffs, NJ, 1971.
- [BH90] J. J. Benedetto and W. Heller, Irregular sampling and the theory of frames, *Mat. note 10 suppl. n. 1*, pages 103–125, 1990.
- [BHW92] J. J. Benedetto, C. Heil, and D. Walnut, Uncertainty principles for time-frequency operators, Technical report, MITRE Corp., McLean, VA, 1992.
- [BMG92] A. Baskurt, I. E. Magnin, and R. Goutte, Adaptive discrete cosine transform coding algorithm for digital mammography, *Optical Engineering*, 31:1922–1928, Sept. 1992.
- [BO94] B. Boashash and P. O'Shea, Polynomial Wigner-Ville distributions and their relationship to time-varying higher order spectra, *To appear in IEEE Trans. Signal Processing*, Jan. 1994.
- [BT92] J. J. Benedetto and A. Teolis, An auditory motivated time-scale signal representation, *IEEE-SP International Symposium on Time-Frequency and Time-Scale analysis*, Oct. 1992.
- [BT93] J. J. Benedetto and A. Teolis, A wavelet auditory model and data compression, *Applied and Comp. Harmonic Analysis*, 1, 1993.
- [BW93] J. J. Benedetto and D. F. Walnut, Gabor frames for L^2 and related spaces, In J. J. Benedetto and M. W. Frazier, editors, *Wavelets: Mathematics and Applications*, pages 97–162, CRC Press, Boca Raton, FL, 1993.
- [CDF92] A. Cohen, I. Daubechies, and J. C. Feauveau, Biorthogonal bases of compactly supported wavelets, *Comm. Pure and Appl. Math.*, 45:485–560, 1992.

- [Chu92] C. K. Chui, On cardinal spline-wavelets, In M.B. Ruskai and et al., editors, *Wavelets and Their Applications*, Jones and Bartlett, Boston, 1992.
- [CKBB93] F. S. Cohen, S. Kadambé, and G. F. Boudreux-Bartels, Tracking of unknown nonstationary chirp signals using unsupervised clustering in the Wigner distribution, *IEEE Trans. Signal Processing*, 41(11):3085–3101, Nov. 1993.
- [Coh89] L. Cohen, Time-frequency distributions – a review, *Proc. IEEE*, 77(7):941–981, 1989.
- [CW92] D. Cochran and C. Wei, Scale based coding of digital communication signals, *Proceedings of the IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis*, pages 455–458, Oct. 1992.
- [Dau88] I. Daubechies, Orthonormal bases of compactly supported wavelets, *Comm. Pure Appl. Math.*, 41:909–996, 1988.
- [Dau90] I. Daubechies, The wavelet transform, time-frequency localization and signal analysis, *IEEE Trans. Information Theory*, 36(5):961–1005, 1990.
- [Dau92] I. Daubechies, *Ten Lectures on Wavelets*, CBMS-NSF, SIAM, Philadelphia, 1992.
- [DGM86] I. Daubechies, A. Grossman, and Y. Meyer, Painless nonorthogonal expansions, *J. Math. Phys.*, 27(5):1271–1283, May 1986.
- [Don93] D. L. Donoho, Unconditional bases are optimal bases for data compression and for statistical estimation, *Applied and Comp. Harmonic Analysis*, 1:100–115, 1993.
- [Don95] D. L. Donoho, De-noising by soft thresholding, *IEEE Trans. Information Theory*, 41(3):613–627, May 1995.
- [DS52] R. Duffin and S. Schaeffer, A class of nonharmonic Fourier series, *Trans. Amer. Math. Soc.*, 72:341–366, 1952.
- [Feh95] K. Feher, *Wireless Digital Communications*, Prentice-Hall, Englewood Cliffs, NJ, 1995.
- [FG89] H. G. Feichtinger and K. Gröchenig, Banach spaces related to integrable group representations and their atomic decompositions, I, *J. Functional Anal.*, 86:307–340, 1989.
- [FG92] H. G. Feichtinger and K. Gröchenig, Iterative reconstruction of multivariate band-limited functions from irregular sampling values, *SIAM J. Math Anal.*, 23(1):1–18, Jan. 1992.

- [FG93] H. G. Feichtinger and K. Gröchenig, Theory and practice of irregular sampling, In J. J. Benedetto and M. W. Frazier, editors, *Wavelets: Mathematics and Applications*, pages 305–363, CRC Press, Boca Raton, FL, 1993.
- [FGS95] H .G. Feichtinger, K. Gröchenig, and T. Strohmer, On discrete band-limited extrapolation, *Numerische Mathematik*, 69:423–440, 1995.
- [FJ85] M. Frazier and B. Jawerth, Decomposition of Besov spaces, *Indiana Univ. Math. J.*, 34(4):777–799, 1985.
- [Gab46] D. Gabor, Theory of communication, *J. IEE*, 93:429–457, 1946.
- [GG80] I. Gohberg and S. Goldberg, *Basic Operator Theory*, Birkhauser, Boston, 1980.
- [GG92] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Press, Norwell, MA, 1992.
- [Gib93] J. D. Gibson, *Principles of Digital and Analog Communications*, second edition, Macmillan, New York, 1993.
- [GM84] A. Grossman and J. Morlet, Decomposition of Hardy functions into square integrable wavelets of constant shape, *SIAM J. Math. Anal.*, 15:723–736, 1984.
- [GMP85] A. Grossman, J. Morlet, and T. Paul, Transforms associated to square integrable group representations, I, general results, *J. Math. Phys.*, 26:2473–2579, 1985.
- [Gra90] R. M. Gray, *Source Coding Theory*, Kluwer Academic Press, Norwell, MA, 1990.
- [Gro77] C.W. Groetsch, *Generalized Inverses of Linear Operators*, Marcel Dekker, Inc., New York, 1977.
- [Grö93a] K. Gröchenig, Acceleration of the frame algorithm, *IEEE Trans. Signal Processing*, 41:3331–3340, Dec. 1993.
- [Grö93b] K. Gröchenig, Irregular sampling of wavelet and short time Fourier transforms, *Constr. Approx.*, 9:283–297, 1993.
- [HC93] C. Heil and D. Colella, Dilation equations and the smoothness of compactly supported wavelets, In J. J. Benedetto and M. Frazier, editors, *Wavelets: Mathematics and Applications*, CRC Press, Boca Raton, FL, 1993.
- [Hla92] F. Hlawatsch, Regularity and unitarity of bilinear time-frequency signal representations, *IEEE Trans. Information Theory*, 38(1):82–94, Jan. 1992.

- [HW89] C. Heil and D. Walnut, Continuous and discrete wavelet transforms, *SIAM Review*, 31:628–666, 1989.
- [Jaf91] S. Jaffard, A density criterion for frames of complex exponentials, *Michigan Math. J.*, 38:339–348, 1991.
- [Kad64] M. I. Kadec, The exact value of the Paley–Wiener constant, *Soviet Math. Dokl.*, 5:559–561, 1964.
- [Kai94] G. Kaiser, *A Friendly Guide to Wavelets*, Birkhauser, Boston, 1994.
- [KFFJ92] A. Kumar, D. Fuhrmann, M. Frazier, and B. Jawerth, A new transform for time–frequency analysis, *IEEE Trans. Signal Processing*, 40(7):1697–1707, July 1992.
- [KLB92] P. J. Kootsookos, B. C. Lovell, and B. Boashash, A unified approach to the STFT, TFDs, and instantaneous frequency, *IEEE Trans. Sig. Proc.*, 40(8):1971–1982, 1992.
- [Kot33] V. A. Kotel’nikov, On the transmission capacity of ‘ether’ and wire in electro-communications, *Izd. Red. Upr. Svyazi RKKA*, 1933.
- [Lev40] N. Levinson, Gap and density theorems, *Am. Math. Soc. Colloq. Publ.*, 26, 1940.
- [LKC⁺94] R. E. Learned, H. Krim, B. Claus, A. Willsky, and W. C. Karl, Wavelet-packet-based multiple access communication, Preprint; to appear in the *Proceedings of SPIE International Symposium on Optics, Imaging, and Instrumentation* (2303-20), 1994.
- [Mal89a] S. Mallat, Multiresolution approximations and wavelet orthonormal bases of $L^2(\mathbb{R})$, *Trans. Amer. Math. Soc.*, 315:69–87, 1989.
- [Mal89b] S. Mallat, A theory for multiresolution signal decomposition: The wavelet representation, *IEEE Trans. Pattern Anal. Machine Intell.*, 11:674–693, 1989.
- [Mat97] The Mathworks, *Using Matlab Version 5*, Matlab Documentation, 1997.
- [MF85] W. Martin and P. Flandrin, Wigner-Ville spectral analysis of non-stationary processes, *IEEE Trans. Acoust. Speech and Sig. Proc.*, ASSP-33(6):1461–1470, 1985.

- [MR97] M. Malfait and D. Roose, Wavelet-based image denoising using a Markov random field a priori model, *IEEE Trans. Image Processing*, 6(4):549–565, April 1997.
- [MZ93] S. Mallat and Z. Zhang, Matching pursuits with time-frequency dictionaries, *IEEE Trans. Signal Processing*, 41:3397–3415, Dec. 1993.
- [NK93] A. Nohara and J. Kane, A noise suppression system for FM radio receiver, *IEEE Trans. Consumer Electron.*, 39:533–539, Aug. 1993.
- [OS75] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing*, Prentice-Hall, Englewood Cliffs, NJ, 1975.
- [OS92] P. A. Olsen and K. Seip, A note on irregular discrete wavelet transforms, *IEEE Trans. Information Theory*, 38(2):861–863, March 1992.
- [Pap77] A. Papoulis, *Signal Analysis*, McGraw-Hill, New York, NY, 1977.
- [Pet67] W. V. Petryshyn, On generalized inverses and uniform convergence of $(I - \beta K)^n$ with applications to iterative methods, *J. Math. Anal. Appl.*, 18:417–439, 1967.
- [PK75] R. R. Pfeiffer and D. O. Kim., Cochlear nerve fiber responses: Distribution along the cochlear partition, *J. Acoust. Soc. Am.*, 58:867–869, 1975.
- [Poo88] H. V. Poor, *An Introduction to Signal Detection and Estimation*, Springer-Verlag, New York, 1988.
- [PW34] R. Paley and N. Wiener, Fourier transforms in the complex domain, *Am. Math. Soc. Colloq. Publ.*, 19, 1934.
- [RD92] O. Rioul and P. Duhamel, Fast algorithms for discrete and continuous wavelet transforms, *IEEE Trans. Information Theory*, 38(2):569–586, March 1992.
- [Roy68] H. L. Royden, *Real Analysis*, second edition, Macmillan, New York, 1968.
- [Sad96] J. Sadowsky, Investigation of signal characteristics using the continuous wavelet transform, *Johns Hopkins APL Technical Digest*, 17(3):258–269, 1996.
- [Sha49] C. E. Shannon, Communications in the presence of noise, *Proc. IRE*, 37:10–21, Jan. 1949.

- [She92] M. J. Shensa, The discrete wavelet transform: Wedding the à trous and Mallat algorithms, *IEEE Trans. Signal Processing*, Oct. 1992.
- [Sho67] D. Showalter, Representation and computation of the pseudoinverse, *Proc. Amer. Math. Soc.*, 18:584–586, 1967.
- [SL61] D. Slepian and H. J. Landau, Prolate spheroidal wave functions, Fourier analysis and uncertainty-II, *Bell System Tech. J.*, 40:65–84, 1961.
- [Sle76] D. Slepian, On bandwidth, *Proc. of the IEEE*, 64(3):292–300, 1976.
- [SN96] G. Strang and T. Nguyen, *Wavelets and Filter Banks*, Wellesley-Cambridge Press, Wellesley, MA, 1996.
- [SP61] D. Slepian and H. O. Pollak, Prolate spheroidal wave functions, Fourier analysis and uncertainty-I, *Bell System Tech. J.*, 40:43–64, 1961.
- [Str91] T. Strohmer, *Irregular sampling, frames and psuedo-inverse*, PhD thesis, University of Vienna, 1991.
- [Str93] R. S. Strichartz, Construction of orthonormal wavelets, In J. J. Benedetto and M. Frazier, editors, *Wavelets: Mathematics and Applications*, CRC Press, Boca Raton, FL, 1993.
- [TB93] A. Teolis and J. J. Benedetto, Local frames, *SPIE Math. Imaging*, 2034:310–321, 1993.
- [TB94a] A. Teolis and J. Baras, Wavelet and other transforms for specific emitter identification, AIMS TR-94-08 (SECRET), AIMS, Inc., Aug. 1994.
- [TB94b] A. Teolis and J. J. Benedetto, Noise suppression using a wavelet model, *ICASSP*, 1994.
- [TB95] A. Teolis and J. J. Benedetto, Local frames and noise reduction, *Signal Processing*, 45:369–387, 1995.
- [Teo93] A. Teolis, *Discrete representation of signals from infinite dimensional Hilbert spaces with application to noise suppression and compression*, PhD thesis, University of Maryland, College Park, MD, 1993.
- [Tre76] S. A. Tretter, *Introduction to Discrete-Time Signal Processing*, Wiley, New York, 1976.

- [Tri95] H. Triebel, *Interpolation Theory, Function Spaces, Differential Operators*, Johann Ambrosius Barth Verlag, Heidelberg, 1995.
- [VK95] M. Vetterli and J. Kovacević, *Wavelets and Subband Coding*, Prentice-Hall, Englewood Cliffs, NJ, 1995.
- [Wal94] G. G. Walter, *Wavelets and Other Orthogonal Systems with Applications*, CRC Press, Boca Raton, FL, 1994.
- [Whi15] E. T. Whittaker, On the functions which are represented by the expansion of interpolating theory, *Proc. Roy. Soc. Edinburgh*, 35:181–194, 1915.
- [XKZ94] X. G. Xia, C. J. Kuo, and Z. Zhang, Wavelet coefficient computation with optimal prefiltering, *IEEE Trans. Signal Processing*, 42:2191–2197, Aug. 1994.
- [You80] R. M. Young, *An Introduction to Nonharmonic Fourier Series*, Academic Press, New York, 1980.
- [You93] R. K. Young, *Wavelet Theory and Its Applications*, Kluwer Academic, Norwell, MA, 1993.
- [YWS92] X. Yang, K. Wang, and S. Shamma, Auditory representations of acoustic signals, *IEEE Trans. Information Theory*, 38(2):824–839, March 1992.

Index

- atomic decomposition, 29
- atoms, 29
- basis
 - biorthogonal, 34
 - bounded unconditional, 17
 - definition, 17
 - minimal, 32
 - orthonormal, 17
 - equivalences, 32
 - for $L^2(-T/2, T/2)$, 17
 - harmonic complex exponentials, 33
 - reconstruction, 31
 - wavelet, 89, 109
 - Riesz, 34
 - equivalences, 45
 - Kadec–Levinson, 36
 - reconstruction, 35
 - Schauder, 16
 - standard, 33
 - unconditional, 17
- bit rate, 213
- Cauchy–Schwarz, 9
- classical sampling theorem, 23, 32
- coherence
 - approximation error, 188
 - approximation theorem, 188
 - description, 173, 178
 - distribution, 180
 - functional, 180
 - measure, 181
 - properties, 180
- time–frequency, 182
- compact set, 11
- compression, 212, 216
- convolution
 - for a group, 82
 - in $\ell^2(\mathbb{Z})$, 22
 - in $L^2(\mathbb{R})$, 21
- Duffin–Schaeffer theorem, 158
- dyadic sampling lattice, 90
- FM signal, 150
- Fourier transform
 - continuous, 10, 18
 - continuous-time periodic, 19
 - discrete, 20
 - discrete time, 20
 - dual spaces, 20
- frame
 - bounds, 37, 48
 - correlation
 - definition, 42
 - operator, 41
 - properties, 43
 - definition, 37
 - dual, 50
 - exact, 38
 - irregular sampling, 156
 - local, 185
 - operator, 38
 - overcomplete, 132, 206
 - advantages, 134
 - drawbacks, 134
 - pseudo-inverse, 47, 207

- reconstruction, 178
 - iterative algorithm, 52, 209
- representation, 30
 - adjoint, 36, 40
 - noise robustness, 42
 - operator, 37, 40
 - range, 31
- thresholded
 - reconstruction error, 190
 - representation, 179
- tight, 38
- wavelet, 160
- function
 - analytic, 74
 - characteristic, 7
 - complex exponential
 - nonharmonic, 158
 - complex exponentials
 - harmonic, 17
 - nonharmonic, 36
 - Dirac delta, 8
 - Dirichlet kernel, 23
 - Kronecker delta, 7
 - periodic, 18
 - quantization, 215
 - sinc, 23
- Gabor transform
 - definition, 78
 - inverse, 80
- group
 - affine, 80, 81
 - convolution operator for, 82
 - definition, 80
 - involution operator for, 82
 - measure on, 82
 - Parseval for, 85
 - Plancherel for, 85
 - translation operator for, 82
 - Weyl-Heisenberg, 80, 81
- group representation transform, 80, 84
- inner product, 8
- irregular sampling, 156, 163
- multiresolution analysis
 - definition, 94
 - discrete and continuous time connections, 96
 - Haar, 100
 - projection, 95
 - scale relationships, 97
 - scaling function, 92, 107
- Neumann expansion, 50
- noise
 - ratio of signal to (SNR), 210
 - representation, 206
 - sources, 172
 - suppression, 171, 176, 192
- norm, 8
- object classes
 - sampled_signal, 268
 - wavelet, 268
- object-oriented, 267
- operator
 - adjoint, 14
 - Bessel, 30
 - bijective, 14
 - bounded linear, 13
 - norm, 13
 - self-adjoint, 14
 - compact, 14
 - continuous, 14
 - dilation, 15
 - down-sampling, 117, 215
 - frame, 36, 37
 - injective, 14
 - inverse, 14
 - involution, 15
 - isometry, 15
 - kernel, 14
 - mask, 177
 - modulation, 15
 - orthogonal projection, 14
 - range, 14
 - reflection, 15
 - surjective, 14
 - threshold, 174, 177

- topological isomorphism, 15
- translation, 15
- unitary, 15
- up-sampling, 117, 215
- zero fill, 215
- overcomplete wavelet transform
 - analysis bank, 137
 - definition, 130
 - implementation, 295
 - inverse, 142
 - synthesis bank, 138
- Parseval
 - dual space, 21
 - for $L^2(\mathbb{R})$, 18
 - for orthonormal basis, 31
- phi-transform, 129
- Plancherel
 - dual space, 20
 - for $L^2(\mathbb{R})$, 18
 - for orthonormal basis, 31
- Poisson Summation Formula, 19
- Pythagorean Theorem, 9
- quantization, 24, 214, 215
- rate-distortion, 213
- sampled_signal methods
 - add_noise, 284
 - align, 287
 - arithmetic operators
 - +, -, .*, ./, 275
 - bandpass, 288
 - constructor, 271
 - conv, 279
 - der, 289
 - dilate, 282
 - fft, 278
 - int, 289
 - involute, 282
 - join, 285
 - mean, 290
 - normalize, 294
 - plot, 273
 - powerdB, 292
- relational operators
 - $=, \neq, <, \leq, >, \geq$, 276
- resample, 279
- sigcat, 283
- split, 286
- support, 294
- threshold, 283
- translate, 281
- upshift, 291
- sampling
 - down-, 117
 - semilog-regular, 131, 142
 - time-scale, 136
 - uniform density, 158
 - uniformly dense, 158
 - up-, 117
- scale, 60
- sequence
 - Cauchy, 9
 - convergence, 9
 - Kadec–Levinson, 159
 - orthonormal, 30
- set of functions
 - biorthogonal, 17
 - complete, 16
 - dense, 16
 - orthonormal, 16
 - span of, 16
- signal-to-noise ratio (SNR), 210
- space
 - Banach, 9
 - Hilbert, 9, 81, 83
 - Paley–Wiener, 84
- supporting methods
 - GetFB, 311
 - MakeFB, 311
 - mouse_limits, 309
 - setplot, 310
 - uibank, 296, 311
 - unit labels, 308
- symbols
 - \mathbb{C} , 7
 - $H^2_-(\mathbb{R})$, 12
 - $H^2_+(\mathbb{R})$, 12
 - $\ell^2(J)$, 10

- $\ell^2(\mathbb{Z})$, 10, 12
- $L^2(\mathbb{R})$, 10, 11
- $L^2(-T/2, T/2)$, 11
- $L^\infty(\mathbb{R})$, 10
- $L^1(\mathbb{R})$, 10
- PW_Ω , 10, 12
- \mathbb{R} , 7
- \mathbb{R}^- , 7
- \mathbb{R}^+ , 7
- supp, 7
- TF(\mathbb{R}), 11
- \mathbb{Z} , 7
- uncertainty principle, 146, 148
- variance, 146
- wavelet
 - admissible, 69, 77
 - analyzing, 60
 - PBL, 149
 - analyzing functions
 - B-spline, 64
 - Daubechies, 111
 - Haar, 62
 - Morlet, 66
 - Shannon, 63
 - compactly supported ONB, 89
 - constructor, 300
- continuous transform, 67
 - properties, 69
- Daubechies, 109
 - examples, 112
- design, 145
- discrete transform, 90
- family, 60
- fast wavelet transform, 91, 116
 - examples, 120
- function, 60
- inverse transform
 - for $H^2(\mathbb{R})$, 76
 - for $H^2_+(\mathbb{R})$, 75
 - for $L^2(\mathbb{R})$, 73
- orthonormal basis
 - compactly supported, 109
 - construction, 104, 105
- wavelet methods
 - center_freqs, 304
 - cwt, 300
 - icwt, 304
 - plot, 302
 - TF_bandwidth, 303
- WSPW
 - description, 264
 - signal interface, 265
- Z-transform, 22