The Construction and Application of Wavelets in Numerical Analysis

Wim Sweldens

May 18, 1995

The Construction and Application of Wavelets in Numerical Analysis

Abstract

Wavelets are a new family of basis functions that can be used to approximate general functions. They combine powerful properties such as (bi)orthogonality, compact support, localization in time and frequency, and fast algorithms.

This thesis investigates the use of wavelets in numerical analysis problems. In the first part we construct two basic tools, quadrature formulae and asymptotic error expansions. The former provides an easy way to calculate the wavelet coefficients, while the latter allows a simple comparison of different wavelet families.

In the second part, we construct and study wavelets adapted to a weighted inner product. We show how one can use those wavelets for the rapid solution of ordinary differential equations.

Finally, we study smooth local trigonometric functions, which can be seen as the Fourier transform of wavelets. We generalize their construction to the biorthogonal case, and show how to use them in data compression algorithms. This is illustrated with examples concerning image compression.

Keywords. wavelet, multiresolution analysis, quadrature formula, error expansion, weighted inner product, ordinary differential equations, local trigonometric functions, compression.

AMS subject classification. 41A30, 42A10, 42C05, 65B05, 65D32, 65L60.

Acknowledgements

This thesis would never have been written without the help and support of numerous people. Needless to say I thank all of them and will always bear their efforts in my heart. Therefore, I would like to include the following short story.

When I started as a doctoral student, now more than three and a half years ago, Prof. Robert Piessens agreed to be my advisor; ever since, he has been supportive of my actions. He has allowed me wide freedom in my choice of research area and location at which to study it. With his help, I was able to expand my horizon by frequently attending international conferences.

I joined the "parallelgroep", a research group focused on high performance scientific computing. Three of its members, Prof. Dirk Roose, Danny Goovaerts, and Stefan Vandewalle, were the first to draw my attention to wavelets. This was the starting point of the research that lead to this thesis. Since then, they were always available with advise or comfort. This research group, whose other members include Lutgarde Beernaert, Johan De Keyser, Hugo Embrechts, Jan Janssen, Kurt Lust, Maurits Malfait, Karl Meerbergen, Raf Van Driessche, and Werner Verhoeven, formed a stimulating and pleasant environment. Curiously, it turned out that the only page of this thesis containing the word *parallel*, is this very one.

Discussions with Hugo Embrechts and Pierre Verlinden lead to my first results. It is with great pleasure that I remember their enthusiasm. Pierre showed me more elegant proofs of some results.

A little later, I met Prof. Björn Jawerth and he agreed to take me under his wing. By no means I can describe how much I learned from him. His energy, encouragement, and enthusiasm are inexhaustible, and he constantly kept me motivated. His typical response after being shown a new idea was "Have you started typing yet ?" Keke, Lolo, and Niki Jawerth made me feel at home in Columbia.

At the University of South Carolina, I became part of a "wavelet" research group. The people I had the pleasure of working with include, Baiqiao Deng, Yi Liu, Marius and Dorina Mitrea, Kay Nyström, Gunnar Peters, Arun Tirumalai, Shannaz Vatani, Qun Wu, and Kenny Yarnall. We often organized informal seminars, in which we discussed new material. These turned out to be an ideal source of inspiration.

Next to traveling between the US and Belgium, I got the opportunity to visit

several institutions and meet other researchers. The list includes Mary Ellen Bock (Purdue University), Charles Chui (Texas A&M University), Ingrid Daubechies (AT&T Bell Labs and Rutgers University), Dave Donoho and Gene Golub (Stanford University), Henk Heijmans and Tom Koornwinder (Centrum voor Wiskunde en Informatica), Chris Heil and Gil Strang (MIT), C. K. Majumdar (S.N. Bose Institute of Basic Sciences, Calcutta), Cleve Moler (The Mathworks), Bao Gia Nguyen (Illinois Institute of Technology), and Louise Raphael (Howard University). These visits gave my the opportunity to present my work and be exposed to the current results of other researchers. This is vital in a rapidly evolving field and it helped me to gain a broader understanding. Discussions with Mary Ellen Bock and Ingrid Daubechies lead to some of the results of Chapter 3, and a suggestion of David Donoho was the starting point of the results in Chapter 5.

The story would not be complete without mentioning my long time best friend Maurits Malfait. Ours path have been running extremely parallel (sic) over the last twenty years. Maurits's continuous support, stimulation, humor, but also (well-deserved) criticism have always been an indispensable source of happiness for me.

During the last year I worked with Geert Uytterhoeven in the context of his engineering thesis. Although our contacts were primarily over e-mail, I enjoyed his eagerness, dedication and careful work.

Writing a text in a foreign language is not an easy task, and I would have never been able to accomplish it without the professional help of Kathy Borg-Todd. She spent a great part of her Christmas break carefully proofreading the manuscript. Evidently, I take blame for every error that remains.

During the course of this thesis, I was employed as a research assistant of the Belgian National Fund of Scientific Research (NFWO), and as a research fellow at the University of South Carolina. These positions allowed me great freedom and flexibility, a true luxury for a researcher.

The members of the reading committee, Prof. Adhemar Bultheel, Prof. Björn Jawerth, Prof. Robert Piessens, and Prof. Walter Van Assche, did a thorough job reading and commenting on early drafts, often in minimal time. Prof. Paul Dierckx graciously agreed to join them as a member of the jury.

Last but not least, my family and especially my parents have always been extremely supportive during the whole course of my studies. My parents took care of all practical arrangements associated with finishing a thesis.

The encouragement of close friends was needed at many times. They were always available when I needed them.

Dank U wel – Thank you.
Wim,
Columbia, March 4, 94.

"I humbly thank you; well, well, well.
(Hamlet to Ophelia)"
—Shakespeare, *Hamlet (1623).*

Contents

Notations

List of symbols

Symbol	: Description	Page
$\widehat{f}(\omega)$	: Fourier transform of the function $f(x)$	4
$\lvert x \rvert$	: modulus of a complex number	2
$\lfloor x \rfloor$	: integer part of a real number	93
$\overline{x}$	: complex conjugate of a complex number	2
$\lVert f \rVert$	: general norm	1
$\lVert f \rVert_w$	: weighted norm	124
$\lVert f \rVert_p$	: L_p norm	1
$\lvert f \rvert_{k,p}$	: Sobolev seminorm	1
$\langle f, g \rangle$	: inner product of the functions $f(x)$ and $g(x)$	2
$\langle f, g \rangle_w$	: weighted inner product of $f(x)$ and $g(x)$	124
$\langle\langle f, g \rangle\rangle$	: operator inner product of $f(x)$ and $g(x)$	144
$*$	: convolution of two functions or sequences	16
$\oplus$	: direct sum of two vector spaces	1
$\otimes$	: tensor product of two functions	38
$b_I(x)$	: bell function associated with interval I	165
B_n	: n-th Bernoulli number	102
$B_n(x)$	: Bernoulli polynomial of degree n	92
$\mathbf{B}_n(x)$	: Bernoulli periodic spline of degree n	93
$\mathcal{C}^k$	: space of k-times continuously differentiable functions	3
$\mathcal{C}^k(S)$	: k-times continuously differentiable functions on the set S	3
clos	: closure of a set	3
$d(x)$	: difference of the left and right cutoff functions	176
$\widetilde{d}(x)$	: difference of the dual left and right cutoff functions	177
$\mathcal{D}$	: odd mapping operator	175
$\mathcal{D}^k$	: space of bounded functions with polynomial decay	3
$e(x)$	: sum of the left and right cutoff functions	176
$\widetilde{e}(x)$	: sum of the dual left and right cutoff functions	177
$e_k(x)$	: k-th complex exponential	84

$E_n(x)$	: Euler polynomial of degree n	92
$\mathbf{E}_n(x)$	: Euler periodic spline of degree n	93
$\mathcal{E}$	: even mapping operator	175
$\mathcal{E}_j$	: error operator of the wavelet approximation	79
$\mathcal{E}'_j$	: approximated error operator	104
$F_r[f(x)]$	: error of the quadrature formula $Q_r[f(x)]$	62
$\mathcal{F}_\alpha$	: folding operator around α	162
$\widetilde{\mathcal{F}}_\alpha$	: dual folding operator around α	170
g_k	: coefficients of the wavelet refinement equation	17
$\widetilde{g}_k$	: coefficients of the dual wavelet refinement equation	24
$g(\omega)$	: symbol associated with the wavelet	18
$\widetilde{g}(\omega)$	: symbol associated with the dual wavelet	22
$G(\omega)$	: symbol associated with the operator wavelet	147
$\mathcal{G}_I$	: folding operator associated with interval I	165
$\widetilde{\mathcal{G}}_I$	: dual folding operator associated with interval I	171
h	: grid size ($= 2^{-j}$ on level j)	25
h_k	: coefficients of the refinement equation	15
$\widetilde{h}_k$	: coefficients of the dual refinement equation	22
$h(\omega)$	: symbol associated with the scaling function	16
$\widetilde{h}(\omega)$	: symbol associated with the dual scaling function	22
$H(\omega)$	: symbol associated with the operator scaling function	147
I	: general interval	125
j	: multiresolution level	14
$l_\alpha(x)$	: left cutoff function	162
$\widetilde{l_\alpha}(x)$	: dual left cutoff function	170
$\mathcal{L}$	: differential operator	143
Lip^α	: space of Lipschitz continuous functions	3
L_p, $\mathrm{L}_p(\mathbf{R})$	: space of p-th power integrable functions	2
L_2, $\mathrm{L}_2(\mathbf{R})$	: Hilbert space of square integrable functions	2
L_2^w	: weighted L_2 space	124
ℓ_2, $\ell_2(\mathbf{Z})$	: Hilbert space of square summable sequences	2
m_p	: p-th discrete moment of the sequence $\{h_k\}$	55
$\mathcal{M}_p$	: p-th moment of the scaling function	24
$\mathcal{M}_p^t$	: p-th moment of $\varphi^*(x)$	56
$\mathcal{M}_p^*$	: p-th modified moment of $\varphi^*(x)$	56
$\widetilde{\mathcal{M}}_p$	: p-th moment of the dual scaling function	24
$\mathcal{M}_\alpha$	: mirror operator around α	24
$N_m(x)$	: cardinal B-spline of order m	16
N	: number of vanishing dual wavelet moments	24
$\widetilde{N}$	: number of vanishing wavelet moments	24
N_{tot}	: total number of vanishing wavelet moments	85
$\mathcal{N}_p$	: p-th moment of the wavelet	24

$\widetilde{\mathcal{N}}_p$	: p-th moment of the dual wavelet	24
$\mathcal{O}$	: big "O" symbol	13
$\mathcal{P}_j$	: projection operator onto V_j	18
$\mathcal{P}'_j$	: approximated projection operator onto V_j	63
$p_{i,j}$	: coefficient of $\Pi(x)$	53
q	: degree of accuracy of a quadrature formula	47
$\mathcal{Q}_j$	: projection operator onto W_j	18
$q_{i,j}$	: coefficient of $\Pi(x)$ in the Chebyshev basis	56
$Q_r[f(x)]$	: quadrature formula applied to the function $f(x)$	46
$r_\alpha(x)$	: right cutoff function	162
$\widetilde{r_\alpha}(x)$	: dual right cutoff function	170
$\mathcal{R}$	: restriction operator	173
$s_{p,k}$	: k-th Fourier coefficient of $\sigma_p(x)$	84
$s^*_{p,k}$	: k-th Fourier coefficient of $\sigma^*_p(x)$	85
$\mathcal{S}$	: sampling operator	67
span	: linear span of a set	35
supp	: support of a function	3
$t_{p,k}$	: k-th Fourier coefficient of $\tau_p(x)$	84
$T_p(x)$	: Chebyshev polynomial of degree p	55
$t_{p,i}$	: coefficient of Chebyshev polynomial	58
$\mathcal{T}$	: total folding operator	164
$\widetilde{\mathcal{T}}$	: dual total folding operator	171
$\mathcal{U}_j$	: subdivision operator on level j	130
V_j	: multiresolution subspace (level j)	14
$\widetilde{V}_j$	: dual multiresolution subspace (level j)	21
$\mathbf{V}_j$	: operator multiresolution subspace (level j)	147
$\mathcal{V}$	: square root differential operator	145
W_j	: multiresolution detail subspace (level j)	17
$\widetilde{W}_j$	: dual multiresolution detail subspace (level j)	21
$\mathbf{W}_j$	: operator multiresolution detail subspace (level j)	147
w_k	: weight of a quadrature formula	46
$\mathcal{W}(a,b)$	: continuous wavelet transform	12
x_k	: abscis of a quadrature formula	46
$\mathcal{X}$	: extension operator	173
$\mathcal{Z}\, f(x,\omega)$	: Zak transform of the function $f(x)$	5
$\gamma_{j,l}$	: wavelet coefficient (level j, position l)	18
δ_i	: Kronecker symbol	15
$\lambda_{j,l}$	: scaling function coefficient (level j, position l)	26
$\Pi(x)$	: product polynomial	53
$\sigma_p(x)$	: monowavelet, 1-periodization of $x^p\,\varphi(x)$	81
$\sigma^*_p(x)$	: monowavelet in the expansion of $\mathcal{Q}_n$	82

τ	: shift of a quadrature formula	47
$\tau_p(x)$	: monowavelet in the expansion of $\mathcal{E}_n$	82
$\varphi(x)$	: scaling function	14
$\varphi^*(x)$	: scaling function transformed to the interval $[\Leftrightarrow 1, 1]$	56
$\widetilde{\varphi}(x)$	: dual scaling function	21
$\Phi(x)$	: operator scaling function	147
$\Phi^*(x)$	: operator dual scaling function	148
$\psi(x)$	: wavelet	17
$\widetilde{\psi}(x)$	: dual wavelet	21
$\varphi_{j,l}(x)$	: scaling function (level j, position l)	15
$\psi_{j,l}(x)$	: wavelet (level j, position l)	17
$\Phi(x)$	: operator wavelet	145
$\Phi^*(x)$	: operator dual wavelet	148
$\chi_S(x)$	: indicator function on the set S	3

List of abbreviations

DC	: Direct Current (constant term)
DCT	: Discrete Cosine Transform
DFT	: Discrete Fourier Transform
EPF	: Equal Parity Folding
FFT	: Fast Fourier Transform
JPEG	: Joint Photometric Expert Group
MSE	: Mean Square Error
FWT	: Fast Wavelet Transform
ODE	: Ordinary Differential Equation
a.e.	: almost everywhere

Constructie en Toepassingen van Wavelets in Numerieke Analyse

Samenvatting

1 Inleiding

1.1 Vertaling van "wavelet"

Aan het eind van de jaren '80 kwam de onderzoeksgroep rond Jean Morlet, een Frans olie-ingenieur, op het idee om verschuivingen en schaalwijzigingen van één welbepaalde functie te gebruiken voor de analyse van seismische gegevens. Uit de wiskundige analyse volgde dat de integraal van deze functie nul moet zijn en dat deze functie naar nul moet convergeren als het argument naar oneindig gaat. M.a.w. deze functie moet een beetje "schommelen" en dan geleidelijk uitsterven; het is een soort "lokaal golfje". Zij doopten functies met deze eigenschappen in het Frans "ondelettes". Logisch, "onde" betekent golf en "-ette" is een suffix dat verkleining aangeeft. Denk bijvoorbeeld maar aan "maisonette", "casquette" en "cassette". Nochtans is het in het Frans niet de regel om via een suffix verkleinwoorden te vormen. "Ondelette" was dan ook een neologisme in het Frans.

Een vertaling in het Engels was snel gevonden: "wavelet". In het Engels wordt het suffix "-let" bij gelegenheid gebruikt om verkleining aan te geven, denk maar aan "starlet", "circlet", "piglet" en "booklet". Ook hier geldt dit niet in het algemeen. In tegenstelling tot het Frans was "wavelet" geen neologisme in het Engels. Het bestond reeds in de betekenis van "little wave". Daarenboven werd het reeds gebruikt in de theoretische fysica, maar in een totaal andere betekenis dan de "wavelets" van dit doctoraat. Dit leidde aanvankelijk tot een Babylonische spraakverwarring.

In de handleiding van de K.U.Leuven voor het schrijven van een doctoraat staat: "Engelstalige termen zullen in het Nederlands vertaald worden". We moeten dus een vertaling voor "wavelet" vinden. Uiteraard willen we consistent blijven met andere werken. Er zijn echter nog geen nederlandstalige artikels of boeken over "wavelets" verschenen. Er bestaan wel enkele andere bronnen.

Op de cursus "Wavelets" (sic) die in 1991 door het Centrum voor Wiskunde en Informatica in Amsterdam werd georganiseerd, lieten alle sprekers (met inbegrip van de auteur) "wavelet" onvertaald. Tom Koornwinder bracht wel de discussie op gang door een wedstrijd uit te schrijven voor een vertaling. Enkele inzendingen waren: golflein, golvelet, kabbel, piefje, resolutiegolfje, rimpel, schaalgolfje, wervel, zoempje. Hijzelf stelde "golflet" voor [118]. Een winnaar werd echter nooit bekendgemaakt.

In het doctoraat van Pierre Verlinden (K.U.Leuven 1993, [186]), wordt "golfje" gebruikt. Men kan echter argumenteren dat "golfje" niet zo geschikt is gezien het "-je" suffix in het Nederlands de regel is voor het vormen van een verkleinwoord. Als we "golfje" als vertaling nemen, hoe kunnen we dan nog onderscheid maken tussen "golfje" in de betekenis van "wavelet" en "golfje" als verkleinwoord van "golf"?

In de cursus die Prof. Bultheel aan de K.U.Leuven doceert, in het college van Prof. Koornwinder aan de Vrije Universiteit van Amsterdam en in de ingenieurs-

thesis van Geert Uytterhoeven (K.U.Leuven 1994, [184]), wordt "wavelet" niet vertaald.

Uit navraag blijkt dat in andere talen (Italiaans, Zweeds, Spaans, Portugees, Grieks, Pools, Chinees, enz.) "wavelet" meestal niet vertaald wordt. Prof. Wei-Chang Shann van de National Central University in Taiwan stelde "ling po" als een mogelijke vertaling in het Chinees voor, maar de kans dat dit algemeen aanvaard wordt in China is klein.

Het voorschrift uit de handleiding voor doctoraten leidt blijkbaar tot een waar probleem. Uit de voorbeelden van de CWI-cursus blijkt dat de Nederlandse taal zeker rijk genoeg is om een geschikte vertaling te vinden. Maar wie moet zoiets beslissen? Spijtig genoeg beschikken we niet over een equivalent van de "Académie Française". Wij kunnen niet eens een consensus (konsensus?) bereiken over een consistente (konsistente?) spelling. Wie geeft de auteur van een doctoraat in deze kwestie het recht om de knoop door te hakken? Het lijkt een dilemma.

Na lang wikken en wegen heb ik besloten om "wavelet" in deze samenvatting niet te vertalen, voornamelijk om verwarring te vermijden. De zoektocht naar een taalkundig correcte, algemeen aanvaarde vertaling is hiermee natuurlijk niet ten einde.

Om de discussie af te ronden, citeren we E. Berode uit zijn stukje "Ongewenste vreemdelingen" verschenen in De Standaard van 7 april 1994:

"Dit is de moraal van het verhaal. Eén: het is chauvenistische nonsens om alle Engelse woorden als ongewenste vreemdelingen in de ban te doen. Buitenlandse zuurstof is goed voor het Nederlandse bloed: een computer is een computer en daarmee basta. Twee: het is kosmopolitische grootheidswaanzin om Engelse woorden kritiekloos en ongenuanceerd te gebruiken. ... Gebruik toch je moers taal, denk ik dan. ... Drie: het is asociale egotripperij om je lezers om de oren te slaan met moeilijke Engelse termen, als er eenvoudige Nederlandse woorden voorhanden zijn. ... Samengevat: Engelse woorden zijn geen smetten, maar maken deel uit van onze woordenschat. Wie ze evenwel te pas en te onpas gebruikt, lijdt aan de Engelse ziekte. Whisky is lekker, maar ik drink hem met mate en serveer hem niet, als mijn gasten hem niet lusten."

1.2 Numerieke analyse en discrete benaderingen

In de numerieke analyse tracht men problemen uit de klassieke analyse benaderend op te lossen met behulp van computers. Het oorspronkelijk probleem is meestal geformuleerd via functies van een continue veranderlijke. In een eerste stap formuleert men een benaderend probleem via een eindig aantal discrete parameters. Vervolgens lost men dit discrete probleem, eventueel weer benaderend, op met een computer. Typische voorbeelden zijn het berekenen van integralen, het oplossen van integraal- en differentiaalvergelijkingen en taken uit de signaalverwerking.

Uiteraard stijgt de nauwkeurigheid, maar ook het rekenwerk, met het aantal discrete parameters. Daarom zijn benaderingmethodes die met een relatief klein

aantal parameters toch een hoge nauwkeurigheid bereiken, erg belangrijk. Benaderingen met deze eigenschap noemen we efficiënt.

Efficiënte benaderingen hebben belangrijke toepassingen. Indien men gegevens nauwkeurig kan benaderen met een een klein aantal parameters, bespaart men opslagruimte, verhoogt men transmissiesnelheid en kan men wiskundige bewerkingen snel uitvoeren. Dit is van cruciaal belang in reële-tijd toepassingen.

1.3 Wat zijn wavelets?

Een voor de hand liggende discretisatie bekomt men door een functie te evalueren op een regelmatig rooster, hierbij veronderstellend dat de functie continu is. Dit is meestal niet erg efficiënt. Andere klassieke methodes gebruiken veeltermen, splines of trigonometrische functies. Wavelets vormen een alternatief dat een aantal interessante eigenschappen combineert. De voornaamste zijn:

1. (bi)orthogonaliteit,
2. multiresolutie,
3. lokaliteit in tijd en frequentie,
4. snelle transformatie-algoritmes.

Als een gevolg hiervan kan men de coëfficiënten van een waveletbenadering snel berekenen en zijn, voor een grote groep van functies, slechts een klein aantal coëfficiënten nodig om een nauwkeurige benadering te bekomen.

Het centraal idee van de wavelettheorie bestaat erin basisfuncties te genereren met behulp van verschuivingen en schaalwijzigingen van één welbepaalde functie, de moederwavelet $\psi(x)$. Oorspronkelijk werkte men met de genormaliseerde functies

$$\frac{1}{\sqrt{a}}\,\psi\left(\frac{x \Leftrightarrow b}{a}\right),$$

waar de schaal- en verschuivingsparameter a en b willekeurige reële waarden aannemen ($a > 0$). Er bestaan echter verscheidene varianten van waveletbenaderingen, afhankelijk van welke waarden men a en b laat aannemen. Wij zullen bijna altijd werken met een dyadische keuze. We stellen dan een functie f voor als

$$f \;=\; \sum_{j,l} \gamma_{j,l}\,\psi_{j,l}, \tag{0.1}$$

waarbij

$$\psi_{j,l}(x) \;=\; \sqrt{2^j}\,\psi(2^j x \Leftrightarrow l).$$

Er bestaan verschillende voorbeelden van wavelets die een basis voor L_2, de ruimte der kwadratisch integreerbare functies, vormen. Voor iedere functie $f \in \mathrm{L}_2$ bestaan dan unieke coëfficiënten $\gamma_{j,l}$ waarvoor (0.1) geldig is. Voor een groot aantal functies zijn de meeste coëfficiënten verwaarloosbaar klein. Door deze gelijk aan nul te stellen, bekomt men een efficiënte benadering.

1.4 Kort overzicht van het doctoraat

Het onderwerp van dit doctoraat is de toepassing van wavelets in de numerieke analyse. Waar de klassieke wavelets blijken te kort te schieten, construeren we nieuwe families. De tekst is als volgt ingedeeld. Na een inleidend hoofdstuk, worden de beginselen van wavelets en multiresolutieanalyse uitgelegd in Hoofdstuk 2. De overige hoofdstukken bevatten originele bijdragen.

Een eerste probleem dat opduikt bij het gebruik van wavelets in numerieke analyse is het nauwkeurig berekenen van de waveletcoëfficiënten. Theoretisch is een coëfficiënt gedefinieerd als de integraal van de functie vermenigvuldigd met een duale wavelet. In praktijk, zeker wanneer men van f slechts een discrete benadering heeft, kan men deze integralen niet exact berekenen. Met kwadratuurformules is het mogelijk benaderingen van deze integralen te berekenen. In Hoofdstuk 3 worden verscheidene kwadratuurformules, speciaal aangepast voor waveletbenaderingen, opgesteld. In combinatie met de snelle wavelettransformatie kan men dan de waveletcoëfficiënten snel en nauwkeurig berekenen.

Het is uiteraard van groot belang het gedrag van de fout van de waveletbenadering te begrijpen. In Hoofdstuk 4 wordt een asymptotische foutenontwikkeling voor de waveletbenadering afgeleid. Hiermee kan men het gedrag van de waveletbenadering voorspellen, algoritmes voor convergentieversnelling opstellen en de benaderingseigenschappen van verschillende wavelets eenvoudig vergelijken.

De klassieke wavelets zijn aangepast aan het inwendig product van de L_2-ruimte. In vele toepassingen komt men echter meer algemene inwendige producten tegen. In Hoofdstuk 5 worden zogenaamde gewogen wavelets opgesteld. Dit zijn wavelets die aangepast zijn aan een gewogen inwendig product. Hun eigenschappen worden nauwkeurig onderzocht in verschillende stellingen. Hieruit blijkt dat ze alle interessante eigenschappen van de klassieke wavelets overnemen.

Een belangrijke toepassing van gewogen wavelets is het numeriek oplossen van gewone differentiaalvergelijkingen met randvoorwaarden. Dit bestuderen we in Hoofdstuk 6. Het belangrijkste resultaat is de constructie van wavelets die algemene differentiaaloperatoren diagonaliseren. Hiermee kan een algoritme voor het oplossen van gewone differentiaalvergelijkingen, dat sneller is dan de tot nog toe gekende methodes, afgeleid worden.

In vele toepassingen is het interessant om niet rechtstreeks met de wavelets te werken, maar wel met hun Fouriergetransformeerde. Deze functies noemt men lokale trigonometrische functies. In Hoofdstuk 7 stellen we veralgemeningen van lokale trigonometrische functies voor, en bestuderen we hun toepassing in gegevenscompressie. We tonen aan dat met deze veralgemening de ongewenste blokpatronen, een typisch probleem in beeldcompressie, bijna volledig verdwijnen.

Tenslotte bevat Hoofdstuk 8 een besluit en een aantal ideeën voor toekomstig onderzoek. We stellen twee nieuwe concepten voor: tijd-frequentie-bases en tweede-generatie-wavelets.

Deze samenvatting volgt de indeling in hoofdstukken van het doctoraat.

2 Wavelets en multiresolutieanalyse

2.1 Inleiding

Beschouw een deelruimte $V_0 \subset \mathrm{L}_2$ en veronderstel dat deze ruimte de volgende eigenschappen heeft:

1. $f(x) \in V_0 \Leftrightarrow f(x \Leftrightarrow 1) \in V_0$,
2. $f(x) \in V_0 \Rightarrow f(x/2) \in V_0$.

"Verschuiven" en "uitrekken" zijn inwendige bewerkingen. Veronderstel ook dat er een schaalfunctie φ met compacte drager bestaat zodat de verzameling $\{\varphi(x \Leftrightarrow l) \mid l \in \mathbf{Z}\}$ een basis voor V_0 vormt. Beschouw de reeks van deelruimtes V_j $(j \in \mathbf{Z})$ gedefinieerd d.m.v. de eigenschap dat

$$f(x) \in V_j \Leftrightarrow f(2^j x) \in V_0. \tag{0.2}$$

Hieruit volgt automatisch dat deze ruimtes genesteld zijn, $V_j \subset V_{j+1}$. De verzameling $\{\varphi_{j,l} \mid l \in \mathbf{Z}\}$, met

$$\varphi_{j,l} \;=\; \sqrt{2^j}\varphi(2^j \Leftrightarrow x),$$

is een basis voor V_j. De resolutie van de ruimte V_j, d.i. het aantal basisfuncties per lengte-eenheid, is gelijk aan 2^j. Deze ruimtes vormen een multiresolutieanalyse indien

1. $\bigcup_j V_j$ dicht is in L_2,
2. $\bigcap_j V_j = \{0\}$.

Beschouw deze ruimtes als benaderingsruimtes voor een algemene functie $f \in \mathrm{L}_2$. Men kan begrijpen dat naarmate j toeneemt de nauwkeurigheid van de benadering stijgt. In de limiet voor $j \to \infty$ is het mogelijk de originele functie f terug te bekomen.

Neem twee opeenvolgende benaderingen van f, bijvoorbeeld deze in V_0 en de meer nauwkeurigere in V_1. Het verschil tussen deze twee, m.a.w. het "detail" nodig om van resolutie 0 naar resolutie 1 te gaan, ligt in een ruimte die een complement vormt van V_0 in V_1. Deze ruimte noteren we als W_0. Dus $V_0 \oplus W_0 = V_1$. De ruimte W_j wordt analoog gedefinieerd als V_j in (0.2),

$$f(x) \in W_j \Leftrightarrow f(2^j x) \in W_0, \tag{0.3}$$

zodat

$$V_j \;=\; V_{j-1} \oplus W_{j-1}.$$

Hieruit volgt dat

$$\mathrm{L}_2 \;=\; \bigoplus_j W_j. \tag{0.4}$$

Deze splitsing van L_2 definieert automatisch projectieoperatoren $\mathcal{P}_j$ en $\mathcal{Q}_j$ van L_2 naar respectievelijk V_j en W_j. De projectieoperator $\mathcal{Q}_j$ is gelijk aan $\mathcal{P}_j \Leftrightarrow \mathcal{P}_{j-1}$. Voor een functie $f \in \mathrm{L}_2$ geldt dat

$$\lim_{j\to\infty} \mathcal{P}_j\, f \;=\; f,$$

en

$$f \;=\; \sum_j \mathcal{Q}_j f.$$

Een wavelet is een functie ψ, zodanig dat de verzameling $\{\psi(x \Leftrightarrow l) \mid l \in \mathbf{Z}\}$ een basis voor W_0 vormt. Uit (0.3) en (0.4) volgt dat de verzameling $\{\psi_{j,l} \mid j, l \in \mathbf{Z}\}$ een basis vormt voor L_2, zodanig dat (0.1) geldig is.

2.2 Biorthogonaliteit

In een multiresolutieanalyse wordt iedere waveletcoëfficiënt $\gamma_{j,l}$ in de voorstelling (0.1) gegeven als het inwendig product van f met een *duale* wavelet $\widetilde{\psi}_{j,l}$,

$$\gamma_{j,l} \;=\; \langle\, f, \widetilde{\psi}_{j,l} \,\rangle\,.$$

Ook de duale wavelets zijn de verschuivingen en schaalwijzigingen van één functie. De duale wavelet $\widetilde{\psi}_{j,l}$ is als het ware het "zusje" van de wavelet $\psi_{j,l}$. De wavelets en duale wavelets zijn biorthogonaal,

$$\langle\, \psi_{j,l}, \widetilde{\psi}_{j',l'} \,\rangle \;=\; \delta_{j-j'}\, \delta_{l-l'} \qquad \text{voor} \qquad j, j', l, l' \in \mathbf{Z}.$$

Er bestaan ook duale schaalfuncties $\widetilde{\varphi}_{j,l}$. Deze zijn biorthogonaal met de schaalfuncties, maar uiteraard enkel binnen één niveau,

$$\langle\, \varphi_{j,l}, \widetilde{\varphi}_{j,l'} \,\rangle \;=\; \delta_{l-l'} \qquad \text{voor} \qquad l, l' \in \mathbf{Z}.$$

De projectieoperatoren kunnen nu geschreven worden als

$$\mathcal{P}_j\, f \;=\; \sum_l \langle\, f, \widetilde{\varphi}_{j,l} \,\rangle\, \varphi_{j,l}, \tag{0.5}$$

en

$$\mathcal{Q}_j\, f \;=\; \sum_l \langle\, f, \widetilde{\psi}_{j,l} \,\rangle\, \psi_{j,l}. \tag{0.6}$$

Wanneer de duale functies en de basisfuncties samenvallen, is de basis orthonormaal.

Een belangrijke eigenschap van een multiresolutieanalyse, is het aantal nulmomenten (N) van de duale wavelet,

$$\int_{-\infty}^{+\infty} x^p \, \tilde{\psi}(x) \, dx \;=\; 0 \quad \text{voor} \quad 0 \leqslant p < N.$$

Het feit dat de duale wavelet N nulmomenten heeft, is equivalent met het feit dat iedere veelterm met graad kleiner dan N kan geschreven worden als een lineaire combinatie van de schaalfuncties $\varphi_{j,l}$ met $l \in \mathbf{Z}$. Er is ook een direct verband met de convergentiesnelheid van de waveletbenadering in de zin dat

$$\| f \Leftrightarrow \mathcal{P}_j f \| \;=\; \mathcal{O}(h^N) \quad \text{met} \quad h = 2^{-j}.$$

2.3 De snelle wavelettransformatie

Veronderstel dat men de coëfficiënten $\lambda_{n,l}$ van een functie $f \in V_n$ kent,

$$f \;=\; \sum_l \lambda_{n,l} \, \varphi_{n,l}.$$

Vermits $V_n = V_{n-1} \oplus W_{n-1}$, kan deze functie ook voorgesteld worden m.b.v. de schaalfunctiebasis van V_{n-1} en de waveletbasis van W_{n-1},

$$f \;=\; \sum_l \lambda_{n-1,l} \, \varphi_{n-1,l} + \sum_l \gamma_{n-1,l} \, \psi_{n-1,l}.$$

Het verband tussen de coëfficiënten $\lambda_{n,l}$ enerzijds en $\lambda_{n-1,l}$ en $\gamma_{n-1,l}$ anderzijds wordt gegeven door eenvoudige lineaire relaties. Deze kunnen gezien worden als convoluties met vaste, eindige filters.

Door het recursief toepassen van deze relaties kan men een algoritme opstellen voor de transformatie tussen de schaalfunctiecoëfficiënten $\lambda_{n,l}$ en de waveletcoëfficiënten $\gamma_{j,l}$ met $j < n$. Dit algoritme heeft lineaire complexiteit en wordt het snelle wavelettransformatie-algoritme genoemd.

2.4 Toepassingen

Aangezien waveletbenaderingen efficiënt zijn, is één van de belangrijke toepassingen gegevenscompressie. Een typisch voorbeeld is beeldcompressie. Een 8-bit grijswaardenbeeld met 512×512 pixels neemt een kwart Mbyte schijfruimte in beslag. Bij digitale video gebruikt men typisch 25 beelden per seconde. De benodigde opslagruimte kan dus snel voor practische problemen zorgen. Met behulp van wavelets is het mogelijk stilstaande beelden met een factor 20 tot 30 te comprimeren zonder essentieel verlies van kwaliteit. In het geval van video zijn nog hogere compressiefactoren mogelijk.

Een andere toepassing is het numeriek oplossen van integraalvergelijkingen. Een nadeel van de klassieke methodes is dat deze meestal leiden tot het inverteren

van een grote, volle matrix. Vermits de meeste integraaloperatoren zachtverlopende kernen hebben met enkel singulariteiten op de diagonaal, is het mogelijk deze met wavelets efficiënt te discretiseren. De matrix is dan bij benadering ijl en kan dus snel geïnverteerd worden.

Andere toepassingen van wavelets werden reeds gevonden in de statistiek, quantumfysica, medische beeldverwerking, spraakherkenning, ruisverwijdering, stromingsmechanica, optica, etc.

3 Kwadratuurformules

3.1 Basisidee

Beschouw de projectie van een functie f in de ruimte V_j (0.5). Zoals we reeds aangaven wordt de coëfficiënt van een basisfunctie in de voorstelling van f gegeven door het inwendig product van f met de bijhorende duale functie. Het inwendig product van twee functies is gedefinieerd als de integraal van hun product over de reële as. In praktijk kan deze integraal bijna nooit analytisch berekend worden. Meestal is er immers geen expliciete uitdrukking voor de schaalfunctie en kent men van f enkel een benadering. We gebruiken daarom numerieke methodes.

Een kwadratuurformule benadert een integraal als een gewogen som van functie-evaluaties:

$$\int_{-\infty}^{+\infty} \widetilde{\varphi}(x)\, f(x)\, dx \;\approx\; \sum_{k=1}^{r} w_k\, f(x_k).$$

De gewichten w_k en abscissen x_k worden bepaald door te eisen dat de kwadratuurformule het exacte resultaat geeft voor veeltermen tot een bepaalde graad. Deze graad noemt men de nauwkeurigheidsgraad (q). In Hoofdstuk 3 worden verscheidene algoritmes voor het opstellen van kwadratuurformules voor waveletcoëfficiënten voorgesteld en vergeleken.

3.2 Constructie

We beschouwen enkel het niveau $j = 0$, vermits de andere niveaus eenvoudig gevonden kunnen worden met een schaalwijziging, zie (0.2). We hebben dus formules nodig voor het berekenen van de coëfficiënten

$$\lambda_{0,l} \;=\; \int_{-\infty}^{+\infty} \widetilde{\varphi}(x \Leftrightarrow l)\, f(x)\, dx.$$

Een eerste belangrijke observatie is dat de abscissen equidistant dienen gekozen te worden om de volgende redenen:

1. In toepassingen is de functie f vaak enkel gekend via evaluaties op een regelmatig rooster.

2. Als de abscissen equidistant zijn, kunnen kwadratuurformules voor naburige coëfficiënten abscissen delen en zijn er dus minder functie-evaluaties nodig.

De enige vrijheidsgraad die nog overblijft in de keuze van de abscissen is een verschuiving τ. We kiezen dus de abscissen als $x_k = k \Leftrightarrow \tau$ met k geheel. Vermits we nu $r+1$ onbekenden hebben (r abscissen en de verschuiving), kunnen we hopen op een nauwkeurigheidsgraad $q = r$. Dit is echter niet gegarandeerd vermits het verband niet lineair is.

We construeren een algoritme voor het berekenen van de verschuiving en de gewichten. De verschuiving τ kan gevonden worden als het nulpunt van een veelterm , (τ). Het criterium voor het bestaan van een kwadratuurformule met nauwkeurigheidsgraad r is dan dat , (τ) een nulpunt moet hebben zodanig dat de abscissen tot de drager van $\tilde{\varphi}$ behoren. We tonen aan dat de meeste schaalfuncties hieraan voldoen.

Deze constructie is gebaseerd op ééntermen. De conditie blijkt echter zeer slecht te zijn. Daarom stellen we een gewijzigde constructie voor die steunt op Chebyshev veeltermen en tonen we aan dat deze wel goed geconditioneerd is.

We bestuderen ook de eigenschappen van de fout. Indien $\lambda'_{n,l}$ de benadering is berekend met de kwadratuurformule, geldt dat

$$|\lambda_{n,l} \Leftrightarrow \lambda'_{n,l}| \;=\; \mathcal{O}(h^{q+1}) \quad \text{met} \quad h = 2^{-n}.$$

Hieruit volgt dat indien de kwadratuurformule de benaderingseigenschappen van de wavelets niet wil verknoeien, q groter dan of gelijk aan $N \Leftrightarrow 1$ moet zijn. Benaderingen $\gamma'_{j,l}$ voor de waveletcoëfficiënten van de grovere niveaus ($j < n$), kunnen nu gevonden worden door de snelle wavelettransformatie toe te passen op de $\lambda'_{n,l}$. Met deze methode neemt de nauwkeurigheid niet af op de grovere niveaus, m.a.w.

$$|\gamma_{j,l} \Leftrightarrow \gamma'_{j,l}| \;=\; \mathcal{O}(h^{q+1}) \quad \text{met} \quad h = 2^{-n} \;\text{ en }\; j < n.$$

3.3 Bijzondere gevallen

We besteden aandacht aan twee bijzondere gevallen.

Eénpuntsformules: Een éénpuntsformule is een kwadratuurformule met $r = 1$. Eénpuntsformules zijn interessant omdat ze triviaal te berekenen zijn. Normaal hebben ze slechts een nauwkeurigheidsgraad $q = 1$. We bewijzen echter dat in het geval van orthogonale schaalfuncties de nauwkeurigheidsgraad gelijk is aan 2. Men kan een éénpuntsformule dus gebruiken voor wavelets met drie of minder nulmomenten. We besteden ook aandacht aan speciale wavelets, zogenaamde coiflets, waarvoor de nauwkeurigheidsgraad van een éénpuntsformule nog hoger is.

Trapeziumregel: De trapeziumregel wordt normaal gezien weinig gebruikt aangezien hij slechts een nauwkeurigheidsgraad $q = 1$ heeft. We tonen echter aan dat voor het berekenen van inwendige producten met schaalfuncties, de trapeziumregel een nauwkeurigheidsgraad $q = N \Leftrightarrow 1$ heeft. Hier is $N \Leftrightarrow 1$ weer de hoogste graad van veeltermen die als een lineaire combinatie van de schaalfunctie en zijn verschuivingen kunnen geschreven worden. Dit is dus een tweede situatie, naast het geval van periodische functies, waar de trapeziumregel een hoge nauwkeurigheidsgraad kan hebben. De trapeziumregel is vooral nuttig in gevallen waar de lengte van de drager van de schaalfunctie vergelijkbaar is met N, bijvoorbeeld voor cardinale B-splines.

3.4 Alternatieven

We bestuderen ook een aantal mogelijke alternatieven. In het geval van interpolatie, tracht men de coëfficiënten $\lambda_{0,l}$ te vinden zodat

$$\sum_l \lambda_{0,l}\, \varphi(x_k \Leftrightarrow l) \;=\; f(x_k),$$

met opnieuw

$$x_k \;=\; k \Leftrightarrow \tau.$$

We tonen aan dat in het algemeen een interpolatieschema niet lokaal is, m.a.w. een coëfficiënt $\lambda_{0,l}$ hangt af van alle functie-evaluaties $f(x_k)$ met $k \in \mathbf{Z}$. We beschouwen echter het bijzonder geval van een interpolerende schaalfunctie, d.i. een schaalfunctie die voldoet aan

$$\varphi(x_k) \;=\; \delta_k,$$

zodanig dat de coëfficiënten gelijk zijn aan de functie-evaluaties. We tonen aan dat sommige Daubechies-schaalfuncties bijna interpolerend zijn indien τ gelijk is aan hun eerste moment. We tonen ook aan hoe het mogelijk is interpolerende schaalfuncties te construeren.

Tenslotte bestuderen we enkele methodes uit de signaalverwerking en hun verband met kwadratuurformules en interpolatieschemas.

4 Foutenontwikkelingen

4.1 Constructie

Voor iedere benaderingsmethode is het belangrijk om het gedrag van de fout te kennen. In Hoofdstuk 4 construeren we een puntsgewijze foutenontwikkeling voor de waveletbenadering van zachtverlopende functies. We bewijzen dat indien $f \in$

$\mathcal{C}^{M+1}$, de foutenontwikkeling eruit ziet als

$$f(x) \Leftrightarrow \mathcal{P}_n f(x) \;=\; \sum_{p=N}^{M} \frac{h^p\, f^{(p)}(x)}{p!}\, \tau_{p-N}(2^n x) + \mathcal{O}(h^{M+1}) \quad \text{met} \quad h = 2^{-n}. \quad (0.7)$$

Zoals we reeds zagen is N het aantal nulmomenten van de duale wavelet. De functies $\tau_p(x)$ zijn periodische functies met periode 1. Ze zijn onafhankelijk van n en we noemen ze monowavelets. Ze kunnen eenvoudig geconstrueerd worden uitgaand van de wavelet. Aan de hand van de Fourierreeks bestuderen we hun eigenschappen in detail. In het geval van splinewavelets tonen we een verband aan tussen de monowavelets en Euler en Bernoulli veeltermen. We bespreken nu de verschillende toepassingen van de foutenontwikkeling.

4.2 Interpolatie

Indien de fout in een punt x gelijk is aan nul, interpoleert de benadering de oorspronkelijke functie: $f(x) = \mathcal{P}_n(x)$. We bewijzen dat de monowavelet $\tau_0(x)$ tenminste twee nulpunten heeft in het interval $[0, 1)$. Bijgevolg heeft de eerste term van de foutenontwikkeling tenminste 2^{n+1} nulpunten per lengte-eenheid. Hieruit volgt dat, voor voldoend kleine h, de waveletbenadering op niveau n de originele functie interpoleert in tenminste 2^{n+1} punten per lengte-eenheid. Dit aantal is het dubbel van de resolutie. We illustreren dit met verscheidene voorbeelden.

4.3 Extrapolatie

Indien men over informatie over de fout beschikt, kan men altijd proberen deze te gebruiken om de nauwkeurigheid van de benadering te verbeteren. Een foutenontwikkeling van de vorm (0.7) kan gebruikt worden in zogenaamde extrapolatiealgoritmes. Hierin probeert men aan de hand van een reeks opeenvolgende benaderingen verschillende componenten van de fout te schatten en vervolgens te elimineren. Op die manier bekomt men convergentieversnelling. Een multiresolutieanalyse levert automatisch een reeks opeenvolgende benaderingen. Wij stellen een convergentieversnellingsalgoritme voor dat gebruikt maakt van Richardson extrapolatie. We geven een voorbeeld met 9 niveaus, waar met extrapolatie de nauwkeurigheid van 5 naar 13 cijfers stijgt.

4.4 Vergelijking van wavelets

De foutenontwikkeling (0.7) kan ook gebruikt worden om verschillende waveletbenaderingen met elkaar te vergelijken.

Een eerste mogelijkheid bestaat erin multiresolutieanalyses met dezelfde waarde voor N te vergelijken. We weten dan dat de convergentiesnelheid altijd $\mathcal{O}(h^N)$ is. Nochtans kan de constante voor deze factor sterk verschillen. Met behulp van

de foutenontwikkeling kunnen we deze constante eenvoudig berekenen. Het blijkt dat de splinewavelets altijd de kleinste constante hebben. Daarenboven gedraagt de verhouding tussen de constante voor de Daubechies-wavelets en de splinewavelets zich als 2^N. Om een bepaalde nauwkeurigheid te halen hebben Daubechies-wavelets dus over het algemeen één extra niveau (en dus dubbel zoveel werk) nodig in vergelijking met splinewavelets. We illustreren dit met verschillende voorbeelden.

Een tweede vergelijking betreft multiresolutieanalyses met dezelfde V_j ruimtes, maar verschillende W_j ruimtes. Inderdaad, eens de V_j ruimtes vastliggen, zijn er nog verschillende mogelijkheden voor de complementaire W_j ruimtes. Eén mogelijkheid bestaat erin de W_j ruimtes als orthogonale complementen te kiezen. In dit geval zijn de projectieoperatoren $\mathcal{P}_j$ en $\mathcal{Q}_j$ orthogonaal en geven ze optimale benaderingen in de L_2 norm.

We bewijzen echter dat de eerste $\widetilde{N}$ termen van de foutenontwikkeling enkel afhangen van de V_j ruimtes en dus onafhankelijk zijn van de manier waarop de W_j ruimtes worden gekozen ($\widetilde{N}$ is het aantal nulmomenten van de wavelet). Voor het bewijs is een nauwkeurige analyse van alle afhankelijkheden in een multiresolutieanalyse nodig. Tenslotte tonen we aan hoe door een verschuiving de nauwkeurigheid van de eerste term sterk verhoogd kan worden.

5 De constructie van gewogen wavelets

De wavelets die we tot nu toe beschouwden, zijn aangepast aan het klassiek inwendig product van L_2. De constructie van deze wavelets steunt volledig op de Fouriertransformatie. Aangezien verschuiving en schaalwijziging in het frequentiedomein algebraïsche operaties worden, noemen we deze wavelets algebraïsche wavelets.

In Hoofdstuk 5 construeren we zogenaamde gewogen wavelets. Deze zijn biorthogonaal met betrekking tot een gewogen inwendig product van de vorm:

$$\langle f, g \rangle_w = \int_{-\infty}^{+\infty} w(x)\, f(x)\, \overline{g(x)}\, dx.$$

De gewichtsfunctie w is integreerbaar en

$$0 < w(x) < \infty.$$

De veralgemening van algebraïsche naar gewogen wavelets kan gezien worden als een equivalent voor de veralgemening van Legendre veeltermen naar bijvoorbeeld Chebyshev of Jacobi veeltermen.

Een essentieel verschil met de algebraïsche wavelets is dat de gewogen wavelets niet langer de verschuivingen of schaalwijzigingen van één functie kunnen zijn. Bijgevolg kan de Fouriertransformatie niet meer gebruikt worden in de constructie.

5.1 Constructie

Wij gaan uit van het zogenaamde Donoho-gemiddeld-interpolatieschema. Dit is een alternatieve constructie van algebraïsche schaalfuncties en wavelets die niet op de Fouriertransformatie steunt. We tonen aan hoe het Donoho-schema kan veralgemeend worden voor de constructie van gewogen wavelets.

Het Donoho-schema is essentieel een subdivisieschema. We kiezen eerst de duale schaalfuncties gelijk aan de indicatorfuncties op de dyadische intervallen: $\widetilde{\varphi}_{j,l} = \chi_{[2^{-j}l,2^{-j}(l+1))}$. Veronderstel nu dat we de functie $\varphi_{i,k}$ willen construeren. Deze moet biorthogonaal zijn met de duale schaalfuncties met betrekking tot het gewogen inwendig product. Vertrek van een rij $\{\lambda_{i,l}\}_l$, met $\lambda_{i,l} = \delta_{l-k}$. Eén stap van het subdivisieschema vervangt nu de rij $\{\lambda_{j,l}\}_l$ door $\{\lambda_{j+1,l}\}_l$ als volgt:

1. Construeer een veelterm P met graad $N = 2D + 1$ zodat

$$\langle P, \widetilde{\varphi}_{j,m+l} \rangle_w = \lambda_{j,m+l} \quad \text{voor} \quad \Leftrightarrow D \leqslant l \leqslant D.$$

2. Bereken twee coëfficiënten op het volgend niveau als

$$\lambda_{j+1,2m} = \langle P, \widetilde{\varphi}_{j+1,2m} \rangle_w \quad \text{en} \quad \lambda_{j+1,2m+1} = \langle P, \widetilde{\varphi}_{j+1,2m+1} \rangle_w .$$

We tonen aan dat men met deze $\lambda_{j,l}$ een rij van functies kan construeren die gelijkmatig naar $\varphi_{i,k}$ convergeert als j naar oneindig gaat. Een belangrijk resultaat is dat, zelfs indien de gewichtsfunctie discontinuïteiten vertoont, de schaalfuncties zachtverlopend zijn. De gewogen wavelets kunnen nu geconstrueerd worden als eindige lineaire combinaties van de schaalfuncties.

5.2 Eigenschappen

De belangrijkste eigenschappen van de gewogen wavelets en schaalfuncties zijn:

1. Zij hebben een compacte drager.

2. Zij zijn biorthogonaal,

$$\langle \varphi_{j,l}, \widetilde{\varphi}_{j,l'} \rangle_w = \delta_{l-l'} \quad \text{voor} \quad l, l' \in \mathbf{Z},$$

en

$$\langle \psi_{j,l}, \widetilde{\psi}_{j',l'} \rangle_w = \delta_{j-j'}\,\delta_{l-l'} \quad \text{voor} \quad j, j', l, l' \in \mathbf{Z}.$$

3. De wavelets hebben één gewogen nulmoment en de duale wavelets N gewogen nulmomenten. Veeltermen met graad kleiner dan N kunnen geschreven worden als lineaire combinaties van de schaalfuncties $\varphi_{j,l}$ met $l \in \mathbf{Z}$. De gewogen waveletbenadering van zachtverlopende functies convergeert als h^N.

Uit deze eigenschappen volgt een snel (lineair) wavelettransformatie-algoritme voor de gewogen waveletcoëfficiënten. Het belangrijkst verschil met het algebraïsch geval is dat de filters nu voor iedere coëfficiënt verschillen.

6 Wavelets en differentiaalvergelijkingen

6.1 Basisidee

In Hoofdstuk 6 onderzoeken we de toepasbaarheid van wavelets in het numeriek oplossen van gewone differentiaalvergelijkingen met randvoorwaarden. Beschouw de vergelijking

$$\mathcal{L}\,u(x) \;=\; f(x) \quad \text{met} \quad x \in (0,1)$$

en met Dirichlet of Neumann randvoorwaarden voor $x = 0, 1$. De methodes die tot nu toe gebruikt worden, kunnen opgesplitst worden in twee grote groepen:

1. Indien $\mathcal{L}$ constante coëfficiënten heeft, worden meestal spectrale methodes gebruikt. De reden is dat de trigonometrische functies eigenfuncties zijn en dus de operator diagonaliseren. Met behulp van de snelle Fouriertransformatie kan men een algoritme afleiden met complexiteit $M \log M$, waarbij M het aantal discrete parameters is.

2. Indien de operator veranderlijke coëfficiënten heeft, gebruikt men eindige elementen of eindige differenties om het probleem te discretiseren. Dit leidt tot een lineair stelsel met een ijle matrix dat meestal iteratief opgelost wordt.

We veronderstellen dat $\mathcal{L}$ een lineaire, zelftoegevoegde differentiaaloperator is die we splitsen als

$$\mathcal{L} \;=\; \mathcal{V}^*\,\mathcal{V}.$$

Beschouw het operator-inwendig-product gegeven door

$$\langle\langle\, f, g \,\rangle\rangle \;=\; \langle\, \mathcal{L} f, g \,\rangle\,.$$

In de Galerkin-methode neemt men twee eindigdimensionale ruimtes S and S^*, en bekomt men een benaderende oplossing $\bar{u} \in S$ door te eisen dat

$$\forall v \in S^* : \quad \langle\langle\, \bar{u}, v \,\rangle\rangle \;=\; \langle\, f, v \,\rangle\,.$$

Dit leidt tot een lineair stelsel van vergelijkingen. De matrix van dit stelsel noemt men de stijfheidsmatrix. Haar elementen zijn de operator-inwendige-producten van de basisfuncties van S en S^*.

Veronderstel nu dat de functies $\{\Psi_{j,k}\}$ en $\{\Psi^*_{j,k}\}$, voor een bepaald bereik van indices, een basis vormen voor respectievelijk S and S^*. De elementen van de stijfheidsmatrix worden gegeven door

$$\langle\langle\, \Psi_{j,k}, \Psi^*_{j',k'} \,\rangle\rangle \;=\; \langle\, \mathcal{L}\,\Psi_{j,k}, \Psi^*_{j',k'} \,\rangle \;=\; \langle\, \mathcal{V}\,\Psi_{j,k}, \mathcal{V}\,\Psi^*_{j',k'} \,\rangle\,.$$

We nemen nu

$$\Psi_{j,k} \;=\; \mathcal{V}^{-1}\,\psi_{j,k} \quad \text{en} \quad \Psi^*_{j,k} \;=\; \mathcal{V}^{-1}\,\widetilde{\psi}_{j,k}\;,$$

waarbij $\psi_{j,k}$ en $\widetilde{\psi}_{j,k}$ biorthogonale wavelets zijn. Bijgevolg is de stijfheidsmatrix diagonaal en kan zij triviaal geïnverteerd worden. We noemen $\Psi_{j,k}$ en $\Psi^*_{j,k}$ operatorwavelets en $\psi_{j,k}$ en $\widetilde{\psi}_{j,k}$ originele wavelets. De operatorwavelets diagonaliseren de operator, hoewel ze geen eigenfuncties zijn.

6.2 Algoritme

De vraag is nu of de operatorwavelets een multiresolutieanalyse genereren en of er een snel transformatie-algoritme bestaat. Dit is niet meteen duidelijk.

In het geval van harmonische of polyharmonische operatoren vindt men de operatorwavelets als de onbepaalde integraal van de originele wavelets. De originele wavelets en de operatorwavelets zijn dan algebraïsche wavelets. De operatorschaalfunctie kan geconstrueerd worden als de convolutie van de oorspronkelijke schaalfunctie met een indicatorfunctie. Uit deze constructie volgt automatisch een snel transformatie-algoritme. Het oplossingsschema voor de differentiaalvergelijking is dan:

1. bereken de wavelettransformatie van het rechterlid,
2. deel iedere coëfficiënt door het overeenkomstig element van de diagonale stijfheidsmatrix,
3. bereken de inverse wavelettransformatie.

Dit is een lineair algoritme. De randvoorwaarden kunnen op een eenvoudige manier in rekening gebracht worden.

Voor andere operatoren zoals de Helmholtz operator en meer algemene operatoren met veranderlijke coëfficiënten kan men niet meer gebruik maken van algebraïsche wavelets. In de plaats daarvan dient men voor de originele wavelets gewogen wavelets te gebruiken. We beschouwen een operator

$$\mathcal{L} \;=\; \Leftrightarrow D\, p(x) D,$$

met p positief. De constructie is gebaseerd op de volgende observatie:

$$\begin{aligned} \langle\langle\, f, g \,\rangle\rangle &= \langle \Leftrightarrow D\, p\, D\, f, g \,\rangle \\ &= \langle\, p\, D\, f, D\, g \,\rangle \\ &= \langle\, p\, D\, f, p\, D\, g \,\rangle_w\,, \end{aligned}$$

waar het gewicht gelijk is aan

$$w \;=\; 1/p.$$

Beschouw biorthogonale, gewogen wavelets $\psi_{j,k}$ en $\widetilde{\psi}_{j,k}$. Definieer de operatorwavelets als

$$\Psi_{j,k} \;=\; D^{-1}\, \psi_{j,k}/p,$$

en analoog voor de duale wavelets. Hieruit volgt opnieuw dat de operatorwavelets de operator diagonaliseren. We tonen aan dat zij een compacte drager hebben en dat er weer een eenvoudig lineair algoritme bestaat. Dit algoritme steunt op de snelle wavelettransformatie voor gewogen wavelets.

6.3 Besluit

We merken op dat voor operatoren met constante coëfficiënten, de waveletmethode sneller is dan de bestaande methodes. De verbetering is echter beperkt ($\mathcal{O}(M)$ in plaats van $\mathcal{O}(M \log M)$). Het belangrijkste resultaat is echter dat voor variable coëfficiënten nu een niet-iteratief lineair algoritme bestaat. Dit was voordien niet mogelijk.

7 Locale trigonometrische functies

Indien men de Fouriertransformatie van wavelets neemt, bekomt men lokale trigonometrische functies. De constructie van orthogonale wavelets door Yves Meyer bestond erin om eerst lokale trigonometrische functies te definieren en vervolgens de wavelets te vinden d.m.v. de inverse Fouriertransformatie. In sommige toepassingen is het interessant om rechtstreeks met de lokale trigonometrische functies te werken.

7.1 Basisidee

De Fouriertransformatie geeft een voorstelling van functies met behulp van bouwblokken die slechts één frequentie bevatten, maar geen lokaliteit hebben in het tijdsdomein. Men kan een basis met betere lokaliteit bekomen door de reële as op te splitsen in intervallen en in ieder interval een Fourierreeks te gebruiken. Op die manier bekomt men een orthogonale basis met tijd en frequentie lokaliteit.

Deze eenvoudige aanpak heeft echter verscheidene nadelen:

1. Fourierreeksen hebben enkel snelle convergentie voor zachtverlopende periodische functies. De beperking van een zachtverlopende functie tot een interval is niet noodzakelijk periodisch. Bijgevolg gaan de coëfficiënten niet snel naar nul.

2. Een benadering met deze functies vertoont discontinuïteiten aan de rand van ieder interval. Dit veroorzaakt ongewenste blokpatronen bij beeldcompressie.

3. Correlatie tussen de verschillende intervallen kan niet uitgebuit worden.

Coifman en Meyer vonden een oplossing voor de eerste twee problemen. Zij definieerden zogenaamde vouwoperatoren die een zachtverlopende functie transformeren in een functie die de even rechterafgeleiden gelijk aan nul heeft in het linkse eindpunt van het interval, en oneven linkerafgeleiden nul heeft in het rechtse eindpunt. Deze functies kunnen efficiënt voorgesteld worden als een combinatie van sinusfuncties met kwart golflengte op ieder interval. Door hierop de inverse vouwoperator toe te passen bekomt men zachtverlopende lokale trigonometrische functies.

In Hoofdstuk 7 introduceren we een nieuwe notatie en beschrijving van hun werk. Dit laat ons toe de constructie te veralgemenen naar het biorthogonaal geval. Hiermee kunnen een aantal nadelen van hun methode vermeden worden.

7.2 Biorthogonale bases

Beschouw een verdeling van de reële as in intervallen met lengte 1. De basis van Coifman en Meyer bestaat dan uit functies van de vorm $b_l\, s_{l,k}$ waarbij

$$s_{l,k} \;=\; \sqrt{2}\sin\frac{2k+1}{2}\pi\,(x \Leftrightarrow l),$$

en b_l een zachtverlopende functie is waarvan de drager tot $[l \Leftrightarrow 1/2, l+3/2]$ behoort. Indien

$$\sum_l b_l^2(x) \;=\; 1,$$

vormen deze functies met $l, k \in \mathbf{Z}$ een orthonormale basis voor L_2. We kunnen dus iedere functie $f \in \mathrm{L}_2$ schrijven als

$$f \;=\; \sum_{l,k} c_{l,k}\, b_l\, s_{l,k}.$$

De coëfficiënten worden gegeven als

$$c_{l,k} \;=\; \langle\, f, b_l\, s_{l,k}\,\rangle \;=\; \langle\, \mathcal{F}f, s_{l,k}\,\rangle\,.$$

Hierin is $\mathcal{F}$ een vouwoperator die op het interval $[l, l+1]$ gelijk is aan

$$\mathcal{F} \;=\; (1 \Leftrightarrow \mathcal{M}_l + \mathcal{M}_{l+1})\, b_l,$$

waar $\mathcal{M}_l$ de reflectieoperator rond l is. De vouwoperator is eenvoudig te implementeren en laat ons toe de coëfficiënten via een snelle sinustransformatie te berekenen.

Het nadeel van deze basis is dat er geen resolutie van de constante meer is. Dit betekent dat de constante enkel kan voorgesteld worden met een oneindig aantal basisfuncties voor ieder interval. Het feit dat een eenvoudige functie zoals een constante niet tot de basis behoort, vormt een serieuze belemmering in allerlei toepassingen.

Dit probleem kan opgelost worden m.b.v. de veralgemening naar een biorthogonale basis. De basisfuncties worden nu gegeven door $b_l\, t_{l,k}$ waarbij

$$t_{2l,k} \;=\; \sqrt{2}\sin(k+1)\pi\,(x \Leftrightarrow 2l) \quad \text{met} \quad k \geqslant 0,$$

$$t_{2l+1,k} \;=\; \sqrt{2}\cos k\pi\,(x \Leftrightarrow 2l \Leftrightarrow 1) \quad \text{met} \quad k \geqslant 1,$$

en

$$t_{2l+1,0} \;=\; 1.$$

Deze basis is stabiel indien er constanten A en B bestaan zodat

$$0 < A < \sum_l b_l^2 < B < \infty.$$

We tonen aan dat het mogelijk is functies b_l te vinden zodanig dat de basis nu wel een resolutie van de constante heeft, of

$$\sum_l b_l \, t_{l,0} \;=\; 1.$$

Ook de duale functies kunnen eenvoudig geconstrueerd worden.

We bespreken ook een variant, die gebaseerd is op een vouwoperator met gelijke pariteit links en rechts. Deze heeft als nadeel dat de functies geen basis voor L_2 meer vormen, maar als voordeel dat men zachtverlopende functies nauwkeuriger kan benaderen. We bewijzen dat met deze basis zelfs een resolutie van lineaire functies mogelijk is.

7.3 Toepassingen

De belangrijkste toepassing die we beschouwen is gegevenscompressie. We vergelijken de verschillende lokale trigometrische bases in allerlei gevallen. Het blijkt dat de basis met gelijke pariteit meestal beter is wanneer men de L_2 norm gebruikt ter vergelijking. In het geval van beeldcompressie blijkt echter dat de biorthogonale basis visueel het beste resultaat geeft. De typische blokpatronen verdwijnen bijna volledig.

8 Besluit en toekomstig onderzoek

In Hoofdstuk 8 formuleren we een besluit, dat aanleiding geeft tot de definitie van een tijd-frequentie-basis, en stellen we enkele ideeën voor toekomstig onderzoek voor, die aanleiding geven tot de definitie van tweede-generatie-wavelets.

8.1 Tijd-frequentie-bases

Beschouw de volgende twee voorstellingen van een algemene functie f:

$$f(t) \;=\; \int_{-\infty}^{+\infty} f(s)\, \delta(t \Leftrightarrow s)\, ds,$$

en

$$f(t) \;=\; \frac{1}{2\pi} \int_{-\infty}^{+\infty} \widehat{f}(\omega)\, e^{i\omega t}\, d\omega.$$

In het eerste geval zijn de bouwblokken extreem lokaal in het tijdsdomein, maar helemaal niet lokaal in het frequentiedomein; in het andere geval is het net andersom. Een tijd-frequentie-basis levert een compromis tussen deze twee extremen.

We beschouwen daarom een voorstelling

$$f = \sum_{\lambda} c_\lambda \, \psi_\lambda, \tag{0.8}$$

waarbij λ een algemene, aftelbare index is. We noemen de verzameling $\{\psi_\lambda\}$ een tijd-frequentie-basis indien:

1. De functies φ_λ een stabiele basis vormen. Hiermee bedoelen we dat voor een functie f behorend tot een algemene functieruimte, unieke coëfficiënten c_λ bestaan zodat de uitdrukking (0.8) convergeert.

2. Er uitdrukkingen bestaan voor de coördinaatfunctionalen.

3. De basisfuncties lokaal zijn in tijd en in frequentie. Ze moeten snel naar nul gaan buiten hun centrum of, indien mogelijk, een compacte drager hebben.

4. Het mogelijk is de bij deze basis horende transformatie eenvoudig op een computer te implementeren.

Typische voorbeelden van tijd-frequentie-bases zijn algebraïsche wavelets en lokale trigonometrische functies. Deze twee noemen we eerste-generatie-wavelets.

8.2 Nadelen van de eerste-generatie-wavelets

1. De eerste-generatie-wavelets vormen een basis voor functies gedefinieerd op de volledige Euclidische ruimte $\mathbf{R}^n$. In vele toepassingen, zoals segmentatie en de oplossing van differentiaalvergelijkingen, heeft men echter wavelets nodig die gedefinieerd zijn op vrij willekeurige deelverzamelingen van de Euclidische ruimte. Vaak heeft men ook wavelets nodig die gedefinieerd zijn op curves en oppervlakken.

2. Eerste-generatie-wavelets vormen een basis voor L_2 met het klassiek inwendig product. Vaak heeft men echter wavelets nodig aangepast aan meer algemene inwendige producten. Voorbeelden zijn gewogen inwendige producten of operator-inwendige-producten.

3. Eerste-generatie-wavelets zijn niet invariant zijn onder eenvoudige meetkundige operaties zoals verschuiving, schaalwijziging en draaiing. Dit is nodig in toepassingen zoals de analyse en compressie van digitale video en in automatische doelherkenning.

8.3 Tweede-generatie-wavelets

We kunnen nu eenvoudigweg tweede-generatie-wavelets definieren als iedere tijd-frequentie-basis die een (gedeeltelijke) oplossing biedt voor één van deze problemen. Voorbeelden zijn de gewogen wavelets van Hoofdstuk 5, de operatorwavelets van Hoofdstuk 6 en wavelets op een interval. Het toekomstig onderzoek omvat de verdere ontwikkeling van tweede-generatie-wavelets.

Chapter 1

Introduction

"The beginner ... should not be discouraged if ... he finds that he does not have the prerequisites for reading the prerequisites."
—Paul Halmos, *borrowed from* [146].

Understanding the material in this thesis requires some elementary knowledge of Fourier analysis, linear algebra, and functional analysis. In this chapter, we quickly review these preliminaries and set more notation. Secondly, we give a brief history of the evolution of wavelet theory. Finally, we outline the thesis and situate its contribution.

1.1 Preliminaries

1.1.1 General

We use the standard notation $\mathbf{N}$, $\mathbf{Z}$, $\mathbf{R}$, and $\mathbf{C}$ for the sets of naturals, integers, reals, and complex numbers, respectively. We typically use the symbols j, k, l, m, and n to denote integers. We often use conditions like $0 < k \leqslant 4$ and symbols like k under a summation sign. In these cases $k \in \mathbf{Z}$ is always understood.

We only consider vector spaces whose scalar field is $\mathbf{C}$. We use the symbol $\|\cdot\|$ for the norm in a metric space. To avoid confusion we sometimes add the symbol of the space as a subscript. A *Banach space* is a complete normed space and a *Hilbert space* is a Banach space whose norm is induced by an inner product. Two subspaces A and B of a space S form a *direct sum decomposition* of that space provided every element of S can be written uniquely as a sum of an element of A and an element of B. We then use the notation $S = A \oplus B$.

1.1.2 Functions and Function Spaces

We mostly work with functions defined on $\mathbf{R}$ that take values in $\mathbf{C}$. On the real line we use only the *Lebesgue measure*, see [154, 155]. A real-valued function is

measurable whenever its domain is measurable and when the sets

$$\{x \mid f(x) < A\}$$

are measurable for all $A \in \mathbf{R}$. A property is said to hold *almost everywhere*, or a.e., if the set of points where it fails has measure zero. A countable set is a typical example of a set with measure zero. As commonly done, we often omit the term a.e. if we feel no confusion is possible. For example, we sometimes use the word bounded when, strictly speaking, we should use essentially bounded.

We denote a function both by simply f and by $f(x)$ when we want to emphasize its dependency on x. Note that in the latter case, $f(x)$ does not stand for the value of the function at the point x. We usually avoid confusion by using phrases like "the *function* $f(x)$".

A measurable function f belongs to the *Lebesgue space* $\mathrm{L}_p(\mathbf{R})$, $1 \leqslant p < \infty$, if

$$\|f\|_p \;=\; \left(\int_{-\infty}^{+\infty} |f(x)|^p \, dx\right)^{1/p} \;<\; \infty,$$

and to $\mathrm{L}_\infty(\mathbf{R})$ if

$$\|f\|_\infty \;=\; \operatorname*{ess\,sup}_{x \in \mathbf{R}} |f(x)| \;<\; \infty.$$

Each Lebesgue space is a Banach space. Their elements are actually equivalence classes of functions that coincide almost everywhere. Therefore, equalities in these spaces should be seen as equalities between the equivalence classes.

The space $\mathrm{L}_2(\mathbf{R})$ is a Hilbert space where the inner product of two functions f and g is defined as

$$\langle\, f, g \,\rangle \;=\; \int_{-\infty}^{+\infty} f(x)\, \overline{g(x)}\, dx.$$

Two functions are *orthogonal* if their inner product is zero. We also use Lebesgue spaces of functions defined on subsets of the real line. In that case we just replace $\mathbf{R}$ by the symbol for the subset. Whenever it is clear that we are working on the real line, we use the notation L_p.

A *sequence* is a function defined on the integers and is denoted by $\{t_k\}$, again without explicitly indicating the range of k. A sequence $\{t_k\}$ with $t_k \in \mathbf{C}$ belongs to the Hilbert spaces of square summable sequences, $\ell_2(\mathbf{Z})$ or ℓ_2 for short, if

$$\sum_k |t_k|^2 \;<\; \infty.$$

A countable subset $\{f_k\}$ of a Hilbert space is a *Riesz basis* if every element f of the space can be written uniquely as $f = \sum_k c_k\, f_k$, and if positive constants A and B exist such that

$$A\, \|f\|^2 \;\leqslant\; \sum_k |c_k|^2 \;\leqslant\; B\, \|f\|^2.$$

A Riesz basis is an *orthogonal basis* if the f_k are mutually orthogonal. In this case $A = B = 1$.

A function f is *Hölder* continuous of order α $(0 < \alpha \leqslant 1)$ at a point x if

$$|f(x) \Leftrightarrow f(x+h)| \;=\; \mathcal{O}(h^\alpha).$$

A function f satisfies a *Lipschitz condition* of order α $(0 < \alpha \leqslant 1)$ in a set S if

$$\sup_{x,y \in S,\, |x-y|<h} |f(x) \Leftrightarrow f(y)| \;=\; \mathcal{O}(h^\alpha).$$

We then say it belongs to the space $\mathrm{Lip}^\alpha(S)$. Higher order Hölder and Lipschitz regularity can be defined in a straightforward way by using higher order differences of f.

The space $\mathcal{C}^k$ is the space of k times continuously differentiable functions. Here we use derivatives in the classic sense. We also define the homogeneous *Sobolev* space,

$$\mathrm{H}_{k,p}(\mathbf{R}) \;=\; \{f \mid D^k f \in \mathrm{L}_p\},$$

where we use derivatives in a distributional sense. Here a seminorm is defined as

$$|f|_{k,p} \;=\; \|f^{(k)}\|_p.$$

The space $\mathcal{D}^k$ is defined as the space of bounded functions that decay faster than an inverse polynomial,

$$\mathcal{D}^k \;=\; \{f \mid \;\; |f(x)| \;\leqslant\; C\,(1+|x|)^{-(k+1+\epsilon)}\,, \text{ for some } \epsilon > 0\}.$$

The *T-periodization* of a function $f \in \mathcal{D}^0$ is a periodic function with period T that is defined as

$$\sum_l f(x + T\,l).$$

The *support* of a function is given by

$$\mathrm{supp}\, f \;=\; \mathrm{clos}\,\{x \mid f(x) \neq 0\}.$$

Here $\mathrm{clos}\, S$ stands for the closure of the set S. The *indicator function* $\chi_S(x)$ of a set S is defined as

$$\chi_S(x) \;=\; \begin{cases} 1 & \text{if} \quad x \in S \\ 0 & \text{if} \quad x \notin S. \end{cases}$$

We say that a function w is an $\mathcal{L}$*-spline* if

$$\mathcal{L}^*\mathcal{L}\,w \;=\; 0 \quad \text{ and } \quad w \in \mathcal{C}^{2m-2},$$

where $\mathcal{L}^*$ is the adjoint of $\mathcal{L}$, a linear differential operator of order m. This definition leads to the classic piecewise polynomial splines of order $2m$ in case $\mathcal{L} = D^m$.

1.1.3 Operators

We only consider operators that are continuous linear maps from a Hilbert space S to itself. The *norm* of an operator T is then defined as

$$\|T\| \;=\; \sup_{\|f\|_S=1} \|T\,f\|_S.$$

The operator T^* is the *adjoint* of an operator T if

$$\langle\, T\,f, g\,\rangle \;=\; \langle\, f, T^*\,g\,\rangle,$$

for all $f,\ g \in S$. The *kernel* of an operator is given by

$$\ker T \;=\; \{x \in S \mid T\,x = 0\},$$

and its *range* by

$$\mathrm{range}\,T \;=\; \{Tx \mid x \in S\}.$$

An operator is *invertible* if its kernel is equal to $\{0\}$ and if a bounded operator G exists so that $GF = FG = \mathbf{1}$. The *condition number* of an operator is defined as $\kappa = \|T\| \cdot \|T^{-1}\|$. An operator is *selfadjoint* in case $T^* = T$ and *unitary* in case $T^* = T^{-1}$.

An operator is a *projection* whenever $T^2 = T$. The range and kernel of a projection operator form a direct sum decomposition of the space. A projection operator is orthogonal when its range is orthogonal to its kernel. A projection operator is orthogonal if and only if it is selfadjoint.

1.1.4 Fourier Analysis

The *Fourier transform* of a function $f \in \mathrm{L}_2$ is defined as

$$\widehat{f}(\omega) \;=\; \int_{-\infty}^{+\infty} f(x)\,e^{-i\omega x}\,dx.$$

The inverse Fourier transform is given by

$$f(x) \;=\; \frac{1}{2\pi}\int_{-\infty}^{+\infty} \widehat{f}(\omega)\,e^{i\omega x}\,d\omega.$$

The Fourier transform is a unitary transform up to a constant factor, or, more precisely,

$$\sqrt{2\pi}\,\|f\|_2 \;=\; \|\widehat{f}\|_2.$$

We use the Poisson summation formula in the following two forms:

$$\sum_l f(x \Leftrightarrow l) \;=\; \sum_k \widehat{f}(2k\pi)\,e^{i2k\pi x},$$

and

$$\sum_l \langle f, g(\cdot \Leftrightarrow l) \rangle \, e^{-i\omega l} \;=\; \sum_k \widehat{f}(\omega + k2\pi)\, \overline{\widehat{g}(\omega + k2\pi)}.$$

The *Fourier series* of a function $f \in \mathrm{L}_2([0,1])$ is given by

$$f(x) \;=\; \sum_k c_k \, e^{i2\pi xk} \quad \text{with} \quad c_k \;=\; \int_0^1 f(x)\, e^{-i2\pi xk}\, dx.$$

The *discrete Fourier transform* of a sequence $\{a_k\}$ is given by

$$a(\omega) \;=\; \sum_k a_k \, e^{-i\omega k}.$$

The *Zak transform* of a function $f(x) \in \mathrm{L}_2(\mathbf{R})$ is defined as [104, 200]

$$(\mathcal{Z}f)(x,\omega) \;=\; \sum_l e^{-i\omega l}\, f(x+l) \quad \text{for} \quad x, \omega \in \mathbf{R},$$

and satisfies

$$(\mathcal{Z}f)(x,\omega) \;=\; \sum_k \widehat{f}(\omega + 2\pi k)\, e^{i(\omega+2\pi k)x}.$$

1.2 A brief history of wavelets

In this section we give an overview of the development of wavelets. We do so by pointing out the important milestones.

The basic idea of the theory is to represent general functions in terms of simpler, fixed building blocks at different scales and positions. This has been found to be a useful approach in several different areas. For example, in signal processing we have techniques such as subband coding, quadrature mirror filters, and pyramid schemes, while in mathematical physics similar ideas are studied as part of the theory of Coherent States. Wavelet theory presents a synthesis of these different approaches.

In abstract mathematics, it has been known for some time that techniques based on Fourier series and Fourier transforms are not quite adequate for many problems. So-called *Littlewood-Paley techniques* are often effective substitutes. These techniques were developed in the 30's to help understand problems such as the summability of Fourier series and the boundary behavior of analytic functions. In the 50's and 60's, they developed into powerful tools for studying solutions of differential and integral equations. People realized that they fit into *Calderón-Zygmund theory*, an area of harmonic analysis.

One of the standard approaches in Calderón-Zygmund theory is to break up a complicated phenomenon into many simple pieces and study each of the pieces separately. In the 70's, atomic decompositions, which are sums of simple functions, were widely used in Hardy space theory [50]. One method used to establish that

a general function f has such a decomposition is to start with the "Calderón formula". This states that a general function f satisfies

$$f(x) \;=\; \int_0^{+\infty} \int_{-\infty}^{+\infty} (\psi_t * f)(y)\, \widetilde{\psi}_t(x \Leftrightarrow y)\, \frac{dy\, dt}{t},$$

where $\psi_t(x) = t^{-1}\psi(x/t)$, likewise for $\widetilde{\psi}_t(x)$, and where ψ and $\widetilde{\psi}$ are appropriate fixed functions. As we shall see in Chapter 2, this representation is an example of a continuous wavelet transform. In the early 80's, Strömberg discovered the first orthogonal wavelet [169]. This was done in the context of trying to further understand Sobolev and Hardy spaces, as well as other spaces used to measure the size and smoothness of functions. In 1976, a discrete version of the Calderón formula (something that we presently would call a discrete wavelet transform) was used by Björn Jawerth for similar purposes in his thesis [105]. Long before this, there were motivating results by Haar [99], Franklin [86], Ciesielski [40], Peetre [146], and others.

Independent from these developments in harmonic analysis, Alex Grossmann, Jean Morlet, and their coworkers studied the wavelet transform in its continuous form [96, 97, 98]. The theory of "frames" provided a suitable general framework for their investigations [67].

In the early to mid 80's, one realized, with some excitement, that the Littlewood-Paley representations had discrete analogs and could give a unified view of many of the results in harmonic analysis. This was done independently by Yves Meyer and his collaborators as well as by Mike Frazier and Björn Jawerth in their work on the φ-transform [87, 88, 89]. One also started to understand that these techniques could be effective substitutes for Fourier series in numerical applications. The emphasis then shifted more towards the representations themselves and the building blocks involved. As a result the name of the theory changed. Yves Meyer and Jean Morlet suggested the word "wavelet" for the building blocks and what earlier had been referred to as Littlewood-Paley theory, now started to be called wavelet theory.

Pierre-Gilles Lemarié and Yves Meyer [126], independent of Strömberg, constructed orthogonal wavelet expansions. With the notion of multiresolution analysis, introduced by Stéphane Mallat and Yves Meyer, a systematic framework for understanding these orthogonal expansions was developed [134, 135, 136]. Soon, Ingrid Daubechies [63] gave a construction of compactly supported wavelets with arbitrarily high, but fixed, regularity. This takes us up to a fairly recent time in the history of wavelet theory. Several other people have made substantial contributions to the field over the past few years. Their work and the appropriate references will be discussed in Chapter 2.

1.3 Outline of the thesis

As one can see from the history, the solution of differential and integral equations was one of the motivations for the development of wavelet theory from the onset. Over the last few years substantial progress was made concerning the numerical treatment of these problems. However, we are far from the ultimate goal, still needed is the development of fast, accurate numerical solvers for a wide range of linear and nonlinear differential and integral equations. This thesis can be seen as one step towards this goal. It is organized into eight chapters whose interrelation is represented in Figure 1.1.

In Chapter 2 we describe the notions from wavelets and multiresolution analysis that are needed to understand the remainder of the thesis. It mostly addresses developments since the definition of multiresolution analysis by Mallat and Meyer (approximately 1985). It discusses several independent contributions such as semi-orthogonal and biorthogonal wavelets in a unified setting.

Chapters 3 and 4 address two typical problems that appear when implementing wavelet approximations on a computer. The first one involves the calculation of the wavelet coefficients. Theoretically, these are defined as inner products with dual functions. In practice, however, these inner products cannot be calculated exactly and one needs to approximate them numerically. In Chapter 3, we present a solution based on quadrature formulae. We compare it with other possibilities such as interpolation and filtering. The second problem concerns the behavior of the error of a wavelet approximation. The main result of Chapter 4 is an asymptotical expansion for the error in case of smooth functions. This leads to the definition of a new family of functions, which we refer to as "monowavelets". The expansion is useful for practical purposes such as convergence acceleration and interpolation. This is illustrated with numerical examples. Moreover, the expansion also serves as a simple tool for comparing the approximation properties of different wavelets.

In the following two chapters, we study how the definition of multiresolution analysis can be generalized. In Chapter 5, we abandon the classic inner product and construct wavelets that are adapted to a weighted inner product. By only assuming a mild condition on the weight, we show that these wavelets can be compactly supported and smooth. Their construction also can easily be implemented on a computer. Until now it was an open problem whether such wavelets exist. They can be seen as a part of a new development, which we call "second generation wavelets". The whole idea of the second generation wavelets is to generalize the classic definition of wavelets without giving up on their powerful properties.

In Chapter 6, the weighted wavelets are used to solve boundary value ordinary differential equations. It is shown that, starting from the weighted wavelets, one can automatically construct a basis that diagonalizes a general differential operator. As a result, the numerical solution becomes trivial. A linear, non-iterative algorithm is presented and numerical results are given.

One can read Chapters 5 and 6 immediately after Chapter 2. However, it is recommended to read Chapters 3 and 4 first.

Chapter 7 can be read immediately after Chapter 2. It concerns the construction of wavelets on the Fourier transform side, or so-called local trigonometric functions. We present two generalizations of the original construction of Coifman and Meyer and study their properties. One of the applications here is reducing artifacts in image compression.

At the end of each chapter, a discussion of possible future research is included and new ideas related to the material in the chapter are discussed. Chapter 8 draws a final conclusion and discusses new directions of research that opened up as a result of the work done in this thesis.

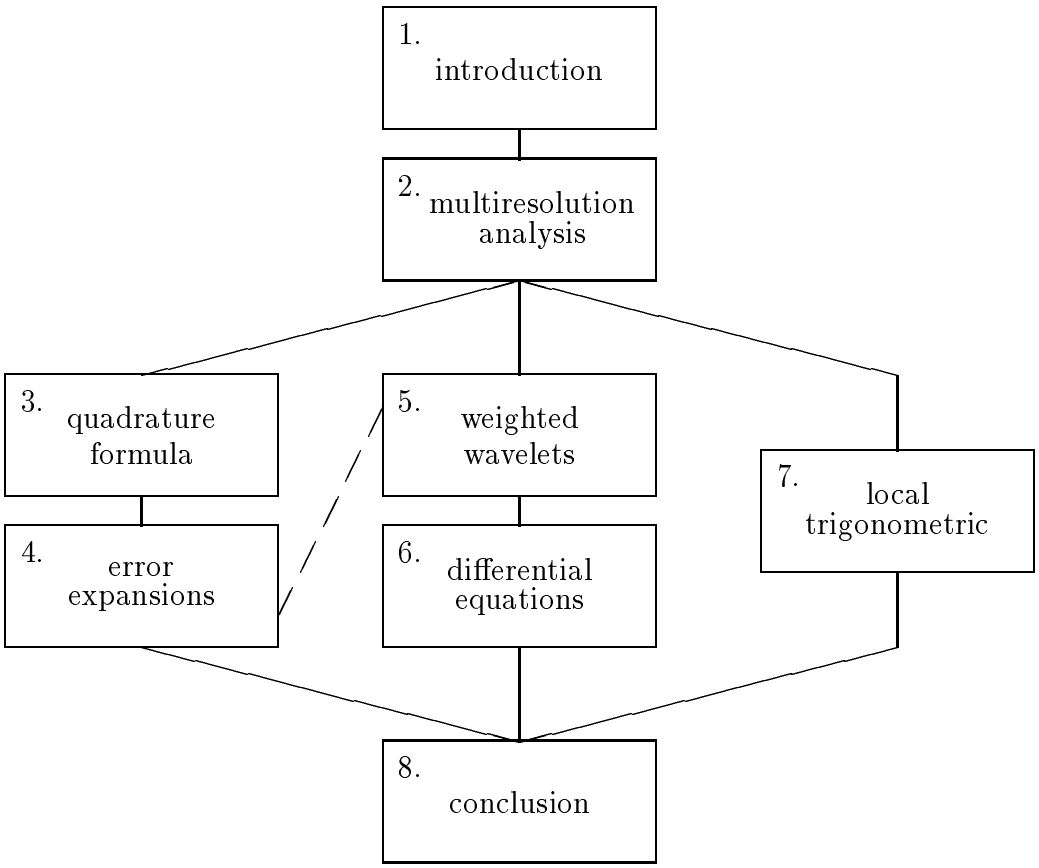

Figure 1.1: Flow chart of the thesis.

"And so I think I have omitted nothing
that is necessary to an understanding of curved lines."
—René Descartes, *La Géométrie (1637).*

Chapter 2

Multiresolution Analysis and Wavelets

> *"There is only one thing you should do. Go into yourself. Find out the reason that commands you to write; see whether it has spread its roots into the very depths of your heart; confess to yourself whether you would have to die if you were forbidden to write. This most of all: ask yourself in the most silent hour of your night:* must *I write? Dig into yourself for a deep answer. And if this answer rings with a strong, simple "*I must,*" then build your life in accordance with this necessity; your whole life, even into its humblest and most indifferent hour, must become a sign and witness to this impulse."*
>
> —Rainer Maria Rilke, *Letters to a young poet (1903–1908).*

2.1 Introduction

In this chapter, we present wavelets and multiresolution analysis. The material represents the core of the wavelet theory that is needed to understand and situate the following chapters. Its main contribution is that it treats several recent developments in a unified setting.

The chapter is organized as follows. We first introduce the "continuous wavelet transform". This discussion is mainly included for comparison with the multiresolution analysis wavelets. Next, we give the definitions of "multiresolution analysis" and "scaling function" (Section 2.3), derive some basic properties and illustrate these with some examples. In this section, we also give the basic definition of "wavelet". Wavelets are then studied in more detail in the next sections. Section 2.4 discusses orthogonal wavelets, while Section 2.5 treats biorthogonal wavelets, a generalization of the orthogonal ones, and semiorthogonal wavelets, a compromise between the previous two. In the following section, we study the connection between wavelets and polynomials, and show how this relates to the ap-

proximation properties of wavelet expansions. In Section 2.7 we show how a "fast wavelet transform" can be derived from the multiresolution analysis properties.

At this point, i.e. after the study of the basic properties of multiresolution analysis, we are ready to single out some desirable properties of wavelets. This is done in Section 2.8. We also give several examples of wavelet families, such as Daubechies' and spline wavelets, and compare their properties. The next three sections focus on more recent developments such as wavelets on an interval, wavelet packets, and multidimensional wavelets. These sections can be read independently. Finally, in the last section (Section 2.12) we consider the basic ideas associated with two important applications: data compression and analysis of linear operators.

2.2 The continuous wavelet transform

A wavelet expansion typically uses translations and dilations of one fixed function, the wavelet $\psi \in \mathrm{L}_2(\mathbf{R})$. In the case of the continuous wavelet transform, the translation and dilation parameters vary continuously. In other words, the transform makes use of the functions

$$\psi_{a;b}(x) \;=\; \frac{1}{\sqrt{|a|}}\,\psi\left(\frac{x \Leftrightarrow b}{a}\right) \qquad \text{with} \qquad a, b \in \mathbf{R},\ a \neq 0.$$

These functions are normalized to have a constant L_2 norm. The *continuous wavelet transform* of a function $f \in \mathrm{L}_2(\mathbf{R})$ is now defined by

$$\mathcal{W}(a, b) \;=\; \langle\, f, \psi_{a;b} \,\rangle\,. \tag{2.1}$$

Using Parseval's identity, we can write this as

$$2\pi\,\mathcal{W}(a, b) \;=\; \langle\, \widehat{f}, \widehat{\psi}_{a;b} \,\rangle \tag{2.2}$$

where

$$\widehat{\psi}_{a;b}(\omega) \;=\; \frac{a}{\sqrt{|a|}}\,e^{-i\omega b}\,\widehat{\psi}(a\omega).$$

We assume now that the wavelet ψ and its Fourier transform $\widehat{\psi}$ are functions with finite centers $\bar{x}$ and $\bar{\omega}$ and finite radii Δ_x and Δ_ω. These quantities are defined as

$$\begin{aligned} \bar{x} \;&=\; \frac{1}{\|\psi\|^2}\int_{-\infty}^{+\infty} x\,|\psi(x)|^2\,dx, \\ \Delta_x^2 \;&=\; \frac{1}{\|\psi\|^2}\int_{-\infty}^{+\infty} (x \Leftrightarrow \bar{x})^2\,|\psi(x)|^2\,dx, \end{aligned}$$

and similarly for $\bar{\omega}$ and Δ_ω. The variable x usually represents either time or space; we choose the first and refer to x as time. From (2.1) and (2.2), we see that the

continuous wavelet transform at (a, b) picks up information about f, mostly from the time interval $[b + a\bar{x} \Leftrightarrow a\Delta_x, b + a\bar{x} + a\Delta_x]$ and from the frequency interval $[(\bar{\omega} \Leftrightarrow \Delta_\omega)/a, (\bar{\omega} + \Delta_\omega)/a]$. These two intervals determine a *time-frequency window*. Its width, height and position are governed by a and b. Its area is constant and equal to $4\Delta_x\Delta_\omega$. The Heisenberg uncertainty principle now says that this area has to be bigger than 2. These time-frequency windows are also called *Heisenberg boxes*.

Suppose that the wavelet ψ satisfies the *admissibility condition*

$$C_\psi \;=\; \int_{-\infty}^{+\infty} \frac{|\widehat{\psi}(\omega)|^2}{\omega}\, d\omega \;<\; \infty.$$

Then, the continuous wavelet transform $\mathcal{W}(a, b)$ is invertible on its range, and an inverse transform is given by the relation

$$f(x) \;=\; \frac{1}{C_\psi} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} \mathcal{W}(a, b)\, \psi_{a;b}(x)\, \frac{da\, db}{a^2}. \tag{2.3}$$

From the admissibility condition, we see that $\widehat{\psi}(0)$ has to be 0, and, in particular, ψ has to oscillate. This, together with the decay property, has given ψ the name *wavelet* or "small wave" (French: *ondelette*).

In applications, it is of interest to find inverse transforms that do not make use of $\mathcal{W}(a, b)$ over the whole range of a and b. Transforms exist that only use positive values of a or even only discrete values for a. Furthermore, using the theory of frames, it is possible to study the case where only discrete values for a *and* b are used. See [101] for an excellent overview. The common choice is to use a dyadic grid, i.e. to let $a = 2^{-j}$ and $b/a = l$ with $j, l \in \mathbf{Z}$ [64, 89]. In general, the fewer values of a and b one wants to use, the more restrictive the condition on the wavelet becomes. The continuous wavelet transform allows us to use a very general wavelet. At the other extreme, we shall see that much more restrictive conditions hold for a wavelet used in multiresolution analysis. This allows us, on the other hand, to obtain powerful results such as the construction of orthogonal bases.

The transform that only uses the dyadic values of a and b was originally called the *discrete wavelet transform*. Presently, however, this term is ambiguous, since it is also used to denote the transform from the sequence of scaling function coefficients of a function to its wavelet coefficients (see Section 2.7).

The continuous wavelet transform is used in *singularity detection* and characterization [87, 131]. A typical result in this direction is that if a function f is Hölder continuous of order α, $(0 < \alpha < 1)$, then the continuous wavelet transform has an asymptotic behavior like

$$\mathcal{W}(a, b) \;=\; \mathcal{O}(a^{\alpha+1/2}) \quad \text{for} \quad a \to 0.$$

The converse is true as well. The advantage of this characterization compared to the Fourier transform, is that it does not only provide information about the kind of

singularity, but also about its location in time. A corresponding characterization of Hölder continuous functions of higher order ($\alpha \geqslant 1$) exists; the number of vanishing moments of the wavelet then has to be bigger than α, i.e.

$$\int_{-\infty}^{+\infty} \psi(x)\, x^p\, dx \;=\; 0 \quad \text{for} \quad 0 \leqslant p \leqslant \alpha \quad \text{and} \quad p \in \mathbf{Z}.$$

We note that the number of vanishing wavelet moments limits the order of smoothness that can be characterized.

Example: A classical example of a wavelet is the *Mexican hat* function,

$$\psi(x) \;=\; (1 \Leftrightarrow 2x^2)e^{-x^2}.$$

Being the second derivative of a Gaussian, it has two vanishing moments.

Note: More detailed treatments of the continuous wavelet transform can be found in [34, 95, 96, 101].

2.3 Multiresolution analysis

2.3.1 The scaling function and the subspaces V_j

There are two ways to introduce wavelets: one is through the continuous wavelet transform as mentioned in the previous section, and another is through multiresolution analysis. Here we start by defining multiresolution analysis, and then point out some of the connections with the continuous wavelet transform.

A *multiresolution analysis* of $\mathrm{L}_2(\mathbf{R})$ is defined as a sequence of closed subspaces $V_j \subset \mathrm{L}_2(\mathbf{R})$, $j \in \mathbf{Z}$, with the following properties [63, 134]:

1. $V_j \subset V_{j+1}$,
2. $v(x) \in V_j \Leftrightarrow v(2x) \in V_{j+1}$,
3. $v(x) \in V_0 \Leftrightarrow v(x+1) \in V_0$,
4. $\bigcup_{j=-\infty}^{+\infty} V_j$ is dense in $\mathrm{L}_2(\mathbf{R})$ and $\bigcap_{j=-\infty}^{+\infty} V_j = \{\mathbf{0}\}$,
5. A *scaling function* $\varphi \in V_0$, with a non-vanishing integral, exists so that the collection $\{\varphi(x \Leftrightarrow l) \mid l \in \mathbf{Z}\}$ is a Riesz basis of V_0.

We will use the following terminology: a *level* of a multiresolution analysis is one of the V_j subspaces and one level is *coarser* (respectively *finer*) with respect to another whenever the index of the corresponding subspace is smaller (respectively bigger).

An introduction to the concept of multiresolution analysis and its usefulness can be found in [165, 166].

Let us make a couple of simple observations concerning this definition. Since $\varphi \in V_0 \subset V_1$, a sequence $\{h_k\} \in \ell_2$ exists so that the scaling function satisfies

$$\varphi(x) \;=\; 2\sum_k h_k\, \varphi(2x \Leftrightarrow k). \tag{2.4}$$

This functional equation goes by several different names: the *refinement equation*, the *dilation equation* or the *two-scale difference equation*. We shall use the first.

It also follows immediately that the collection of functions $\{\varphi_{j,l} \mid l \in \mathbf{Z}\}$, with $\varphi_{j,l}(x) = \sqrt{2^j}\,\varphi(2^j x \Leftrightarrow l)$, is a Riesz basis of V_j.

By integrating both sides of (2.4), and dividing by the (non-vanishing) integral of φ, we see that

$$\sum_k h_k \;=\; 1. \tag{2.5}$$

The scaling function is, under very general conditions, uniquely defined by its refinement equation and the normalization [69],

$$\int_{-\infty}^{+\infty} \varphi(x)\, dx \;=\; 1.$$

In many cases, no explicit expression for φ is available. However, there are fast algorithms that use the refinement equation to evaluate the scaling function φ at dyadic points ($x = 2^{-j}k,\ j,k \in \mathbf{Z}$) [21, 32, 63, 165]. In many applications, we never need the scaling function itself; instead we may often work directly with the h_k.

The spaces V_j will be used to approximate general functions. This will be done by defining appropriate projections onto these spaces. Since the union of all the V_j is dense in $\mathrm{L}_2(\mathbf{R})$, we are guaranteed that any given function of L_2 can be approximated arbitrarily close by such projections.

To be able to use the collection $\{\varphi(x \Leftrightarrow l) \mid l \in \mathbf{Z}\}$ to approximate even the simplest functions (such as constants), it is natural to assume that the scaling function and its integer translates form a *partition of unity*, or, in other words,

$$\forall\, x \in \mathbf{R} \;:\; \sum_k \varphi(x \Leftrightarrow k) \;=\; 1. \tag{2.6}$$

Note that by Poisson's summation formula, the partition of unity is (essentially) equivalent to

$$\widehat{\varphi}(2\pi k) \;=\; \delta_k \quad \text{for} \quad k \in \mathbf{Z}. \tag{2.7}$$

By (2.4), the Fourier transform of the scaling function must satisfy

$$\widehat{\varphi}(\omega) \;=\; h(\omega/2)\, \widehat{\varphi}(\omega/2), \tag{2.8}$$

where h is a 2π-periodic function defined by

$$h(\omega) \;=\; \sum_k h_k\, e^{-ik\omega}. \tag{2.9}$$

Since $\widehat{\varphi}(0) = 1$, we can apply (2.8) recursively. This yields, at least formally, the product formula

$$\widehat{\varphi}(\omega) \;=\; \prod_{j=1}^{\infty} h(2^{-j}\omega).$$

The convergence of this product is examined in [41, 63]. The representation of $\widehat{\varphi}$ is nice to have in many situations. For example, it can be used to construct $\varphi(x)$ from the sequence $\{h_k\}$. Using (2.7) and (2.8), we see that we obtain a partition of unity if

$$h(\pi) \;=\; 0 \quad \text{or} \quad \sum_k (\text{-}1)^k\, h_k \;=\; 0.$$

Also note that (2.5) can be written as

$$h(0) \;=\; 1.$$

Examples of scaling functions:

- A well-known family of scaling functions is the set of cardinal B-splines. The cardinal B-spline of order 1 is the box function $N_1(x) = \chi_{[0,1]}(x)$. For $m > 1$ the cardinal B-spline N_m is defined recursively as a convolution:

  $$N_m \;=\; N_{m-1} * N_1.$$

  These functions satisfy

  $$N_m(x) \;=\; 2^{m-1} \sum_{k=0}^{m} \binom{m}{k} N_m(2x \text{-} k),$$

  and

  $$\widehat{N}_m(\omega) \;=\; \left(\frac{1 \text{-} e^{-i\omega}}{i\omega} \right)^m.$$

- Another classical example is the Shannon sampling function,

  $$\varphi(x) \;=\; \frac{\sin(\pi x)}{\pi x} \quad \text{with} \quad \widehat{\varphi}(\omega) \;=\; \chi_{[-\pi,\pi]}(\omega).$$

  We may take

  $$h(\omega) \;=\; \chi_{[-\pi/2,\pi/2]}(\omega) \quad \text{for} \quad \omega \in [\text{-}\pi, \pi],$$

  and, consequently,

  $$h_{2k} \;=\; 1/2\, \delta_k \quad \text{and} \quad h_{2k+1} \;=\; \frac{(\text{-}1)^k}{(2k+1)\pi} \quad \text{for} \quad k \in \mathbf{Z}.$$

Now, for later reference, let us introduce the following 2π-periodic function:

$$b(\omega) \;=\; \sum_k |\widehat{\varphi}(\omega + k2\pi)|^2. \tag{2.10}$$

The fact that φ and its translates form a Riesz basis, corresponds to the fact that there are positive constants A and B so that

$$0 < A \leqslant b(\omega) \leqslant B < \infty.$$

Using (2.8) and rearranging the even and odd terms, we have

$$\begin{aligned} b(2\omega) &= \sum_k |\widehat{\varphi}(2\omega + k2\pi)|^2 \\ &= \sum_k |h(\omega + k\pi)|^2 \, |\widehat{\varphi}(\omega + k\pi)|^2 \\ &= \sum_k |h(\omega + k2\pi)|^2 \, |\widehat{\varphi}(\omega + k2\pi)|^2 + |h(\omega + \pi + k2\pi)|^2 \, |\widehat{\varphi}(\omega + \pi + k2\pi)|^2 \\ &= |h(\omega)|^2 \, b(\omega) + |h(\omega + \pi)|^2 \, b(\omega + \pi). \end{aligned} \tag{2.11}$$

This shows that b is actually π-periodic.

2.3.2 The wavelet function and the detail spaces W_j

Let W_j denote a space complementing V_j in V_{j+1}, i.e. a space that satisfies

$$V_{j+1} \;=\; V_j \oplus W_j.$$

We note that the space W_j is not necessarily unique; there may be several ways to complement V_j in V_{j+1}.

The space W_j contains the "detail" information needed to go from an approximation at resolution j to an approximation at resolution $j+1$. Consequently,

$$\bigoplus_j W_j \;=\; \mathrm{L}_2(\mathbf{R}).$$

A function ψ is a *wavelet* if the collection of functions $\{\psi(x \Leftrightarrow l) \mid l \in \mathbf{Z}\}$ is a Riesz basis of W_0. The collection of wavelet functions $\{\psi_{j,l} \mid l, j \in \mathbf{Z}\}$ is then a Riesz basis of $\mathrm{L}_2(\mathbf{R})$. (The definition of $\psi_{j,l}$ is similar to the one of $\varphi_{j,l}$ in the previous section.) Since the wavelet ψ is an element of V_1, a sequence $\{g_k\} \in \ell_2$ exists so that

$$\psi(x) \;=\; 2\sum_k g_k \, \varphi(2x \Leftrightarrow k). \tag{2.12}$$

The Fourier transform of the wavelet can be written as

$$\widehat{\psi}(\omega) \;=\; g(\omega/2) \, \widehat{\psi}(\omega/2), \tag{2.13}$$

where g is a 2π-periodic function given by

$$g(\omega) \;=\; \sum_k g_k \, e^{-ik\omega}. \tag{2.14}$$

Each space V_j and W_j has a complement in $\mathrm{L}_2(\mathbf{R})$ denoted by V_j^c and W_j^c, respectively. We have:

$$V_j^c \;=\; \bigoplus_{i=j}^{\infty} W_i \quad \text{and} \quad W_j^c \;=\; \bigoplus_{i \neq j} W_i.$$

We define $\mathcal{P}_j$ as the projection operator onto V_j and parallel to V_j^c, and $\mathcal{Q}_j$ as the projection operator onto W_j and parallel to W_j^c. A function f can now be written as

$$f(x) \;=\; \sum_j \mathcal{Q}_j f(x) \;=\; \sum_{j,l} \gamma_{j,l} \, \psi_{j,l}(x).$$

Recalling the discussion in Section 2.2, we see that this last equation is an inverse "discrete" wavelet transform. At this moment, the exact conditions on the wavelet are still unclear. They will be made more precise in the next sections. There, it will also become clear how to find the coefficients $\gamma_{j,l}$. We first turn to the case where the $\psi_{j,l}$ form an orthonormal basis for $\mathrm{L}_2(\mathbf{R})$.

2.4 Orthogonal wavelets

The class of *orthogonal wavelets* is particularly interesting. We start by introducing the concept of an *orthogonal multiresolution analysis*. This is a multiresolution analysis where the wavelet spaces W_j are defined as the *orthogonal* complement of V_j in V_{j+1}. Consequently, the spaces W_j with $j \in \mathbf{Z}$ are all mutually orthogonal, the projections $\mathcal{P}_j$ and $\mathcal{Q}_j$ are orthogonal, and the expansion

$$f(x) \;=\; \sum_j \mathcal{Q}_j f(x)$$

is an orthogonal expansion. A sufficient condition for a multiresolution analysis to be orthogonal is

$$W_0 \;\perp\; V_0,$$

or

$$\langle \, \psi, \varphi(\cdot \Leftrightarrow l) \, \rangle \;=\; 0 \quad \text{for} \quad l \in \mathbf{Z},$$

since the other conditions simply follow from scaling. Using Poisson's summation formula, we see that this condition is (essentially) equivalent to

$$\forall \, \omega \in \mathbf{R} \;:\; \sum_k \widehat{\psi}(\omega + k2\pi) \, \overline{\widehat{\varphi}(\omega + k2\pi)} \;=\; 0. \tag{2.15}$$

An *orthogonal scaling function* is a function φ so that the set $\{\varphi(x \Leftrightarrow l) \mid l \in \mathbf{Z}\}$ is an orthonormal basis, or

$$\langle\, \varphi, \varphi(\cdot \Leftrightarrow l)\,\rangle \;=\; \delta_l \quad \text{for} \quad l \in \mathbf{Z}. \tag{2.16}$$

With such a φ, the collection of functions $\{\varphi(x \Leftrightarrow l) \mid l \in \mathbf{Z}\}$ is an orthonormal basis of V_0 and the collection of functions $\{\varphi_{j,l} \mid l \in \mathbf{Z}\}$ is an orthonormal basis of V_j. Using Poisson's formula, (2.16) is (essentially) equivalent to

$$\forall\, \omega \in \mathbf{R} \;:\; \sum_k |\widehat{\varphi}(\omega + k2\pi)|^2 \;=\; f(\omega) \;=\; 1. \tag{2.17}$$

From (2.11) we now see that,

$$\forall\, \omega \in \mathbf{R} \;:\; |h(\omega)|^2 + |h(\omega + \pi)|^2 \;=\; 1, \tag{2.18}$$

or

$$\sum_k h_k\, h_{k-2l} \;=\; \delta_l/2 \quad \text{for} \quad l \in \mathbf{Z}.$$

The last two equations are equivalent, but they only provide a necessary condition for the orthogonality of the scaling function and its translates. This relationship is investigated in detail in [42, 122].

Now, an *orthogonal wavelet* is a function ψ so that the collection of functions $\{\psi(x \Leftrightarrow l) \mid l \in \mathbf{Z}\}$ is an orthonormal basis of W_0. This is the case if

$$\langle\, \psi, \psi(\cdot \Leftrightarrow l)\,\rangle \;=\; \delta_l.$$

Again these conditions are (essentially) equivalent to

$$\forall\, \omega \in \mathbf{R} \;:\; \sum_k |\widehat{\psi}(\omega + k2\pi)|^2 \;=\; 1,$$

and, using a similar argument as above, a necessary condition is given by

$$\forall\, \omega \in \mathbf{R} \;:\; |g(\omega)|^2 + |g(\omega + \pi)|^2 \;=\; 1.$$

Since the spaces W_j are mutually orthogonal, the collection of functions $\{\psi_{j,l} \mid j, l \in \mathbf{Z}\}$ is an orthonormal basis of $\mathrm{L}_2(\mathbf{R})$.

The projection operators $\mathcal{P}_j$ and $\mathcal{Q}_j$ can now be written as

$$\mathcal{P}_j f(x) \;=\; \sum_l \langle\, f, \varphi_{j,l}\,\rangle\, \varphi_{j,l}(x) \quad \text{and} \quad \mathcal{Q}_j f(x) \;=\; \sum_l \langle\, f, \psi_{j,l}\,\rangle\, \psi_{j,l}(x).$$

They yield the best L_2 approximations of the function f in V_j and W_j, respectively. For a function $f \in \mathrm{L}_2(\mathbf{R})$ we have the orthogonal expansion

$$f(x) \;=\; \sum_{j,l} \gamma_{j,l}\, \psi_{j,l}(x) \quad \text{with} \quad \gamma_{j,l} \;=\; \langle\, f, \psi_{j,l}\,\rangle\,.$$

Again, this can be viewed as a discrete version of the continuous wavelet transform. Examples of orthogonal wavelets will be given in Section 2.8.

Using (2.17) we can write the condition (2.15) as

$$\forall\, \omega \in \mathbf{R} \;:\; g(\omega)\,\overline{h(\omega)} + g(\omega+\pi)\,\overline{h(\omega+\pi)} \;=\; 0. \tag{2.19}$$

From this last equation it follows that the function $g(\omega)$ needs to be of the form

$$g(\omega) \;=\; a(\omega)\,\overline{h(\omega+\pi)},$$

where a is a 2π-periodic function so that

$$a(\omega+\pi) \;=\; \Leftrightarrow a(\omega).$$

The orthogonality of the wavelet immediately follows from the orthogonality of the scaling function if

$$|a(\omega)| \;=\; 1.$$

As we will see later on, it is important for the scaling function and wavelet to have compact support. The compact support of the wavelet and scaling function is equivalent with the fact that h and g are trigonometric polynomials (i.e. the sums in (2.9) and (2.14) are finite). In the above case, we see that if the scaling function is compactly supported, so is the wavelet, provided that a is a trigonometric polynomial. The only trigonometric polynomials that satisfy the conditions for a are monomials of the form,

$$C\,e^{-(2k+1)\omega i} \quad \text{with} \quad |C| = 1 \quad \text{and} \quad k \in \mathbf{Z}.$$

Up to the constant C and an integer translation, the different choices for a all give rise to the same wavelet. Any other choice for a will lead to a wavelet without compact support. If the coefficients h_k are real, so are the g_k if $C = \pm 1$. The standard choice is $a(\omega) = \Leftrightarrow e^{-i\omega}$. This means that we derive an orthogonal wavelet from an orthogonal scaling function by choosing

$$g_k \;=\; (\Leftrightarrow 1)^k\,\overline{h_{1-k}}. \tag{2.20}$$

This still leaves us with the problem of constructing a compactly supported scaling function. We will comment on this in Section 2.6.

In [125], an orthogonalization procedure to find orthonormal wavelets is proposed. It states that if a function φ and its integer translates form a Riesz basis of V_0, then an orthonormal basis of V_0 is given by φ_{orth} and its integer translates with

$$\widehat{\varphi}_{\text{orth}}(\omega) \;=\; \frac{\widehat{\varphi}(\omega)}{\sqrt{b(\omega)}}, \tag{2.21}$$

where $b(\omega)$ is the function defined in (2.10). The fact that we started from a Riesz basis guarantees that $b(\omega)$ is strictly positive. We see that φ indeed satisfies the orthogonality condition (2.17). Note that if φ is compactly supported, φ_{orth} will, in general, not be compactly supported.

2.5 Biorthogonal wavelets

The orthogonality property puts a strong limitation on the construction of wavelets. For example, it is known that the Haar wavelet is the only real-valued wavelet that is compactly supported, symmetric and orthogonal [63]. The generalization to biorthogonal wavelets has been considered to gain more flexibility. Here, a dual scaling function $\widetilde{\varphi}$ and a dual wavelet $\widetilde{\psi}$ exist. They generate a dual multiresolution analysis with subspaces $\widetilde{V}_j$ and $\widetilde{W}_j$, so that

$$\widetilde{V}_j \perp W_j \quad \text{and} \quad V_j \perp \widetilde{W}_j, \tag{2.22}$$

and, consequently,

$$\widetilde{W}_j \perp W_{j'} \quad \text{for} \quad j \neq j'.$$

The dual multiresolution analysis is not necessarily the same as the one generated by the original basis functions. A condition equivalent to (2.22) is:

$$\langle\, \widetilde{\varphi}, \psi(\cdot \Leftrightarrow l) \,\rangle \;=\; \langle\, \widetilde{\psi}, \varphi(\cdot \Leftrightarrow l) \,\rangle \;=\; 0.$$

Moreover, the dual functions also have to satisfy

$$\langle\, \widetilde{\varphi}, \varphi(\cdot \Leftrightarrow l) \,\rangle \;=\; \delta_l \quad \text{and} \quad \langle\, \widetilde{\psi}, \psi(\cdot \Leftrightarrow l) \,\rangle \;=\; \delta_l.$$

By using a scaling argument, we have the seemingly more general properties that

$$\langle\, \widetilde{\varphi}_{j,l}, \varphi_{j,l'} \,\rangle \;=\; \delta_{l-l'} \quad \text{for} \quad l, l', j \in \mathbf{Z} \tag{2.23}$$

and

$$\langle\, \widetilde{\psi}_{j,l}, \psi_{j',l'} \,\rangle \;=\; \delta_{j-j'}\delta_{l-l'} \quad \text{for} \quad l, l', j, j' \in \mathbf{Z}. \tag{2.24}$$

Here the definitions of $\widetilde{\varphi}_{j,l}$ and $\widetilde{\psi}_{j,l}$ are similar to the ones for $\varphi_{j,l}$ and $\psi_{j,l}$. Note that the role of the basis (i.e. the φ and ψ) and the dual basis can be interchanged. Using the same Fourier techniques as in the previous section, the biorthogonality conditions are (essentially) equivalent to

$$\forall \omega \in \mathbf{R} \;:\; \begin{cases} \displaystyle\sum_k \widehat{\widetilde{\varphi}}(\omega + k2\pi)\, \overline{\widehat{\varphi}(\omega + k2\pi)} &= 1 \\ \displaystyle\sum_k \widehat{\widetilde{\psi}}(\omega + k2\pi)\, \overline{\widehat{\psi}(\omega + k2\pi)} &= 1 \\ \displaystyle\sum_k \widehat{\widetilde{\psi}}(\omega + k2\pi)\, \overline{\widehat{\varphi}(\omega + k2\pi)} &= 0 \\ \displaystyle\sum_k \widehat{\widetilde{\varphi}}(\omega + k2\pi)\, \overline{\widehat{\psi}(\omega + k2\pi)} &= 0. \end{cases} \tag{2.25}$$

Since they define a multiresolution analysis, the dual functions must satisfy

$$\widetilde{\varphi}(x) \;=\; 2\sum_k \widetilde{h}_k\, \widetilde{\varphi}(2x \Leftrightarrow k) \quad \text{and} \quad \widetilde{\psi}(x) \;=\; 2\sum_k \widetilde{g}_k\, \widetilde{\varphi}(2x \Leftrightarrow k). \tag{2.26}$$

If we define the functions $\widetilde{h}$ and $\widetilde{g}$ in the same fashion as we did for h and g, then necessary conditions are again given by

$$\forall \omega \in \mathbf{R} \; : \; \left\{ \begin{array}{rcrcl} \widetilde{h}(\omega)\,\overline{h(\omega)} & + & \widetilde{h}(\omega+\pi)\,\overline{h(\omega+\pi)} & = & 1 \\ \widetilde{g}(\omega)\,\overline{g(\omega)} & + & \widetilde{g}(\omega+\pi)\,\overline{g(\omega+\pi)} & = & 1 \\ \widetilde{g}(\omega)\,\overline{h(\omega)} & + & \widetilde{g}(\omega+\pi)\,\overline{h(\omega+\pi)} & = & 0 \\ \widetilde{h}(\omega)\,\overline{g(\omega)} & + & \widetilde{h}(\omega+\pi)\,\overline{g(\omega+\pi)} & = & 0, \end{array} \right. \tag{2.27}$$

or

$$\forall \omega \in \mathbf{R} \; : \; \begin{bmatrix} \widetilde{h}(\omega) & \widetilde{h}(\omega+\pi) \\ \widetilde{g}(\omega) & \widetilde{g}(\omega+\pi) \end{bmatrix} \overline{\begin{bmatrix} h(\omega) & g(\omega) \\ h(\omega+\pi) & g(\omega+\pi) \end{bmatrix}} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Hence, if we let

$$m(\omega) \; = \; \begin{bmatrix} h(\omega) & h(\omega+\pi) \\ g(\omega) & g(\omega+\pi) \end{bmatrix},$$

and similarly for $\widetilde{m}$, then

$$\widetilde{m}(\omega)\,\overline{m^t(\omega)} \; = \; \mathbf{1}.$$

By interchanging the matrices on the left-hand side, we get

$$\forall \omega \in \mathbf{R} \; : \; \left\{ \begin{array}{rcrcl} \overline{h(\omega)}\,\widetilde{h}(\omega) & + & \overline{g(\omega)}\,\widetilde{g}(\omega) & = & 1 \\ \overline{h(\omega)}\,\widetilde{h}(\omega+\pi) & + & \overline{g(\omega)}\,\widetilde{g}(\omega+\pi) & = & 0. \end{array} \right. \tag{2.28}$$

Note that the orthogonal case corresponds to m being a unitary matrix. Cramer's rule now states that

$$\widetilde{h}(\omega) \; = \; \frac{\overline{g(\omega+\pi)}}{\overline{\delta(\omega)}} \tag{2.29}$$

and

$$\widetilde{g}(\omega) \; = \; \Leftrightarrow \frac{\overline{h(\omega+\pi)}}{\overline{\delta(\omega)}}, \tag{2.30}$$

where

$$\delta(\omega) \; = \; \det m(\omega).$$

The space generated by the set of functions $\{\psi_{j,l} \mid l \in \mathbf{Z}\}$ complements V_j in V_{j+1} if and only if $\delta(\omega)$ does not vanish.

The projection operators take the form

$$\mathcal{P}_j f(x) \; = \; \sum_l \langle\, f, \widetilde{\varphi}_{j,l} \,\rangle\, \varphi_{j,l}(x) \quad \text{and} \quad \mathcal{Q}_j f(x) \; = \; \sum_l \langle\, f, \widetilde{\psi}_{j,l} \,\rangle\, \psi_{j,l}(x),$$

so that

$$f \; = \; \sum_{j,l} \langle\, f, \widetilde{\psi}_{j,l} \,\rangle\, \psi_{j,l}.$$

Note that this can be viewed as a "discrete" wavelet transform and that the conditions on ψ are less restrictive than in the orthogonal case. From the equations (2.23), (2.24), and (2.26) we see that

$$\widetilde{h}_{k-2l} \;=\; \langle\, \widetilde{\varphi}(x \Leftrightarrow l), \varphi(2x \Leftrightarrow k)\,\rangle \quad \text{and} \quad \widetilde{g}_{k-2l} \;=\; \langle\, \widetilde{\psi}(x \Leftrightarrow l), \varphi(2x \Leftrightarrow k)\,\rangle\,.$$

In particular, by writing $\varphi(2x \Leftrightarrow k) \in V_1$ in the bases of V_0 and W_0, we obtain

$$\varphi(2x \Leftrightarrow k) \;=\; \sum_l \widetilde{h}_{k-2l}\, \varphi(x \Leftrightarrow l) + \sum_l \widetilde{g}_{k-2l}\, \psi(x \Leftrightarrow l). \tag{2.31}$$

Even if the scaling function and the wavelet are not orthogonal, the multiresolution analysis may still be orthogonal. Let us study this in more detail.

A biorthogonal scaling function and wavelet are *semiorthogonal* if they generate an orthogonal multiresolution analysis [4, 6, 34]. (The name *pre-wavelet* is also used for such a wavelet.) Since the W_j subspaces are mutually orthogonal we have that

$$W_j \perp \widetilde{W}_{j'} \quad \text{and} \quad W_j \perp W_{j'} \quad \text{for} \quad j \neq j'.$$

Consequently, $W_j = \widetilde{W}_j$, which implies that $V_j = \widetilde{V}_j$. Thus, the primary and dual functions generate the same (orthogonal) multiresolution analysis. A dual scaling function can now be found by letting

$$\widehat{\widetilde{\varphi}}(\omega) \;=\; \frac{\widehat{\varphi}(\omega)}{b(\omega)}.$$

Here $b(\omega)$ again is the function defined in (2.10). We see that the first equation of (2.25) is satisfied, and, since b is a bounded, 2π-periodic function that does not vanish, the translates of φ and $\widetilde{\varphi}$ generate the same space. This corresponds to:

$$\widetilde{h}(\omega) \;=\; \frac{h(\omega)\, b(\omega)}{b(2\omega)}.$$

In order to have an orthogonal multiresolution analysis, (2.19) must also be satisfied. As before, this means that we need to pick g so that

$$g(\omega) \;=\; a(\omega)\, \overline{h(\omega + \pi)},$$

where a is a 2π-periodic function with

$$a(\omega + \pi) \;=\; \Leftrightarrow a(\omega).$$

If a is a trigonometric polynomial, then the scaling function is compactly supported. By looking at the last equation of (2.27) it is clear that a simple choice is

$$a(\omega) = \Leftrightarrow e^{-i\omega}\, b(\omega + \pi),$$

so that

$$\delta(\omega) \;=\; e^{-i\omega}\, b(2\omega),$$

and, consequently,

$$\widetilde{g}(\omega) \;=\; \Leftrightarrow e^{-i\omega}\, \frac{\overline{h(\omega+\pi)}}{b(2\omega)}.$$

If φ is compactly supported, this construction guarantees that ψ is compactly supported too, since h and b, and hence also g, are trigonometric polynomials. Note, however, that the dual functions are not compactly supported.

2.6 Wavelets and polynomials

The moments of the scaling function and wavelet are defined by:

$$\mathcal{M}_p \;=\; \int_{-\infty}^{+\infty} x^p\, \varphi(x)\, dx \quad \text{and} \quad \mathcal{N}_p \;=\; \int_{-\infty}^{+\infty} x^p\, \psi(x)\, dx \quad \text{with} \quad p \in \mathbf{N},$$

and similarly for the dual functions. The scaling functions are normalized with $\mathcal{M}_0 = \widetilde{\mathcal{M}}_0 = 1$.

Recall that we want the scaling function to satisfy a "partition of unity" property and, furthermore, that this corresponds to $h(\pi) = 0$. From (2.30) we see that this implies that $\widetilde{g}(0) = 0$ and, hence, that $\widetilde{\mathcal{N}}_0 = 0$. Thus, the dual wavelet needs to have a vanishing integral. This is reminiscent of the case of the continuous wavelet transform where we needed the wavelet to have a vanishing integral.

As we pointed out before, the fact that the wavelet has a vanishing integral allows us to give a precise characterization of the functions with a certain smoothness (when the order of smoothness α is less than 1), in terms of the decay of the continuous wavelet transform. The analogous fact is true here. The wavelet coefficients are given by inner products with the dual wavelets. The fact that these have a vanishing integral allows us to characterize exactly which functions will be of a certain smoothness by looking at the decay of the coefficients.

As in the case of the continuous wavelet transform, to obtain similar characterizations of classes of functions of smoothness $\alpha > 1$, the dual wavelet needs to have more vanishing moments. This is closely related to the property that the scaling function and its translates can represent polynomials. We make this statement more precise.

Let N denote the number of vanishing moments of the dual wavelet,

$$\widetilde{\mathcal{N}}_p \;=\; 0 \quad \text{for} \quad 0 \leqslant p < N \quad \text{and} \quad \widetilde{\mathcal{N}}_N \neq 0.$$

This is the same as saying that $\widehat{\widetilde{\psi}}(\omega)$ has a root of multiplicity N at $\omega = 0$. Since $\widehat{\widetilde{\varphi}}(0) \neq 0$, it is also equivalent to the fact that $\widetilde{g}(\omega)$ has a root of multiplicity N at $\omega = 0$. Thus, the sequence $\{\widetilde{g}_k\}$ also has N vanishing discrete moments,

$$\sum_k \widetilde{g}_k\, k^p = 0, \quad \text{for} \quad 0 \leqslant p < N.$$

From (2.30), we see that this is equivalent to $h(\omega)$ having a root of multiplicity N at $\omega = \pi$, which, by using (2.8), implies that

$$i^p \widehat{\varphi}^{(p)}(2k\pi) \;=\; \delta_k \, \mathcal{M}_p \quad \text{for} \quad 0 \leqslant p < N. \tag{2.32}$$

By Poisson's summation formula, it follows that

$$\sum_l (x \Leftrightarrow l)^p \, \varphi(x \Leftrightarrow l) \;=\; \mathcal{M}_p \quad \text{for} \quad 0 \leqslant p < N. \tag{2.33}$$

Rearranging the last expression, we see that any polynomial with degree smaller than N can be written as a linear combination of the functions $\varphi(x \Leftrightarrow l)$ with $l \in \mathbf{Z}$.

At this point we digress a little and make two small remarks.

1. The fact that $h(\omega)$ has a root of multiplicity N at $\omega = \pi$, means that we can factor $h(\omega)$ as
$$h(\omega) \;=\; \left(\frac{1 + e^{-i\omega}}{2} \right)^N K(\omega),$$
with $K(0) = 1$ and $K(\pi) \neq 0$. This factorization, together with the orthogonality conditions and the fact that K is a trigonometric polynomial, is used as a starting point for the construction of compactly supported wavelets [45, 63].

2. When writing a polynomial as a linear combination of the $\varphi(x \Leftrightarrow l)$, the coefficients in the linear combination themselves are polynomials of the same degree in l. More precisely, if A is a polynomial of degree $N \Leftrightarrow 1$, then a polynomial B, of the same degree, exists so that
$$A(x) \;=\; \sum_l B(l) \, \varphi(x \Leftrightarrow l). \tag{2.34}$$
The fact that B is a polynomial can be seen from
$$B(l) \;=\; \int A(x) \, \widetilde{\varphi}(x \Leftrightarrow l) \, dx \;=\; \int A(x + l) \, \widetilde{\varphi}(x) \, dx.$$
Furthermore,
$$A(x) \;=\; \sum_l B(x \Leftrightarrow l) \, \varphi(l),$$
since the polynomials on the left and right-hand sides match at each integer.

With the extra vanishing moment conditions on the dual wavelet, we can characterize smoothness up to order $\alpha < N$. Another consequence is that the convergence rate of the wavelet approximation for smooth functions now immediately follows: if $f \in \mathcal{C}^N$, then

$$\| \mathcal{P}_j f(x) \Leftrightarrow f(x) \| \;=\; \mathcal{O}(h^N) \quad \text{with} \quad h = 2^{-j}. \tag{2.35}$$

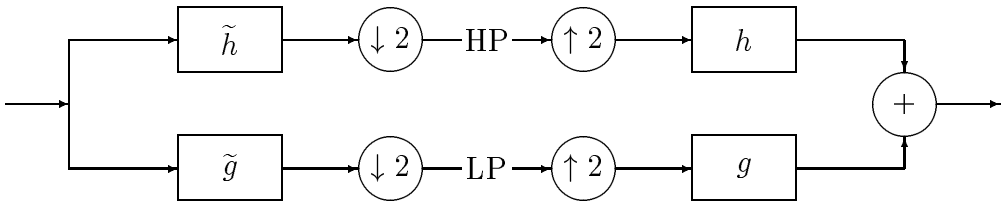

Figure 2.1: The subband filtering scheme.

The conditions (2.32) are referred to as the Strang–Fix conditions, and these were established long before the development of wavelet theory [85, 165, 168].

In Chapter 4, we will derive an asymptotic error expansion in powers of h, which can be used in numerical extrapolation.

The exponent N in the factorization of h also plays a role in the regularity of φ. The Hölder regularity is $N \Leftrightarrow 1$ at most, but in many cases it is lower due to the influence of K. The regularity of solutions of refinement equations is studied in detail in [55, 56, 69, 70, 82, 152, 191, 190].

Note that we never required the dual scaling function to satisfy a partition of unity, nor the wavelet to have a vanishing moment. It is indeed possible to have a wavelet with a non-vanishing integral. In that case, the regularity of the dual functions is very low. It may even be that they are distributions instead of functions, but this is not necessarily a problem in applications. We will encounter such cases in Chapters 3 and 6.

2.7 The fast wavelet transform

Since V_j is equal to $V_{j-1} \oplus W_{j-1}$, a function $v_j \in V_j$ can be written uniquely as the sum of a function $v_{j-1} \in V_{j-1}$ and a function $w_{j-1} \in W_{j-1}$:

$$\begin{aligned} v_j(x) &= \sum_k \lambda_{j,k}\, \varphi_{j,k}(x) \;=\; v_{j-1}(x) + w_{j-1}(x) \\ &= \sum_l \lambda_{j-1,l}\, \varphi_{j-1,l}(x) + \sum_l \gamma_{j-1,l}\, \psi_{j-1,l}(x). \end{aligned}$$

In other words, we have two representations of the function v_j, one as an element in V_j and associated with the sequence $\{\lambda_{j,k}\}$, and another as a sum of elements in V_{j-1} and W_{j-1} and associated with the sequences $\{\lambda_{j-1,k}\}$ and $\{\gamma_{j-1,k}\}$. The following relations show how to pass between these representations. By (2.26),

$$\begin{aligned} \lambda_{j-1,l} &= \langle\, v_j, \widetilde{\varphi}_{j-1,l}\,\rangle \;=\; \sqrt{2}\,\langle\, v_j, \sum_k \widetilde{h}_{k-2l}\, \widetilde{\varphi}_{j,k}\,\rangle \\ &= \sqrt{2} \sum_k \widetilde{h}_{k-2l}\, \lambda_{j,k}, \end{aligned} \tag{2.36}$$

and, similarly,

$$\gamma_{j-1,l} = \sqrt{2} \sum_k \widetilde{g}_{k-2l}\, \lambda_{j,k}. \tag{2.37}$$

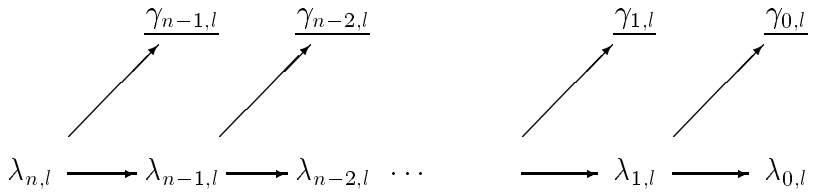

Figure 2.2: The decomposition scheme.

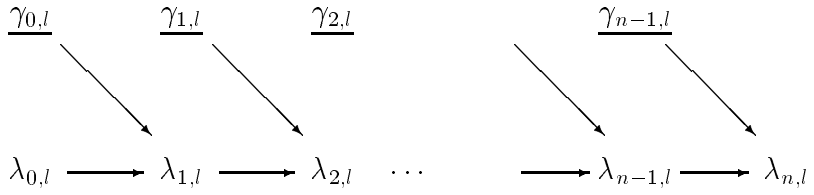

Figure 2.3: The reconstruction scheme.

The opposite direction, from $\{\lambda_{j-1,l}\}$ and $\{\gamma_{j-1,l}\}$ to $\{\lambda_{j,k}\}$, is equally easy. Using (2.31) we have

$$\lambda_{j,k} = \sqrt{2} \sum_l h_{k-2l}\, \lambda_{j-1,l} + \sqrt{2} \sum_l g_{k-2l}\, \gamma_{j-1,l}. \tag{2.38}$$

When applied recursively, these formulae define the *fast wavelet transform*; the relations (2.36) and (2.37) define the forward transform, while (2.38) defines the inverse transform.

Now, from the fact that $h(0) = g(\pi) = 1$ and $g(0) = h(\pi) = 0$, we see that $h(\omega)$ acts like a low pass filter for the interval $[0, \pi/2]$ and $g(\omega)$ similarly behaves like a band pass filter for the interval $[\pi/2, \pi]$. Equation (2.8) (respectively (2.13)) then implies that the major part of the energy of the functions in V_0 (respectively W_0) is concentrated in the intervals $[0, \pi]$ (respectively $[\pi, 2\pi]$). The behavior of the dual functions is the same. In an approximate sense, this means that the wavelet expansion splits the frequency space into dyadic blocks $[2^j\pi, 2^{j+1}\pi]$ with $j \in \mathbf{Z}$ [134, 135].

In signal processing, this idea is known as subband filtering, or, more specifically, as quadrature mirror filtering. Quadrature mirror filters were studied before wavelet theory. The decomposition step consists of applying a low-pass ($\widetilde{h}$) and a band-pass ($\widetilde{g}$) filter followed by downsampling ($\downarrow 2$) (i.e. retaining only the even index samples), see Figure 2.1. The reconstruction consists of upsampling ($\uparrow 2$) (i.e. putting a zero between every two samples) followed by filtering and addition. One can show that the conditions (2.28) correspond to the exact reconstruction of a subband filtering scheme. More details can be found in [153, 185, 188, 189].

An interesting problem now is; given a function f, determine, with a certain

accuracy and in a computationally favorable way, the coefficients $\lambda_{n,l}$ of a function in the space V_n, which are needed to start the fast wavelet transform. This problem is studied in Chapter 3, where solutions such as (quasi-)interpolation and quadrature formulae are introduced.

2.8 Examples of wavelets

Now that we have discussed the essentials of wavelet multiresolution analysis, we take a look at some important properties of wavelets.

Orthogonality: Orthogonality is convenient to have in many situations, e.g. it directly links the L_2 norm of a function to the norm of its wavelet coefficients by

$$\|f\| = \sqrt{\sum_{j,l} \gamma_{j,l}^2}.$$

In the biorthogonal case these two quantities are only equivalent. Another advantage of orthogonal wavelets is that the fast wavelet transform is a unitary transformation. Consequently, its condition number is equal to one, which is the optimal case.

If the multiresolution analysis is orthogonal (remember that this includes semi-orthogonal wavelets), the projection operators onto the different subspaces yield optimal approximations in the L_2 sense.

Compact support: If the scaling function and wavelet are compactly supported, the filters h and g are finite impulse response filters, so that the summations in the fast wavelet transform are finite. This obviously is of use in implementations. If they are not compactly supported, a fast decay is desirable so that the filters can be approximated reasonably by finite impulse response filters.

Rational coefficients: For computer implementations it is of use if the filter coefficients h_k and g_k are rationals or, even better, dyadic rationals. Multiplication by a power of two on a computer corresponds to shifting bits, which is a very fast operation.

Symmetry: If the scaling function and wavelet are symmetric, then the filters have generalized linear phase. The absence of this property can lead to phase distortion. This is important in signal processing applications.

Smoothness: The smoothness of wavelets plays an important role in compression applications. Compression is usually achieved by setting small coefficients $\gamma_{j,l}$ to zero, and thus leaving out a component $\gamma_{j,l}\,\psi_{j,l}$ from the original function.

If the original function represents an image and the wavelet is not smooth, the error can easily be detected visually. Note that the smoothness of the primary functions is more important to this aspect than that of the dual. Also, a higher degree of smoothness corresponds to better frequency localization of the filters. Finally, smooth basis functions are desired in numerical analysis applications where derivatives are involved.

Number of vanishing moments of the dual wavelet: We saw earlier that this is important in singularity detection and characterization of smoothness spaces. Also, it determines the convergence rate of wavelet approximations of smooth functions. Finally, the number of vanishing moments of the dual wavelet is connected to the smoothness of the wavelet (and vice versa).

Analytic expressions: As previously noted, an analytic expression for a scaling function or wavelet does not always exists but in some cases it is available and nice to have. In harmonic analysis, analytic expressions of the Fourier transform are particularly useful.

Interpolation: If the scaling function satisfies

$$\varphi(k) \;=\; \delta_k \quad \text{for} \quad k \in \mathbf{Z},$$

then it is trivial to find the function of V_j that interpolates data sampled on a grid with spacing 2^{-j}, since the coefficients are equal to the samples.

As could be expected, it is not possible to construct wavelets that have all these properties. We have a classic trade-off situation. We take a look at several compromises.

Examples of orthogonal wavelets:

- Two simple examples of orthogonal scaling functions are the box function — $\chi_{[0,1]}(x)$, and the Shannon sampling function — $\mathrm{sinc}(\pi x)$. The orthogonality conditions are easy to verify, either in the time or frequency space. The corresponding wavelet for the box function is the *Haar wavelet*

  $$\psi_{\mathrm{Haar}}(x) \;=\; \chi_{[0,1/2]}(x) \Leftrightarrow \chi_{[1/2,1]}(x),$$

  and the *Shannon wavelet* is

  $$\psi_{\mathrm{Shannon}}(x) \;=\; \frac{\sin(2\pi x) \Leftrightarrow \sin(\pi x)}{\pi x}.$$

  These two, however, are not very useful in practice, since the first has very low regularity and the second has very slow decay.

- A more interesting example is the *Meyer wavelet* and scaling function [140]. These functions belong to $\mathcal{C}^\infty$ and have faster than polynomial decay. Their Fourier transform is compactly supported. The scaling function and wavelet are symmetric around 0 and 1/2, respectively, and the wavelet has an infinite number of vanishing moments. The construction of these functions was done on the Fourier transform side. In Chapter 7, we will discuss such constructions and their applications in more detail.

- The *Battle-Lemarié wavelets* are constructed by orthogonalizing B-spline functions using (2.21) and have exponential decay [19, 125]. The wavelet with N vanishing moments is a piecewise polynomial of degree $N \Leftrightarrow 1$ that belongs to $\mathcal{C}^{N-2}$.

- Probably the most frequently used orthogonal wavelets are the original *Daubechies wavelets* [63, 65]. They are a family of orthogonal wavelets indexed by $N \in \mathbf{N}$, where N is the number of vanishing wavelet moments. They are supported on an interval of length $2N \Leftrightarrow 1$. A disadvantage is that, except for the Haar wavelet (which has $N = 1$), they cannot be symmetric or antisymmetric. Their regularity increases linearly with N and is approximately equal to $0.2075N$ for large N. In [66], three variations of this family, all with orthogonal and compactly supported functions, are constructed:

  1. The previous construction does not lead to a unique solution if N and the support length are fixed. One family is constructed by choosing, for each N, the solution which has closest to linear phase (which is most symmetric). The original family corresponds to choosing the extremal phase.
  2. Another family has more regularity, at the price of a slightly larger support length $(2N + 1)$.
  3. In a third family, the scaling function also has vanishing moments ($\mathcal{M}_p = 0$ for $0 < p < N$). This is of use in numerical analysis applications where inner products of arbitrary functions with scaling functions have to be calculated very quickly [25]. Their construction was requested by Ronald Coifman, and Ingrid Daubechies therefore named them *coiflets*. They are supported on an interval with length $3N \Leftrightarrow 1$.

Examples of biorthogonal wavelets:

- Biorthogonal wavelets were constructed by Albert Cohen, Ingrid Daubechies and Jean-Christophe Feauveau in [41, 45]. Here, $\delta(\omega)$ is chosen equal to $e^{-i\omega}$, and thus

$$g(\omega) \;=\; \Leftrightarrow e^{-i\omega}\,\overline{\tilde{h}(\omega+\pi)} \quad \text{and} \quad \tilde{g}(\omega) \;=\; \Leftrightarrow e^{-i\omega}\,\overline{h(\omega+\pi)}.$$

Table 2.1: A quick comparison of wavelet families.

wavelet	compact support		analytic expression		symmetry	orthogonality		compact
family	primary	dual	primary	dual		semi	full	support $\widehat{\psi}$
a	x	x	o	o	o	x	x	o
b	x	x	x	o	x	o	o	o
c	x	o	x	x	x	x	o	o
d	o	o	o	o	x	x	x	x
e	o	o	x	x	x	x	x	o

a: Daubechies' orthogonal wavelets
b: biorthogonal spline wavelets
c: semiorthogonal spline wavelets
d: Meyer wavelet
e: orthogonal spline wavelets

The scaling functions are the cardinal B-splines and the wavelets also are spline functions. All functions including the dual ones have compact support and linear phase. Moreover, all filter coefficients are dyadic rationals. A disadvantage is that for small filter lengths, the dual functions have very low regularity.

- Semiorthogonal spline wavelets were constructed by Charles Chui and Jianzhong Wang in [37, 38, 39]. The scaling functions are cardinal B-splines of order m and the wavelet functions are splines with support $[0, 2m \Leftrightarrow 1]$. All primary and dual functions still have generalized linear phase and all coefficients used in the fast wavelet transform are rationals. A powerful feature here is that analytic expressions for the wavelet, scaling function, and dual functions are available. A disadvantage is that the dual functions do not have compact support, but have exponential decay instead. The same wavelets, but in a different setting, were also derived by Akram Aldroubi, Murray Eden and Michael Unser in [181, 183]. They also showed that, for N going to infinity, the spline wavelets converge to Gabor functions [182].

- Other semiorthogonal wavelets can be found in [115, 142, 143, 151]. A characterization of all semiorthogonal wavelets is given in [4, 6].

The properties of some of the orthogonal, biorthogonal and semiorthogonal wavelet families are summarized in Table 2.1.

Examples of interpolating scaling functions:

- The Shannon sampling function

$$\varphi_{\text{Shannon}} = \frac{\sin(\pi x)}{\pi x},$$

  is an interpolating scaling function. It is band-limited, but it has very slow decay.

- Constructions of interpolating functions will be discussed in Chapter 3.

2.9 Wavelets on an interval

So far we have been discussing wavelet theory on the real line. For many applications, the functions involved are only defined on a compact set, such as an interval, and to apply wavelets then requires some modifications.

2.9.1 Simple solutions

To be specific, let us discuss the case of the unit interval $[0,1]$. Given a function f on $[0,1]$, the most obvious approach is to set $f(x) = 0$ outside $[0,1]$, and then use wavelet theory on the line. However, for a general function f this "padding with 0" introduces discontinuities at the endpoints 0 and 1; consider for example the simple function $f(x) = 1$, $x \in [0,1]$. Now, as we have said earlier, wavelets are effective for detecting singularities, so artificial discontinuities are likely to introduce significant errors.

Another approach, which is often better, is to consider the function to be periodic with period one, $f(x+1) = f(x)$. In other words, we assume that the function is defined on the circle and identify the circle with $[0,1]$. Wavelet theory on the circle parallels that on the line. Note that if f has period one, then the wavelet coefficients on a given scale satisfy $\langle f, \psi_{j,k} \rangle = \langle f, \psi_{j,k+2^j} \rangle$, $k \in \mathbf{Z}$, $j \geqslant 0$. This simple observation readily allows us to rewrite wavelet expansions on the line as analogous ones on the circle, with wavelets defined on $[0,1]$. A periodic multiresolution analysis on the interval $[0,1]$ can be constructed by periodizing the basis functions as follows,

$$\varphi^*_{j,l}(x) = \chi_{[0,1]}(x) \sum_m \varphi_{j,l}(x+m) \quad \text{for} \quad 0 \leqslant l < 2^j \quad \text{and} \quad j \geqslant 0. \tag{2.39}$$

If the support of $\varphi_{j,l}$, is a subset of $[0,1]$, then $\varphi^*_{j,l} = \varphi_{j,l}$. Otherwise $\varphi_{j,l}$ is chopped into pieces of length one, which are shifted onto $[0,1]$ and added up, yielding $\varphi^*_{j,l}$. Similar definitions hold for $\psi^*_{j,l}$, $\widetilde{\varphi}^*_{j,l}$ and $\widetilde{\psi}^*_{j,l}$. This "wrap around" procedure is satisfactory in many situations (and certainly takes care of functions like $f(x) = 1$, $x \in [0,1]$). However, unless the behavior of the function f at 0 matches that at 1,

the periodic version of f has singularities there. A simple function like $f(x) = x$, $x \in [0,1]$, gives a good illustration of this.

A third method, which works if the basis functions are symmetric, is to use reflection across the edges. This preserves continuity, but introduces discontinuities in the first derivative. This solution is sometimes satisfactory in image processing applications.

2.9.2 Meyer's boundary wavelets

What really is needed, are wavelets intrinsically defined on $[0,1]$. We sketch a construction of orthogonal wavelets on $[0,1]$, recently presented by Yves Meyer [141]. We start from an orthogonal Daubechies scaling function with $2N$ non-zero coefficients:

$$\varphi(x) \;=\; 2 \sum_{k=0}^{2N-1} h_k \, \varphi(2x \Leftrightarrow k). \tag{2.40}$$

It is easy to see that $\operatorname{supp} \varphi = [0, 2N \Leftrightarrow 1]$, and, as a consequence,

$$B_{j,k} \;=\; \operatorname{supp} \varphi_{j,k}(x) \;=\; [2^{-j}k, 2^{-j}(k + 2N \Leftrightarrow 1)]. \tag{2.41}$$

This implies that for sufficiently small scales 2^{-j}, $j \geqslant j_0$, a function $\varphi_{j,k}$ can only intersect at most one of the endpoints 0 or 1. Let us restate this in a different way. Define the set of indices

$$S_j \;=\; \{k : B_{j,k} \cap (0,1) \neq \emptyset\}.$$

We define three subsets of this set containing the indices of the basis functions at the left boundary, in the interior, and at the right boundary:

$$\begin{aligned} S_j^{(1)} &= \{k : 0 \in B_{j,k}^{\circ}\}, \\ S_j^{(2)} &= \{k : B_{j,k}^{\circ} \subset (0,1)\}, \\ S_j^{(3)} &= \{k : 1 \in B_{j,k}^{\circ}\}. \end{aligned}$$

Here E° denotes the interior of the set E. For sufficiently large j the sets $S_j^{(1)}$ and $S_j^{(3)}$ are disjoint and

$$S_j \;=\; S_j^{(1)} \cup S_j^{(2)} \cup S_j^{(3)}.$$

It is easy to write down what these sets are more explicitly:

$$\begin{aligned} S_j^{(1)} &= \{k : \Leftrightarrow 2N + 2 \leqslant k \leqslant \Leftrightarrow 1\}, \\ S_j^{(2)} &= \{k : 0 \leqslant k \leqslant 2^j \Leftrightarrow 2N + 1\}, \\ S_j^{(3)} &= \{k : 2^j \Leftrightarrow 2N + 2 \leqslant k \leqslant 2^j \Leftrightarrow 1\}. \end{aligned}$$

Note, in particular, that the sets $S_j^{(1)}$ and $S_j^{(3)}$ contain the indices of $2N \Leftrightarrow 2$ functions, independently of j. We now let $V_j^{[0,1]}$ denote the restriction of functions to V_j:

$$V_j^{[0,1]} \;=\; \{f : f(x) = g(x),\; x \in [0,1], \text{ for some function } g \in V_j\}.$$

Clearly, since the V_j form an increasing sequence of spaces,

$$V_j^{[0,1]} \;\subset\; V_{j+1}^{[0,1]},$$

and $V_j^{[0,1]}$, $j \geqslant j_0$, form a multiresolution analysis of $\mathrm{L}^2([0,1])$. It is also obvious that the functions in $\{\varphi(x \Leftrightarrow l)|_{[0,1]} : l \in S_j\}$ span $V_j^{[0,1]}$. Here $g(x)\;|_{[0,1]}$ denotes the restriction of $g(x)$ to $[0,1]$. Not quite as obvious is the fact that the functions in this collection are linearly independent, and hence form a basis for $V_j^{[0,1]}$. In order to obtain an orthonormal basis, we may argue as follows. As long as the function $\varphi_{j,k}$ lives entirely inside $[0,1]$, restricting it to $[0,1]$ has no effect. In particular, the functions $\varphi_{j,k}$, $k \in S_j^{(2)}$ are still pairwise orthogonal. A key observation now is that for $k \in S_j^{(1)}$, $l \in S_j^{(2)} \cup S_j^{(3)}$,

$$\langle\, \varphi_{j,k}, \varphi_{j,l} \,\rangle_{\,[0,1]} \;=\; \int_0^1 \varphi_{j,k}(x)\, \varphi_{j,l}(x)\, dx \;=\; \int_{-\infty}^{+\infty} \varphi_{j,k}(x)\, \varphi_{j,l}(x)\, dx \;=\; 0, \quad (2.42)$$

and similarly when $k \in S_j^{(3)}$, $l \in S_j^{(2)} \cup S_j^{(1)}$. We see that the three collections $\{\varphi(x \Leftrightarrow l)|_{[0,1]} : l \in S_j^{(1)}\}$, $\{\varphi(x \Leftrightarrow l)|_{[0,1]} : l \in S_j^{(2)}\}$, and $\{\varphi(x \Leftrightarrow l)|_{[0,1]} : l \in S_j^{(3)}\}$ are mutually orthogonal. So, since the functions in $\{\varphi(x \Leftrightarrow l)|_{[0,1]} : l \in S_j^{(2)}\}$ already form an orthonormal set, it only remains to separately orthogonalize the functions in $\{\varphi(x \Leftrightarrow l)|_{[0,1]} : l \in S_j^{(1)}\}$ and in $\{\varphi(x \Leftrightarrow l)|_{[0,1]} : l \in S_j^{(3)}\}$. This is easily accomplished with a Gram-Schmidt procedure.

Now, if we let $W_j^{[0,1]}$ denote the restriction of functions in W_j to $[0,1]$, then we have that

$$V_{j+1}^{[0,1]} \;=\; V_j^{[0,1]} + W_j^{[0,1]}. \quad (2.43)$$

So, the basis elements in $V_j^{[0,1]}$ together with the restriction of the wavelets $\psi_{j,k}$ to $[0,1]$ span $V_{j+1}^{[0,1]}$. However, there are $2^j + 2N \Leftrightarrow 2$ wavelets that intersect $[0,1]$, and, since $\dim V_{j+1}^{[0,1]} \Leftrightarrow \dim V_j^{[0,1]} = 2^j$ we have too many functions. The restrictions of the wavelets in W_j that live entirely inside $[0,1]$ are still mutually orthogonal and, by an observation similar to (2.42), they are also orthogonal to $V_j^{[0,1]}$. There are $2N \Leftrightarrow 2$ wavelets whose support intersects an endpoint. However, we only need $N \Leftrightarrow 1$ basis functions at each endpoint. One can now use (2.31) to write out the dependencies, and construct $N \Leftrightarrow 1$ basis functions at each endpoint. After that, we just apply a Gram-Schmidt procedure again, and we have an orthonormal basis for $W_j^{[0,1]}$.

This elegant construction of Yves Meyer has a couple of disadvantages. Among the functions $\varphi_{j,k}$ that intersect $[0,1]$ there are some that are almost zero there. Hence, the set $\{\varphi_{j,k}\}_{k\in S_j}$ is almost linearly dependent, and, as a consequence, the condition number of the matrix, corresponding to the change of basis from $\{\varphi_{j,k}\}_{k\in S_j}$ to the orthonormal one, becomes quite large. Furthermore, we have $\dim V_j^{[0,1]} \neq \dim W_j^{[0,1]}$, which means that there is an inherent imbalance between the spaces $V_j^{[0,1]}$ and $W_j^{[0,1]}$, which is not present in the case of the whole real line.

2.9.3 Dyadic boundary wavelets

As we noted earlier, see (2.34), all polynomials of degree $\leqslant N \Leftrightarrow 1$ can be written as linear combinations of the $\varphi_{j,l}$ for $l \in \mathbf{Z}$. Hence, the restriction of such polynomials to $[0,1]$ are in $V_j^{[0,1]}$. Since this fact is directly linked to many of the approximation properties of wavelets, any construction of a multiresolution analysis on $[0,1]$ should preserve this. The construction in [46] uses this as a starting point and is slightly different from the one by Yves Meyer. Let us briefly describe this construction as well. Again we start with an orthogonal Daubechies scaling function φ with $2N$ non-zero coefficients, and assume that we have picked the scale fine enough so that the endpoints are independent as before. By (2.34) and, since the $\{\varphi_{j,k}\}$ is an orthonormal basis for V_j, each monomial x^α, $\alpha \leqslant N \Leftrightarrow 1$, has the representation $x^\alpha = \sum_k \langle\, x^\alpha, \varphi_{j,k} \,\rangle\, \varphi_{j,k}(x)$. The restriction to $[0,1]$ can then be written as

$$x^\alpha|_{[0,1]} \;=\; \left(\sum_{k=-2N+2}^{0} + \sum_{k=1}^{2^j-2N} + \sum_{k=2^j-2N+1}^{2^j-1} \right) \langle\, x^\alpha, \varphi_{j,k} \,\rangle\, \varphi_{j,k}(x)|_{[0,1]}.$$

If we let

$$x^\alpha_{j,L} \;=\; 2^{j(\alpha+1/2)} \sum_{k=-2N+2}^{0} \langle\, x^\alpha, \varphi_{j,k} \,\rangle\, \varphi_{j,k}(x)|_{[0,1]}$$

and, similarly,

$$x^\alpha_{j,R} \;=\; 2^{j(\alpha+1/2)} \sum_{k=2^j-2N+1}^{2^j-1} \langle\, x^\alpha, \varphi_{j,k} \,\rangle\, \varphi_{j,k}(x)|_{[0,1]},$$

then

$$2^{j/2}(2^j x)^\alpha|_{[0,1]} \;=\; x^\alpha_{j,L} + 2^{j(\alpha+1/2)} \sum_{k=1}^{2^j-2N} \langle\, x^\alpha, \varphi_{j,k} \,\rangle\, \varphi_{j,k}(x)|_{[0,1]} + x^\alpha_{j,R}.$$

The spaces $\bar{V}_j$, $j \geqslant j_0$, that form a multiresolution analysis of $\mathrm{L}^2([0,1])$, we take to be the linear span of the functions $\{x^\alpha_{j,L}\}_{\alpha\leqslant N-1}$, $\{x^\alpha_{j,R}\}_{\alpha\leqslant N-1}$, and $\{\varphi_{j,k}|_{[0,1]}\}_{k=1}^{2^j-2N}$:

$$\bar{V}_j \;=\; \text{span}\,\{x^\alpha_{j,L}\}_{\alpha\leqslant N-1} \;\cup\; \text{span}\,\{\varphi_{j,k}\}_{k=1}^{2^j-2N} \;\cup\; \text{span}\,\{x^\alpha_{j,R}\}_{\alpha\leqslant N-1}.$$

Finding an orthonormal basis for $\bar{V}_j$ now is easy; the collections $\{x^\alpha_{j,L}\}_{\alpha\leqslant N-1}$, $\{\varphi_{j,k}\}_{k=1}^{2^j-2N}$, and $\{x^\alpha_{j,R}\}_{\alpha\leqslant N-1}$ are mutually orthogonal, and all of the functions in these are linearly independent. We thus only have to orthogonalize the functions $x^\alpha_{j,L}$ and $x^\alpha_{j,R}$ to get our orthonormal basis. Note that, by construction, $\dim \bar{V}_j = 2^j$ and all polynomials of degree $\leqslant N \Leftrightarrow 1$ are in $\bar{V}_j$. It is also easy to see that

$$\bar{V}_j \;\subset\; \bar{V}_{j+1}.$$

To get to the corresponding wavelets, we let $\bar{W}_j$ be the orthogonal complement of $\bar{V}_j$ in $\bar{V}_{j+1}$. The wavelets $\psi_{j,k}$ with $1 \leqslant k \leqslant 2^j \Leftrightarrow 2N$ are all in $\bar{V}_{j+1}$ and live entirely inside $[0,1]$. The remaining $2N$ functions required for an orthonormal basis of $\bar{W}_j$, can be found, for example by using (2.31) again.

This last construction carries over to more general situations. For example, we can also use biorthogonal wavelets and much more general closed sets than $[0,1]$ [12, 47, 108].

There are also other constructions of wavelets on $[0,1]$. For historical perspective, it is interesting to notice that Franklin's original construction [86] was given for $[0,1]$. Another interesting one, in the case of semiorthogonal spline wavelets, has been given by Charles Chui and Ewald Quak [33]; we refer to the original paper for details.

2.10 Wavelet packets

A simple, but powerful extension of wavelets and multiresolution analysis are wavelet packets [53]. In this section, it will be useful to switch to the following notation:

$$m_e(\omega) \;=\; h^e(\omega)\, g^{1-e}(\omega) \qquad \text{for} \qquad e = 0, 1.$$

The fundamental observation is the following fact, called the *splitting trick* [36, 44, 140]:
Suppose that the set of functions $\{f(x \Leftrightarrow k) \mid k \in \mathbf{Z}\}$ *is a Riesz basis for its closed linear span* S*. Then the functions*

$$f^0_k \;=\; \frac{1}{\sqrt{2}} f^0(x/2 \Leftrightarrow k) \qquad \textit{and} \qquad f^1_k \;=\; \frac{1}{\sqrt{2}} f^1(x/2 \Leftrightarrow k) \qquad \textit{for} \qquad k \in \mathbf{Z},$$

also constitute a Riesz basis for S*, where*

$$\widehat{f^e}(\omega) \;=\; m_e(\omega/2)\, \widehat{f}(\omega/2).$$

We see that the classical multiresolution analysis is obtained by splitting V_j with this trick into V_{j-1} and W_{j-1} and then doing the same for V_{j-1} recursively.

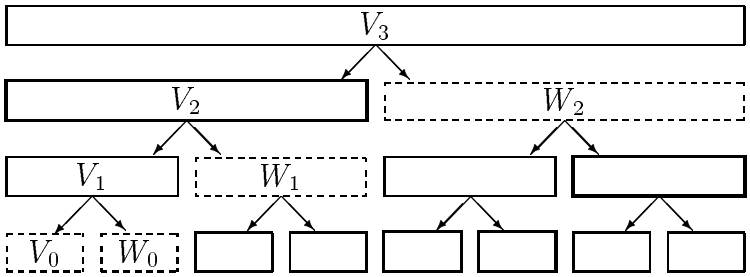

Figure 2.4: Wavelet packets scheme.

The wavelet packets are the basis functions that we obtain if we also use the splitting trick on the W_j spaces. So starting from a space V_j, we obtain, after applying the splitting trick L times, the basis functions

$$\psi^L_{e_1,\ldots,e_L;j,k}(x) \;=\; 2^{(j-L)/2}\psi^L_{e_1,\ldots,e_L}(2^{j-L}x \Leftrightarrow k),$$

with

$$\widehat{\psi}^L_{e_1,\ldots,e_L}(\omega) \;=\; \prod_{i=1}^{L} m_{e_i}(2^{-i}\,\omega)\,\widehat{\varphi}(2^{-L}\omega).$$

So, after L splittings, we have 2^L basis functions and their translates over integer multiples of 2^{L-j} as a basis of V_j. The connection between the wavelet packets and the wavelet and scaling functions is

$$\varphi \;=\; \psi^L_{0,\ldots,0} \quad \text{and} \quad \psi \;=\; \psi^L_{1,0,\ldots,0}.$$

However, we do not necessarily have to split each subspace at every stage. In Figure 2.4, we give a schematical representation of a space and its subspaces after using the splitting on three levels. The top rectangle represents the space V_3 and each other rectangle corresponds to a certain subspace of V_3 generated by wavelet packets. The slanted lines between the rectangles indicate the splitting, the left referring to the filter m_0 and the right to m_1. The dashed rectangles then correspond to the wavelet multiresolution analysis $V_3 = V_0 \oplus W_0 \oplus W_1 \oplus W_2$. The bold rectangles correspond to a possible wavelet packet splitting and a basis with functions

$$\left\{\psi^1_0(4x \Leftrightarrow k),\; \psi^2_{1,1}(2x \Leftrightarrow k),\; \psi^3_{0,0,1}(x \Leftrightarrow k),\; \psi^3_{1,0,1}(x \Leftrightarrow k) \mid k \in \mathbf{Z}\right\}.$$

For the dual functions, a similar procedure has to be followed.

In the Fourier domain, the splitting trick corresponds to dividing the frequency interval essentially represented by the original space into a lower and an upper part. So the wavelet packets allow more flexibility in adapting the basis to the frequency contents of a signal.

It is easy to develop a fast wavelet packet transform. It just involves applying the same low and band pass filters also to the coefficient of functions of W_j again in an iterative manner. This means that, starting from M samples, we construct a full binary tree with $(M \log_2 M)$ entries. The power of this construction lies in the fact that we have much more freedom in deciding which basis functions we will use to represent the given function. We can choose to use the set of M coefficients of the tree to represent the function that is optimal with respect to a certain criterion. This procedure is called *best basis selection*, and one can design fast algorithms that make use of the tree structure. The particular criterion is determined by the application, and which basis functions that will end up in the basis depends on the data.

For applications in image processing, entropy-based criteria were proposed in [54]. The best basis selection in that case has a numerical complexity of $\mathcal{O}(M)$. Applications in signal processing can be found in [52, 195].

This wavelet packets construction can also be combined with wavelets on an interval and wavelets in higher dimensions [73].

2.11 Multidimensional wavelets

Up till now we have focused on functions of one variable and the one-dimensional situation. However, there are also wavelets in higher dimensions. A simple way to obtain these is to use tensor products. To fix ideas, let us consider the case of the plane. Let

$$\Phi(x, y) \;=\; \varphi(x)\,\varphi(y) \;=\; \varphi \otimes \varphi(x, y),$$

and define

$$V_0 \;=\; \{f : f(x, y) = \sum_{k_1,k_2} \lambda_{k_1,k_2}\, \Phi(x \Leftrightarrow k_1, y \Leftrightarrow k_2), \lambda \in l^2(\mathbf{Z}^2)\}.$$

Of course, if $\{\varphi(x \Leftrightarrow l) \mid l \in \mathbf{Z}\}$ is an orthonormal set, then $\{\Phi(x \Leftrightarrow k_1, y \Leftrightarrow k_2)\}$ form an orthonormal basis for V_0. By dyadic scaling we obtain a multiresolution analysis of $\mathrm{L}^2(\mathbf{R}^2)$. The complement W_0 of V_0 in V_1 is similarly generated by the translates of the three functions

$$\Psi^{(1)} \;=\; \varphi \otimes \psi, \quad \Psi^{(2)} \;=\; \psi \otimes \varphi, \quad \text{and} \quad \Psi^{(3)} \;=\; \psi \otimes \psi. \tag{2.44}$$

Another, perhaps even more straightforward, wavelet decomposition in higher dimensions exists. By carrying out a one-dimensional wavelet decomposition for each variable separately, we obtain

$$f(x, y) \;=\; \sum_{i,l} \sum_{j,k} \langle\, f, \psi_{i,l} \otimes \psi_{j,k} \,\rangle\; \psi_{i,l} \otimes \psi_{j,k}(x, y). \tag{2.45}$$

Note that the functions $\psi_{i,l} \otimes \psi_{j,k}$ involve two scales, 2^{-i} and 2^{-j}, and each of these functions are (essentially) supported on a rectangle. The decomposition (2.45) is

therefore called the *rectangular wavelet decomposition* of f while the functions in (2.44) are the basis functions of the *square wavelet decomposition*. For both decompositions, the corresponding fast wavelet transform consists of applying the one-dimensional fast wavelet transform to the rows and columns of a matrix.

These simple constructions are insufficient in many cases. What we need sometimes are wavelets intrinsically constructed for higher dimensions. One of the interesting problems here is how to split a space into complementary subspaces. In the univariate case we split into two spaces, each with essentially the same "size." If we use the square tensor product basis in d dimensions, we split into 2^d subspaces, $2^d \Leftrightarrow 1$ of which are spanned by wavelets. There are several constructions of nonseparable wavelets that use this kind of splitting. One of the problems here is, given the scaling function, is there an easy way, cf. (2.20), to find the wavelets? This was studied in [72, 151, 164]. Another idea is to still try to split into just two subspaces. This involves the use of different lattices [130]. In the bivariate case, Ingrid Daubechies and Albert Cohen constructed smooth, compactly supported, biorthogonal wavelets, using ideas from the univariate construction [43].

Also many other constructions, such as hexagonal lattices [48] and Clifford valued wavelets [13], exist.

2.12 Applications

2.12.1 Data compression

A common applications of wavelet theory is data compression. There are two basic kinds of compression schemes: lossless and lossy. In the case of lossless compression, one is interested in reconstructing the data exactly, without any loss of information. We consider here lossy compression. This means we are ready to accept an error, as long as the quality after compression is acceptable. With lossy compression schemes we potentially can achieve much higher compression ratios than with lossless compression.

To be specific, let us assume that we are given a digitized image. The compression ratio is defined as the number of bits the initial image takes to store on the computer divided by the number of bits required to store the compressed image. The interest in compression has grown as the amount of information we pass around has increased. This is easy to understand when we consider the fact that to store a moderately large image, say a 512×512 pixels, 24 bit color image, takes about 0.75 MBytes. This is only for still images; in the case of digital video, the situation becomes even worse. Then, we need this kind of storage for each frame, and we have something like 30 frames per second.

Let us define, somewhat mathematically, what we mean by an image. Let us for simplicity discuss an $L \times L$ grayscale image with 256 grayscales (i.e. 8 bit).

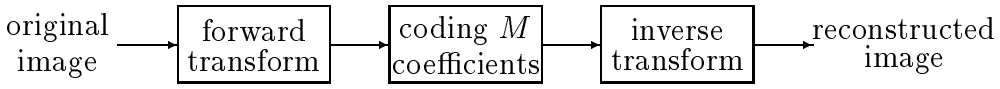

Figure 2.5: Image transform coding.

This can be considered to be a piecewise constant function f defined on a square

$$f(x,y) \;=\; p_{ij} \in \mathbf{N}, \quad \text{for} \quad i \leqslant x < i+1, \;\; j \leqslant y < j+1, \quad \text{and} \quad 0 \leqslant i,j < L,$$

where $0 \leqslant p_{ij} \leqslant 255$. Now, one of the standard procedures for lossy compression is through transform coding, see Figure 2.5. The most common transform used in this context is the "Discrete Cosine Transform" (DCT), a variant of the Fourier transform. However, we are interested in the case when the transform is the fast wavelet transform.

There are several ways to use the wavelet transform for compression purposes [132, 133]. One is to consider compression to be an approximation problem [77, 78]. More specifically, let us fix an orthogonal wavelet ψ. Given an integer $M \geqslant 1$, we try to find the "best" approximation of f by using a representation

$$f_M(x) \;=\; \sum_{kl} b_{jk}\, \psi_{jk}(x) \quad \text{with } M \text{ non-zero coefficients } b_{jk}. \tag{2.46}$$

The basic reason why this potentially might be useful is that each wavelet picks up information about the image f essentially at a given location and at a given scale. Where the image has more interesting features, we can spend more coefficients, and where the image is nice and smooth we can use fewer, and still get good quality of approximation. In other words, the wavelet transform allows us to focus on the relevant parts of f. Now, to give this mathematical meaning, we need to agree on an error measure. Ideally, for image compression we should use a norm that corresponds as closely as possible to the human eye. However, let us make it simple and discuss the case of L_2.

So we are interested in finding an optimal approximation minimizing the error $\| f \Leftrightarrow f_M \|_2$. Because of the orthogonality of the wavelets this equals

$$\left(\sum_{jk} | \, \langle \, f, \psi_{jk} \, \rangle \, \Leftrightarrow b_{jk}|^2 \right)^{1/2}. \tag{2.47}$$

A moment's thought reveals that the best way to pick M non-zero coefficients b_{jk}, making the error as small as possible, is by simply picking the M coefficients with largest absolute value, and setting $b_{j,k} = \langle \, f, \psi_{jk} \, \rangle$ for these numbers. This then yields the optimal approximation f_M^{opt}.

Another fundamental question is which images can be approximated well by using the procedure just sketched. Let us take this to mean that the error satisfies

$$\| f \Leftrightarrow f_M^{opt} \|_2 \;=\; \mathcal{O}(M^{-\beta}), \tag{2.48}$$

for some $\beta > 0$. The larger β, the faster the error decays as M increases and the fewer coefficients are generally needed to obtain an approximation within a given error. The exponent β can be found easily. It can be shown that

$$\left(\sum_{M \geqslant 1} (M^{\beta} \| f \Leftrightarrow f_M^{opt} \|_2)^p \frac{1}{M} \right)^{1/p} \approx (\sum_{jk} | \langle f, \psi_{jk} \rangle |^p)^{1/p} \tag{2.49}$$

with $1/p = 1/2 + \beta$. The maximal β for which (2.48) is valid can be estimated by finding the smallest p for which the right-hand side of (2.49) is finite. The expression on the right is one of many equivalent norms on the Besov space $\dot{B}_p^{2\beta,p}$ (Besov spaces are smoothness spaces generalizing the Hölder continuous functions). The β in the left-hand side of (2.49) is actually not exactly the same as in (2.48). However, for practical purposes, the difference is of no consequence.

In Chapter 7, we construct algorithms for image compression. Here, we use a construction of wavelets on the Fourier transform side. As a result, the transform can be seen as a variant of the DCT.

2.12.2 Operator analysis

As mentioned earlier, interest in wavelets historically grew from the fact that they are effective tools for studying problems in partial differential equations and operator theory. More specifically, they are useful for understanding properties of so-called Calderón-Zygmund operators.

Let us first make a general observation about the representation of a linear operator T and wavelets. Suppose that f has the representation

$$f(x) = \sum_{jk} \langle f, \psi_{jk} \rangle \psi_{jk}(x).$$

Then,

$$T f(x) = \sum_{jk} \langle f, \psi_{jk} \rangle T \psi_{jk}(x),$$

and, using the wavelet representation of the function $T \psi_{jk}(x)$, this equals

$$\sum_{jk} \langle f, \psi_{jk} \rangle \sum_{il} \langle T \psi_{jk}, \psi_{il} \rangle \psi_{il}(x) = \sum_{il} \left(\sum_{jk} \langle T \psi_{jk}, \psi_{il} \rangle \langle f, \psi_{jk} \rangle \right) \psi_{il}(x).$$

In other words, the action of the operator T on the function f is directly translated into the action of the infinite matrix $A_T = \{ \langle T \psi_{jk}, \psi_{il} \rangle \}_{il,jk}$ on the sequence $\{ \langle f, \psi_{jk} \rangle \}_{jk}$. This representation of T as the matrix A_T is often referred to as the "standard representation" of T [25]. Also a "nonstandard representation" exists. For virtually all linear operators a function (or, more generally, a distribution), the *reproducing kernel* K, exists so that

$$T f(x) = \int K(x, y) f(y) \, dy.$$

The nonstandard representation of T is given by the (two-dimensional) wavelet coefficients of the kernel K, using the square decomposition $\{ \langle K, \Psi^{(j)}_{k_1,k_2} \rangle \}$ (again, we have more than one wavelet function in two dimensions), while the standard representation corresponds to the rectangular decomposition.

Let us then briefly discuss the connection with Calderón-Zygmund operators. Consider a typical example. Let $\mathcal{H}$ be the *Hilbert transform*,

$$\mathcal{H} f(x) \;=\; \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{f(s)}{x \Leftrightarrow s}\, ds.$$

The basic idea now is that the wavelets ψ_{jk} are approximate eigenfunctions for this, as well as for many other related (Calderón-Zygmund) operators. We note that if ψ_{jk} were exact eigenfunctions, then we would have $\mathcal{H}\,\psi_{jk}(x) = \lambda_{jk}\,\psi_{jk}(x)$, for some number λ_{jk} and the standard representation would be a diagonal "matrix":

$$A_{\mathcal{H}} \;=\; \{ \langle \mathcal{H}\,\psi_{il}, \psi_{jk} \rangle \} \;=\; \{\lambda_{il} \langle \psi_{il}, \psi_{jk} \rangle \} \;=\; \{\lambda_{il}\,\delta_{il,jk}\}.$$

This is, unfortunately, not the case. However, it turns out that A_T is an almost diagonal operator, in the appropriate, technical sense, with the off-diagonal elements quickly becoming small. To get some idea why this is the case, note that for large $|x|$, we have, at least heuristically,

$$\mathcal{H}\,\psi_{jk}(x) \;\approx\; \frac{1}{\pi x} \int \psi_{jk}(y)\, dy.$$

A priori, the decay of the right-hand side would thus be $\mathcal{O}(1/x)$, which of course is far from the rapid decay of a wavelet ψ_{jk}. Recall, however, that ψ_{jk} has at least one vanishing moment so the decay is in fact much faster than just $\mathcal{O}(1/x)$, and the shape of $\mathcal{H}\,\psi_{jk}(x)$ resembles that of $\psi_{jk}(x)$.

So, for a large class of operators, the matrix representation, either the standard or the nonstandard, has a rather precise structure with many small elements. In this representation, we then expect to be able to compress the operator by simply omitting small elements. Note that this is essentially the same situation, as in the case of image compression, the "image" now being the kernel $K(x, y)$. Hence, if we could do basic operations, such as inversion and multiplication, with compressed matrices, rather than with the discretized versions of T, then we may significantly speed up the numerical treatment. This program of using the wavelet representations for the efficient numerical treatment of operators was initiated in [25]. We also refer to [8, 9] for related material and many more details.

In a different direction, because of the close similarities between the scaling function and finite elements, it seems natural to try wavelets where traditionally finite element methods are used, e.g. for solving boundary value problems [102]. There are interesting results showing that this might be fruitful; for example, it has been shown [25, 60, 145, 197] that for many problems the condition number of the $N \times N$ stiffness matrix remains bounded as the dimension N goes to infinity. This

is in contrast with the situation for regular finite elements, where the condition number tends to infinity.

Wavelets have also been used in the solution of evolution equations [18, 92, 121, 129]. A typical test problem here is *Burgers' equation*:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = \nu \frac{\partial^2 u}{\partial x^2}.$$

The time discretization is obtained here using standard schemes such as Crank-Nicholson or Adams-Moulton. Wavelets are used in the space discretization. Adaptivity can be used both in time and space [18].

We will discuss the connection between wavelets and differential equations further in Chapters 5 and 6.

Chapter 3

Quadrature Formulae for the Calculation of the Wavelet Decomposition

" It has been long my personal understanding that the separation of practical and theoretical work is artificial and injurious. Much of the practical work done in computing, both in software and hardware design, is unsound and clumsy because the people who do it do not have any clear understanding of the fundamental principles underlying their work. Most of the abstract mathematical and theoretical work is sterile because it has no point of contact with real computing."
—Christopher Strachy, *seen on a Computer Science professor's door.*

3.1 Introduction

As we saw in the previous chapter, almost every application using wavelets at some point involves the calculation of the multiresolution coefficients $\lambda_{j,l}$ and $\gamma_{j,l}$ of a function f. Remember that these coefficients are defined as inner products with dual functions. However, we can hardly ever calculate these integrals exactly. Therefore, we need to construct schemes to approximate them numerically. A classic method is a so-called *quadrature formula*, which approximates an integral of a function, possibly multiplied with a weight function, with a linear combination of evaluations of that function at particular points called *abscissae* [59, 148].

We discuss here the construction and use of quadrature formulae in connection with multiresolution analysis. Algorithms for the implementation of the constructions are provided. Since the construction involves the moments of the scaling function we also derive a recursion formula to calculate them. We show that the construction using monomial moments is ill-conditioned and build a well-conditioned construction using Chebyshev polynomials. Numerical examples of

the use of these formulae in a multiresolution analysis are given. Finally, we compare the quadrature formulae with other methods and discuss how wavelets on an interval can be handled.

3.2 General idea

Consider a multiresolution analysis where the finest level is n and assume, without loss of generality, that the coarsest level is 0. This implies that we need to calculate the coefficients $\lambda_{j,l}$ for $0 \leqslant j \leqslant n$ and $\gamma_{j,l}$ for $0 \leqslant j < n$. As described in Section 2.7, the coefficients $\lambda_{j,l}$ and $\gamma_{j,l}$ with $j < n$ can be calculated from the $\lambda_{n,l}$ using the fast wavelet transform. We therefore use the quadrature formula on the finest level to approximate the $\lambda_{n,l}$. Remember that these coefficients are defined as

$$\lambda_{n,l} \;=\; \sqrt{2^n} \int_{-\infty}^{+\infty} f(x)\, \overline{\widetilde{\varphi}(2^n x \Leftrightarrow l)}\, dx.$$

We only consider real-valued functions, so the complex conjugation is superfluous. Because of the translation and dilation properties we can focus on the case $n = l = 0$,

$$\lambda_{0,0} \;=\; \int_{-\infty}^{+\infty} f(x)\, \widetilde{\varphi}(x)\, dx.$$

Here we consider the dual scaling function as a weight function. Furthermore, for notational simplicity, we from now on omit the tilde in the notation of the dual scaling function. We keep in mind that the coefficients are given by the inner products with the dual functions and remember that, as mentioned earlier, basis functions and dual functions are always interchangeable.

The idea of a quadrature formula now is to find *weights* w_k and *abscissae* x_k so that

$$\int_{-\infty}^{+\infty} f(x)\, \varphi(x)\, dx \;\approx\; Q[f(x)] \;=\; \sum_{k=0}^{r} w_k\, f(x_k). \tag{3.1}$$

Once the weights and abscissae are known, the coefficients on the finest level can be approximated by

$$\lambda_{n,l} \;\approx\; \frac{1}{\sqrt{2^n}} \sum_{k=0}^{r} w_k\, f\left(\frac{x_k + l}{2^n}\right). \tag{3.2}$$

Evidently, the quadrature formula can also be used to calculate the coefficients $\lambda_{j,l}$ with $j < n$. Why it is better to use the quadrature formula only on the finest level and the fast wavelet transform on the coarser levels will become apparent later.

We now need to address two important issues: how to choose the abscissae and how to find the weights.

The weights are determined by the fact that the quadrature formula is, in some sense, a good approximation for the integral. We will be more precise later on and

just mention now that this typically leads to a linear system in the unknowns w_k. For the abscissae one classically has two options. On one hand one can fix the abscissae and solve for the weights, while on the other hand one can consider the abscissae as unknowns too, cf. the idea of Gaussian quadrature formulae [170]. In the first case the quadrature formula is easier to construct, while in the second it is more accurate. In connection with multiresolution analysis, we settle for a compromise.

The abscissae have to be chosen equidistant for two reasons. First of all, in many applications such as signal processing, image processing and time series, the function f is only known through its evaluations at equidistant points. The second reason follows from the translation properties of the multiresolution analysis. One typically needs to calculate the coefficients $\lambda_{n,l}$ for a wide range of the parameter l. If the abscissae are not equidistant, each coefficient needs r function evaluations. Note that function evaluations can be computationally very expensive in comparison with the algebraic operations of the quadrature formula. In case the abscissae are equidistant, quadrature formulae for neighboring coefficients can share function evaluations. More precisely, if the distance between two abscissae is 2^{-n}, each extra coefficient only needs one extra function evaluation, cf. (3.2).

The fact that they are equidistant does not pin down their location exactly. We still can allow a shift τ and a different spacing. Therefore, we let the abscissae be of the form $k2^{-s} + \tau$ with $k \in \mathbf{Z}$. The shift τ is an extra unknown and is determined together with the weights by the fact that the quadrature formula needs to approximate the integral.

Let us consider the problem of finding the unknowns. A popular technique to solve problems involving functions in numerical analysis is to design an approximate solution scheme that is exact for polynomials. When numerically approximating integrals, this leads to the following definition.

Definition 3.1 *The* degree of accuracy *of a quadrature formula is q if it yields the exact result for every polynomial of degree less than or equal to q.*

We can write this as

$$Q[x^i] \;=\; \mathcal{M}_i \quad \text{for} \quad 0 \leqslant i \leqslant q. \tag{3.3}$$

The importance of the degree of accuracy can be understood as follows: if the function f is smooth, it locally resembles a polynomial and the quadrature formula gives an accurate result. We give a more precise statement in Section 3.5. Note that we do not impose any regularity conditions on φ. Equations (3.3) now lead to an algebraic system in the unknowns w_k and τ.

The smaller the number of abscissae r, the more efficient the quadrature formula since the number of function evaluations and algebraic operations for one coefficient is proportional to r. From simply counting the number of unknowns and equations we can hope for $q = r$. However, this is not guaranteed as the algebraic system is nonlinear.

3.3 Special cases

3.3.1 Trapezoidal rule

A simple quadrature formula is the *trapezoidal rule*, where

$$Q[f(x)] \;=\; \sum_k \varphi(k)\, f(k). \tag{3.4}$$

In general, the use of this rule is limited because it only has a degree of accuracy equal to one. In connection with multiresolution analysis, however, the following lemma holds:

Lemma 3.2 *If the scaling function satisfies the Strang-Fix condition,*

$$\widehat{\varphi}^{(p)}(k2\pi) \;=\; 0 \quad \textit{for} \quad 0 \leqslant p < N \quad \textit{and} \quad k \neq 0,$$

the degree of accuracy of the trapezoidal rule (3.4) is equal to $N \Leftrightarrow 1$.

Proof : Follows immediately from (2.33) for $x = 0$. □

In other words, if the scaling function and its integer translates can reproduce polynomials up to degree $N \Leftrightarrow 1$, the trapezoidal rule has a degree of accuracy equal to N. This simple result is remarkable. It was already known that the trapezoidal rule is more accurate than expected in special cases such as periodic integrands. This lemma adds another case, namely when the weight function satisfies the Strang-Fix condition up to some order. Note that no regularity of the weight function φ is required. The function only needs to be continuous at the integers. Typical functions that satisfy the Strang-Fix condition are the functions of finite element methods such as B-splines. In case φ satisfies a refinement relation, the values $\varphi(k)$ can easily be calculated as solutions of an eigenvalue problem as described in [165].

A disadvantage is that the trapezoidal rule is not very efficient. In case φ is not compactly supported, the sum in (3.4) has to be broken off, which usually leads to a large number of abscissae. But also when φ is compactly supported, the trapezoidal rule is not really efficient: the Daubechies orthogonal scaling functions have a support length of $2N \Leftrightarrow 1$, and hence $r = 2N \Leftrightarrow 2 = 2q$. Remember that we are hoping for $r = q$. Only in the case of cardinal B-splines is the trapezoidal rule useful. The B-spline of order m has support width of m and can reproduce polynomials of degree less than or equal to $m \Leftrightarrow 1$. Consequently, $q = r = m \Leftrightarrow 1$.

3.3.2 One-point formulae

Before we consider the general construction, let us take a look at the case where the number of abscissae is one. Since the integral of the scaling function is one,

we can write a one-point formula as $Q[f(x)] = f(x_1)$. This means that we take the function value as an approximation for the coefficient. Evidently, if $x_1 = \mathcal{M}_1$, the degree of accuracy is equal to one. In the case of orthogonal wavelets, the following theorem holds:

Theorem 3.3 *If $\varphi(x)$ is an orthogonal scaling function with $N > 1$, then*

$$\mathcal{M}_2 \;=\; \mathcal{M}_1^2.$$

Proof : Define

$$\kappa_m \;=\; \langle\, x, \varphi(x)\, \varphi(x \Leftrightarrow m) \,\rangle\,.$$

Because of the orthogonality it holds that

$$\kappa_{-m} \;=\; \langle\, x \Leftrightarrow m, \varphi(x \Leftrightarrow m)\, \varphi(x) \,\rangle \;=\; \kappa_m.$$

Consequently,

$$0 \;=\; \sum_m m\, \kappa_m \;=\; \langle\, x, \varphi(x) \sum_m m\, \varphi(x \Leftrightarrow m) \,\rangle\,.$$

Since $N > 1$, we have that

$$\sum_m m\, \varphi(x \Leftrightarrow m) \;=\; x \Leftrightarrow \mathcal{M}_1.$$

Combining the last two equations yields $\mathcal{M}_2 \Leftrightarrow \mathcal{M}_1^2 \;=\; 0.$ □

This implies that the degree of accuracy of a one-point quadrature formula is two, which is one more than expected. *Note*: This theorem was proven independently for the case of Daubechies' scaling functions in [93].

3.3.3 Coiflets

The idea of a one-point quadrature is attractive because of its simplicity. Its degree of accuracy is, however, limited. Therefore, Ingrid Daubechies constructed orthogonal scaling functions with compact support for which the one-point quadrature has a higher degree of accuracy [66]. These scaling functions again can reproduce polynomials up to degree $N \Leftrightarrow 1$, but moreover have $N \Leftrightarrow 1$ vanishing moments,

$$\mathcal{M}_p = 0 \quad \text{for} \quad 1 \leqslant p < N. \tag{3.5}$$

The corresponding wavelets were called *coiflets* after Ronald Coifman, who asked for their construction, because he and his collaborators wanted to use them in numerical analysis applications such as the solution of integral equations [25].

Figures 3.1 and 3.2 show them in case $N = 2$ and $N = 4$. We see from (2.33) that they also satisfy

$$\sum_k k^p \varphi(k) \;=\; \delta_p \quad \text{for} \quad 0 \leqslant p < N. \tag{3.6}$$

In this case the one-point quadrature formula with $x_1 = 0$ immediately has a degree of accuracy of $N \Leftrightarrow 1$.

The price to pay for these extra moment conditions is a larger support of the basis functions. The support width of the coiflets and the associated wavelets is $3N \Leftrightarrow 1$, as opposed to $2N \Leftrightarrow 1$ for the original Daubechies orthogonal scaling functions and wavelets. This implies that all fast wavelet transform filters $\{h_k\}$ and $\{g_k\}$ will be approximately 50% longer. This can be a very high price, especially in real time applications or situations where one needs to calculate the fast wavelet transform many times. In these cases one might prefer to use a more complicated quadrature formula on the finest level, which has a sufficiently high degree of accuracy but does not imply the use of longer filters.

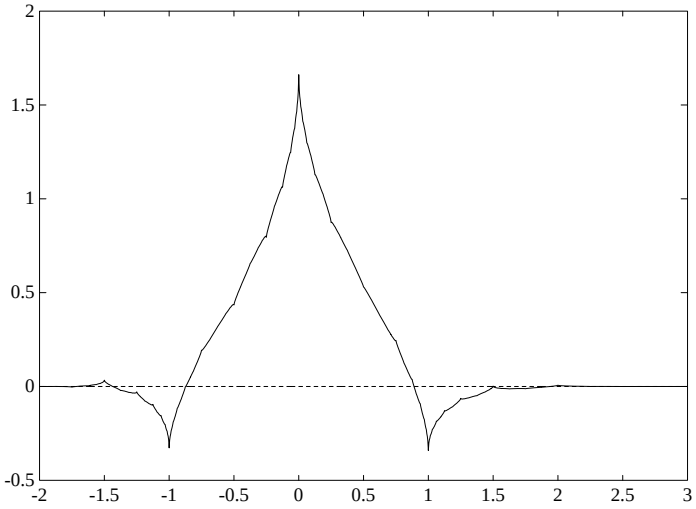

Figure 3.1: Coiflet for $N = 2$.

3.3.4 Practical aspects

In applications such as signal and image processing, usually the data is given in the form of discrete samples $\{d_l\}$. One then has several possibilities regarding how to relate these samples to a continuous function. What is often done is to take the sample values as coefficients of the basis functions. It was proposed in [135] to construct a function $v_n \in V_n$ as

$$v_n(x) \;=\; \sqrt{h} \sum_l d_l \, \varphi_{n,l}(x) \quad \text{with} \quad h = 2^{-n}$$

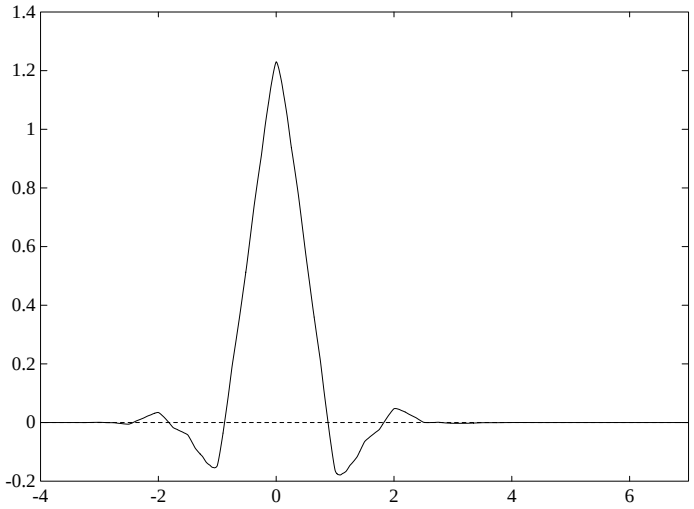

Figure 3.2: Coiflet for $N = 4$.

to start the multiresolution analysis. We can see that the continuous function v_n will in a way "follow" the discrete samples d_l. The quadrature formula can help us to find a relationship between the function v_n and the discrete samples d_l. Indeed, using the notation with tilde again,

$$\sqrt{h}\, d_l \;=\; \langle\, v_n, \widetilde{\varphi}_{n,l}\,\rangle$$

and

$$\langle\, v_n, \widetilde{\varphi}_{n,l}\,\rangle \;=\; \sqrt{h}\,[v_n(h\,(\mathcal{M}_1 + l)) + \mathcal{O}(h^t)] \quad \text{with} \quad h = 2^{-n},$$

so

$$d_l \;=\; v_n(h\,(\mathcal{M}_1 + l)) + \mathcal{O}(h^t).$$

This means that v_n satisfies a quasi-interpolating property. Quasi-interpolation will be studied in greater detail in Section 3.8.3. Here $t = 2$ in general, $t = 3$ for orthogonal wavelets and $t = N$ for coiflets.

Consider the case where the samples d_l can be seen as function evaluations of a smooth function f, $d_l = f(hl)$. Then the following theorem is important:

Theorem 3.4 *If $f \in \mathcal{C}^N$ with $f^{(i)}$ bounded for $i \leqslant N$, then ($h = 2^{-n}$)*

$$\sum_l f(hl)\,\varphi(2^n x \Leftrightarrow l) \;=\; \sum_l \varphi(l)\, f(x \Leftrightarrow hl) + \mathcal{O}(h^N).$$

Proof :

$$\sum_l f(hl)\,\varphi(2^j x \Leftrightarrow l) \;=\; \sum_l \sum_{i=0}^{N-1} \frac{(hl \Leftrightarrow x)^i}{i!} f^{(i)}(x)\,\varphi(2^j x \Leftrightarrow l) + \mathcal{O}(h^N)$$

$$
\begin{aligned}
&= \sum_{i=0}^{N-1} f^{(i)}(x) \frac{(\Leftrightarrow h)^i}{i!} \sum_l (2^j x \Leftrightarrow l)^i \, \varphi(2^j x \Leftrightarrow l) + \mathcal{O}(h^N) \\
&= \sum_{i=0}^{N-1} f^{(i)}(x) \frac{(\Leftrightarrow h)^i}{i!} \sum_l l^i \varphi(l) + \mathcal{O}(h^N) \\
&= \sum_l \varphi(l) \sum_{i=0}^{N-1} f^{(i)}(x) \frac{(\Leftrightarrow hl)^i}{i!} + \mathcal{O}(h^N) \\
&= \sum_l \varphi(l) \, f(x \Leftrightarrow hl) + \mathcal{O}(h^N).
\end{aligned}
$$

□

This theorem states that taking function evaluations as coefficients results in approximating a different function $\widetilde{f}_n(x) = \sum_l \varphi(l) \, f(x \Leftrightarrow hl)$ with an error of $\mathcal{O}(h^N)$. This function can be seen as a "blurred" version of $f(x)$, as $\{\varphi(l)\}$ is a low pass filter. Now, $\widetilde{f}_n(x)$ will converge to $f(x)$ for $n \to \infty$ since $\sum_k \varphi(k) = 1$. However, in general this convergence is only $\mathcal{O}(h)$. In the case of the coiflets we see from (3.6) that $\widetilde{f}_n(x) = f(x) + \mathcal{O}(h^N)$.

3.4 General case

3.4.1 Construction scheme

Since the degree of accuracy of a one-point formula in general is limited, we construct multiple-point quadrature formulae. In this section we assume that φ has compact support $[0, L]$ and satisfies a refinement equation (2.4) with $L+1$ non-zero coefficients h_k. Although the construction is general, we focus on scaling functions with compact support, since in this case we have the extra limitation that the abscissae should fall inside the integration interval. We construct an r point quadrature formula with $x_k = a_k \Leftrightarrow \tau$, $a_k = (k \Leftrightarrow 1)2^s$ and $(r \Leftrightarrow 1)2^s \Leftrightarrow L \leqslant \tau \leqslant 0$. The range of the shift τ is determined by the requirement that no abscissae should fall outside the integration interval. In order to have a non-zero range for the shift τ, the parameters r and s should be chosen so that $(r \Leftrightarrow 1)2^s < L$. This technique to construct quadrature formulae is also used in [25], but there the shift τ is given a fixed value.

Since there are $r+1$ unknowns $\{\tau, w_1, \ldots, w_r\}$, one can try to achieve a degree of accuracy $q = r$. This results in the following system, which is nonlinear in the unknown τ,

$$
\sum_{k=1}^{r} w_k \left[a_k \Leftrightarrow \tau\right]^i = \mathcal{M}_i \qquad\qquad 0 \leqslant i \leqslant r. \tag{3.7}
$$

The value of the shift τ can be determined using the product polynomial $\Pi(x)$.

This polynomial is defined as

$$\Pi(x) \;=\; \prod_{k=1}^{r}(x \Leftrightarrow x_k) \;=\; \prod_{k=1}^{r}(x+\tau \Leftrightarrow a_k) \;=\; \sum_{i=0}^{r} p_i(\tau)\, x^i,$$

where $p_i(\tau)$ is a polynomial of degree $r \Leftrightarrow i$. Since the degree of accuracy is r, the quadrature formula gives the exact result for the product polynomial $\Pi(x)$ so,

$$\begin{aligned} 0 \;&=\; Q_r[\Pi(x)] \;=\; \int_0^L \varphi(x)\,\Pi(x)\,dx \;=\; \int_0^L \varphi(x) \sum_{i=0}^{r} p_i(\tau)\, x^i\, dx \\ &=\; \sum_{i=0}^{r} p_i(\tau)\, \mathcal{M}_i \;=\; ,\,(\tau). \end{aligned}$$

Here $,\,(\tau)$ is a polynomial of degree r in τ. For the quadrature formula to exist, $,\,(\tau)$ must have a root in the interval $[(r \Leftrightarrow 1)2^s \Leftrightarrow L, 0]$. However, the existence of such a root is not theoretically guaranteed. If there is no root in this interval, an arbitrary value for τ must be chosen and one degree of accuracy is lost. Once τ is determined, the weights are the solution of the linear system formed by r equations of (3.7). In order to construct $,\,(\tau)$ we write

$$p_i(\tau) \;=\; \sum_{j=0}^{r-i} p_{i,j}\,\tau^j \quad \text{and} \quad ,\,(\tau) \;=\; \sum_{i=0}^{r}\sum_{j=0}^{r-i} p_{i,j}\,\tau^j\,\mathcal{M}_i \;=\; \sum_{j=0}^{r}\left(\sum_{i=0}^{r-j}\mathcal{M}_i\,p_{i,j}\right)\tau^j.$$

The coefficients $p_{i,j}$ are symmetric ($p_{i,j} = p_{j,i}$), since the product polynomial is symmetric in τ and x, and can be found as $p_{i,j}^{(r)}$ where

$$\Pi^{(m)}(x) \;=\; \prod_{k=1}^{m}(x+\tau \Leftrightarrow a_k) \;=\; \sum_{i=0}^{m}\sum_{j=0}^{m-i} p_{i,j}^{(m)}\tau^j x^i.$$

An algorithm to calculate the $p_{i,j}$ can be derived by writing

$$\Pi^{(m)}(x) \;=\; (x+\tau \Leftrightarrow a_m)\,\Pi^{(m-1)}(x),$$

and identifying the coefficients of the powers of x and τ. It is described in detail in Algorithm 3.1.

A disadvantage of this construction is that the system of equations (3.7) is ill-conditioned if r is large. Table 3.1 shows the condition number of the linear system for the weights in the case of Daubechies' scaling functions and with $r = 2N \Leftrightarrow 1$. We see that the condition becomes very poor in case r is large and consequently the numerical results for the weights can no longer be trusted. In case $N = 8$, the system is even singular within working precision (16 decimal digits).

Algorithm 3.1: Construction of the product polynomial.

$p_{0,0}^{(0)} \leftarrow 1$
for $m \leftarrow 1\,(1)\,r$
  for $i \leftarrow 0\,(1)\,m$
    for $j \leftarrow 0\,(1)\,m \Leftrightarrow i$
      $p_{i,j}^{(m)} \leftarrow p_{i-1,j}^{(m-1)} + p_{i,j-1}^{(m-1)} \Leftrightarrow a_m \cdot p_{i,j}^{(m-1)}$
    end for
  end for
end for

Table 3.1: Condition number of the construction with monomial moments.

N	r	condition
2	3	$4\,10^1$
3	5	1.10^4
4	7	1.10^7
5	9	3.10^9
6	11	1.10^{13}
7	13	5.10^{16}
8	15	"singular"

3.4.2 Calculation of the moments

In order to construct the quadrature formula, we first need an algorithm to calculate the moments of a scaling function numerically. Simple algebraic manipulations lead to

$$\begin{aligned}
\mathcal{M}_p &= \int_{-\infty}^{+\infty} x^p\, \varphi(x)\, dx \\
&= 2 \sum_k h_k \int_{-\infty}^{+\infty} x^p\, \varphi(2x \Leftrightarrow k)\, dx \\
&= 2^{-p} \sum_k h_k \int_{-\infty}^{+\infty} (x+k)^p\, \varphi(x)\, dx \\
&= 2^{-p} \sum_{i=0}^{p} \binom{p}{i} \sum_k h_k\, k^i \int_{-\infty}^{+\infty} x^{p-i}\, \varphi(x)\, dx \\
&= 2^{-p} \sum_{i=0}^{p} \binom{p}{i} m_i\, \mathcal{M}_{p-i},
\end{aligned}$$

where the m_i are the discrete moments of the sequence $\{h_k\}$,

$$m_i \;=\; \sum_k k^i\, h_k.$$

Consequently the moments can be calculated with a p-terms recursion relation,

$$\mathcal{M}_p \;=\; \frac{1}{2^p \Leftrightarrow 1} \sum_{i=1}^{p} \binom{p}{i} m_i\, \mathcal{M}_{p-i}.$$

3.4.3 Modified construction

The ill-conditioning problem of the construction using monomial moments can be overcome if we use the basis of Chebyshev polynomials. This technique is also used successfully in [90, 147].

The *Chebyshev polynomial* $T_k(x)$ of degree k is defined by

$$T_k(\cos\theta) \;=\; \cos k\theta,$$

see [1]. The Chebyshev polynomials are orthogonal with respect to a weight function,

$$\frac{1}{\pi} \int_{-1}^{+1} \frac{1}{\sqrt{1 \Leftrightarrow x^2}}\, T_k(x)\, T_l(x)\, dx \;=\; \begin{cases} 1/2 & \text{if } k = l = 0 \\ 1 & \text{if } k = l \neq 0 \\ 0 & \text{otherwise,} \end{cases}$$

and have equioscillating properties in the interval $[\Leftrightarrow 1, 1]$.

Since the interesting properties of these polynomials only hold in the interval $[\Leftrightarrow 1, 1]$, we transform the scaling function $\varphi(x)$ to this interval giving a function

$\varphi^*(y)$ and use the notation y for an independent variable that varies between $\Leftrightarrow 1$ and 1,

$$2\,\varphi^*(y) \;=\; L\,\varphi(x) \quad \text{with} \quad 2x \;=\; L(y+1).$$

The refinement equation (2.4) becomes

$$\varphi^*(y) \;=\; 2\sum_k h_k\,\varphi^*(2y \Leftrightarrow 2k/L + 1).$$

We construct a quadrature formula,

$$\begin{aligned}\nu_{0,0} \;&=\; \int_0^L \varphi(x) f(x) dx \;=\; \int_{-1}^{+1} \varphi^*(y) f\left(\frac{L(y+1)}{2}\right) dy \\ &=\; \int_{-1}^{+1} \varphi^*(y) f^*(y) dy \;\approx\; \sum_{k=1}^{r} w_k^* f^*(y_k) \;=\; \sum_{k=1}^{r} w_k f(x_k) \;=\; Q_r[f(x)],\end{aligned}$$

with $w_k = w_k^*$, $y_k = a_k^* \Leftrightarrow \tau^*$, $a_k^* = 2a_k/L \Leftrightarrow 1$, and $\tau^* = 2\tau/L$. Let $\mathcal{M}_p^t$ denote the moments of the transformed scaling function

$$\mathcal{M}_p^t \;=\; \int_{-1}^{+1} y^p\,\varphi^*(y)\,dy,$$

and let $\mathcal{M}_p^*$ denote the *modified moments*,

$$\mathcal{M}_p^* \;=\; \int_{-1}^{+1} T_p(y)\,\varphi^*(y)\,dy.$$

The new system can be written as

$$\sum_{k=1}^{r} w_k\,T_i(a_k^* \Leftrightarrow \tau^*) \;=\; \mathcal{M}_i^* \qquad\qquad 0 \leqslant i \leqslant r. \tag{3.8}$$

The solution procedure is similar to the one in the previous section. We construct a polynomial , $^*(\tau^*)$, written as a linear combination of Chebyshev polynomials, and try to find a root in the appropriate interval. In order to construct , $^*(\tau^*)$ we write

$$\Pi(y) \;=\; 2^{-(r-1)} \sum_{i=0}^{r} \sum_{j=0}^{r-i} q_{i,j}\,T_j(\tau^*)\,T_i(y)$$

and

$$,\,^*(\tau^*) \;=\; 2^{-(r-1)} \sum_{j=0}^{r} \sum_{i=0}^{r-j} q_{i,j}\,\mathcal{M}_i^*\,T_j(\tau^*).$$

Now let

$$2^{(m-1)}\,\Pi^{(m)}(y) \;=\; 2^{(m-1)} \prod_{k=1}^{m} (y + \tau^* \Leftrightarrow a_k^*) \;=\; \sum_{i=0}^{m} \sum_{j=0}^{m-i} q_{i,j}^{(m)}\,T_j(\tau^*)\,T_i(y) \tag{3.9}$$

Table 3.2: Condition number of the construction with modified moments.

N	r	condition
2	3	$1\,10^0$
3	5	9.10^0
4	7	4.10^1
5	9	2.10^2
6	11	6.10^2
7	13	1.10^3
8	15	4.10^3

and

$$
\begin{aligned}
2^{(m-1)}\,\Pi^{(m)}(y) &= 2^{(m-1)}\,(y+\bar\tau \Leftrightarrow a_m)\,\Pi^{(m-1)}(y) \\
&= 2\sum_{i=0}^{m-1}\sum_{j=0}^{m-i-1} q_{i,j}^{(m-1)}\,(y+\bar\tau \Leftrightarrow a_m)\,T_j(\bar\tau)\,T_i(y) \\
&= \sum_{i=0}^{m-1}\sum_{j=0}^{m-1-i}\Big(q_{i,j}^{(m-1)}\,T_j(\bar\tau)\,T_{i+1}(y) + q_{i,j}^{(m-1)}\,T_j(\bar\tau)\,T_{|i-1|}(y) \\
&\qquad + q_{i,j}^{(m-1)}\,T_{j+1}(\bar\tau)\,T_i(y) + q_{i,j}^{(m-1)}\,T_{|j-1|}(\bar\tau)\,T_i(y) \\
&\qquad \Leftrightarrow 2\,a_m\,q_{i,j}^{(m-1)}\,T_j(\bar\tau)\,T_i(y)\Big) \\
&= \sum_{i=1}^{m}\sum_{j=0}^{m-i} q_{i-1,j}^{(m-1)}\,T_j(\bar\tau)\,T_i(y) + \sum_{i=-1}^{m-2}\sum_{j=0}^{m-2-i} q_{i+1,j}^{(m-1)}\,T_j(\bar\tau)\,T_{|i|}(y) \\
&\quad + \sum_{i=0}^{m-1}\sum_{j=1}^{m-i} q_{i,j-1}^{(m-1)}\,T_j(\bar\tau)\,T_i(y) + \sum_{i=0}^{m-1}\sum_{j=-1}^{m-2-i} q_{i,j+1}^{(m-1)}\,T_{|j|}(\bar\tau)\,T_i(y) \\
&\quad \Leftrightarrow 2\,a_m\sum_{i=0}^{m-1}\sum_{j=0}^{m-1-i} q_{i,j}^{(m-1)}\,T_j(\bar\tau)\,T_i(y). \qquad (3.10)
\end{aligned}
$$

An algorithm for the calculation of the $q_{i,j} = q_{i,j}^{(r)}$ can be found by identifying the coefficients of the Chebyshev polynomials of equal degree in x and in τ^* in (3.9) and (3.10). It is given in Algorithm 3.2.

The condition number of the system for the construction of the same formula as in the previous section is now given in Table 3.2. The condition is much better, especially for large r.

The roots of the polynomial , (τ^*) can be found as the eigenvalues of its Chebyshev companion matrix. The effect of an orthogonal basis on the condition of the roots of a polynomial is discussed in [91]. It is stated there that the interval of orthogonality should contain the roots of interest. This condition is satisfied in

Algorithm 3.2: Modified construction of the product polynomial.

$q_{0,0}^{(0)} \leftarrow 1/2$
for $m \leftarrow 1\,(1)\,r$
  for $i \leftarrow 0\,(1)\,m$
    for $j \leftarrow 0\,(1)\,m \Leftrightarrow i$
      $q_{i,j}^{(m)} \leftarrow q_{i-1,j}^{(m-1)} + q_{i,j-1}^{(m-1)} + q_{i+1,j}^{(m-1)} + q_{i,j+1}^{(m-1)} \Leftrightarrow 2\, a_m\, q_{i,j}^{(m-1)}$
      if i = 1 **then** $q_{1,j}^{(m)} \leftarrow q_{1,j}^{(m)} + q_{0,j}^{(m-1)}$
      if j = 1 **then** $q_{i,1}^{(m)} \leftarrow q_{i,1}^{(m)} + q_{i,0}^{(m-1)}$
    end for
  end for
end for

most cases here.

3.4.4 Calculation of the modified moments

It is possible to calculate the modified moment as a linear combination of the monomial moments using the coefficients $t_i^{(p)}$ of the Chebyshev polynomials,

$$T_p(y) = \sum_{i=0}^{p} t_i^{(p)} y^i \quad \text{and} \quad \mathcal{M}_p^* = \sum_{i=0}^{p} t_i^{(p)} \mathcal{M}_i^t. \tag{3.11}$$

However, a considerable loss of significant digits will occur since these coefficients tend to be large and different in sign. The condition would be as bad as in the construction using the monomial moments since calculating the modified moments like this essentially does not change the problem. We therefore need a formula to calculate the modified moments directly. We know that

$$\begin{aligned} \mathcal{M}_p^* &= \int_{-1}^{+1} T_p(y)\, \varphi^*(y)\, dy \\ &= 2 \sum_k h_k \int_{k/L-1}^{k/L} T_p(y)\, \varphi^*(2y + 1 \Leftrightarrow 2k/L)\, dy \\ &= \sum_k h_k \int_{-1}^{+1} T_p\left(\frac{u \Leftrightarrow 1 + 2k/L}{2}\right) \varphi^*(u)\, du. \end{aligned} \tag{3.12}$$

In order to find a recursion formula, we write this last shifted and dilated Chebyshev polynomial as a sum of Chebyshev polynomials of degree less than or equal to p,

$$T_p\left(\frac{y \Leftrightarrow 1 + 2k/L}{2}\right) = 2^{-p} \sum_{i=0}^{p} w_i^{(p)}(k)\, T_i(y).$$

Hence

$$\mathcal{M}_p^* = \frac{1}{2^p \Leftrightarrow 1} \sum_{i=0}^{p-1} \left(\sum_{k=0}^{L} h_k w_i^{(p)}(k) \right) \mathcal{M}_i^*. \tag{3.13}$$

The $w_i^{(p)}(k)$ can be calculated recursively. We will use the notation $w_i^{(p)} = w_i^{(p)}(k)$ and $\lambda = 2k/L \Leftrightarrow 1$ for simplicity here. Now

$$T_{p+1}\left(\frac{y+\lambda}{2}\right) \;=\; 2^{-(p+1)} \sum_{i=0}^{p+1} w_i^{(p+1)}\, T_i(y) \tag{3.14}$$

and

$$\begin{aligned}
T_{p+1}\left(\frac{y+\lambda}{2}\right) &= (y+\lambda)\, T_p\left(\frac{y+\lambda}{2}\right) \Leftrightarrow T_{p-1}\left(\frac{y+\lambda}{2}\right) \\
&= 2^{-p} \sum_{i=0}^{p} w_i^{(p)}\, y\, T_i(y) + 2^{-p} \sum_{i=0}^{p} w_i^{(p)} \lambda\, T_i(y) \Leftrightarrow 2^{-(p-1)} \sum_{i=0}^{p-1} w_i^{(p-1)} T_i(y) \\
&= 2^{-(p+1)} \left(\sum_{i=0}^{p} w_i^{(p)}\, T_{i+1}(y) + \sum_{i=0}^{p} w_i^{(p)}\, T_{|i-1|}(y) \right. \\
&\qquad \left. + 2\lambda \sum_{i=0}^{p} w_i^{(p)}\, T_i(y) \Leftrightarrow 4 \sum_{i=0}^{p-1} w_i^{(p-1)}\, T_i(y) \right) \\
&= 2^{-(p+1)} \left(\sum_{i=1}^{p+1} w_{i-1}^{(p)}\, T_i(y) + \sum_{i=-1}^{p-1} w_{i+1}^{(p)}\, T_{|i|}(y) \right. \\
&\qquad \left. + 2\lambda \sum_{i=0}^{p} w_i^{(p)}\, T_i(y) \Leftrightarrow 4 \sum_{i=0}^{p-1} w_i^{(p-1)}\, T_i(y) \right).
\end{aligned} \tag{3.15}$$

The algorithm can be found by identifying the coefficients of the Chebyshev polynomials of equal degree in (3.14) and (3.15). It is given in Algorithm 3.3. To check whether this method is numerically stable, we performed the following numerical experiment. We implemented (3.11) and (3.13) on a computer both in single precision (8 decimal digits) and double precision (16 decimal digits). Comparing the single and double precision result then gives the number of accurate digits of the single precision result. Table 3.3 gives the 30 first modified moments of the orthogonal Daubechies scaling function with $N = 2$. The accurate digits are underlined. It is obvious that (3.13) is stable and (3.11) is not. Note that the purpose of this experiment is not so much to show that the first method is unstable, which can easily be foreseen, but to show that the second is stable.

Table 3.3: Calculation of the modified moments.

p	using (3.11)	using (3.13)
1	–5.77350259 e–01	–5.77350259 e–01
2	–3.33333313 e–01	–3.33333303 e–01
3	8.15621793 e–01	8.15621853 e–01
4	–3.37566257 e–01	–3.37566108 e–01
5	–3.29552889 e–01	–3.29553157 e–01
6	2.66140223 e–01	2.66139805 e–01
7	1.06536865 e–01	1.06537372 e–01
8	–1.27771378 e–01	–1.27768755 e–01
9	2.50720978 e–02	2.50720195 e–02
10	2.38552094 e–02	2.38473043 e–02
11	–9.77897644 e–02	–9.78003293 e–02
12	3.16085815 e–02	3.16353105 e–02
13	1.03759766 e–01	1.03776902 e–01
14	–5.28259277 e–02	–5.30452169 e–02
15	–6.67419434 e–02	–6.67688027 e–02
16	5.24291992 e–02	5.29438257 e–02
17	1.97753906 e–02	1.90952606 e–02
18	–2.29492188 e–02	–2.47176141 e–02
19	9.27734375 e–03	1.74157508 e–02
20	–2.53906250 e–02	–1.71462391 e–02
21	–1.95312500 e–03	–3.38922292 e–02
22	6.25000000 e–02	4.09903340 e–02
23	8.20312500 e–02	3.17543037 e–02
24	–7.81250000 e–03	–3.69057357 e–02
25	–8.28125000 e–01	–2.07382068 e–02
26	–1.84375000 e+00	2.48349365 e–02
27	1.01875000 e+01	1.16013910 e–03
28	1.75000000 e+01	–1.61309689 e–02
29	–6.32500000 e+01	2.14398224 e–02
30	–1.30000000 e+02	3.96057591 e–03

Algorithm 3.3: Calculation of the modified moments.

$w_0^{(0)} \leftarrow 1$

$w_0^{(1)} \leftarrow \lambda$

$w_1^{(1)} \leftarrow 1$

$w_0^{(2)} \leftarrow 2\lambda^2 \Leftrightarrow 3$

$w_1^{(2)} \leftarrow 4\lambda$

$w_2^{(2)} \leftarrow 1$

for $p \leftarrow 2\ (1)\ \ldots$

$w_0^{(p+1)} \leftarrow w_1^{(p)} + 2\lambda w_0^{(p)} \Leftrightarrow 4w_0^{(p-1)}$

$w_1^{(p+1)} \leftarrow 2w_0^{(p)} + w_2^{(p)} + 2\lambda w_1^{(p)} \Leftrightarrow 4w_1^{(p-1)}$

for $i \leftarrow 2\ (1)\ p \Leftrightarrow 1$

$w_i^{(p+1)} \leftarrow w_{i-1}^{(p)} + w_{i+1}^{(p)} + 2\lambda w_i^{(p)} \Leftrightarrow 4w_i^{(p-1)}$

end for

$w_p^{(p+1)} \leftarrow w_{p-1}^{(p)} + 2\lambda w_p^{(p)}$

$w_{p+1}^{(p+1)} \leftarrow w_p^{(p)}$

end for

3.5 Error analysis of the quadrature formula

Remember that the criterion of the degree of accuracy was chosen to have accurate results in case of smooth functions. In this section we make this statement more precise and also show why we need the fast wavelet transform to calculate the coefficients on the coarser levels.

Let $F_r[f(x)]$ denote the error of the quadrature formula,

$$F_r[f(x)] \;=\; \int_{-\infty}^{+\infty} f(y)\,\varphi(y)\,dy \Leftrightarrow Q_r[f(x)].$$

Suppose that $f \in \mathcal{C}^{q+1}(\mathrm{supp}\,\varphi)$. One then immediately understands that the first $q+1$ terms of the Taylor formula of f around a point of $\mathrm{supp}\,\varphi$ are integrated exactly and that the error somehow depends on the $(q+1)$-th derivative of f. This also follows from Peano's theorem [71].

To get a more precise formulation in the case $r = q$, we can reason along the following lines. Let x_0 be an arbitrary point of the interval $(0, L)$ not equal to one of the abscissae and let P_r be the polynomial of degree r that interpolates the function f in $x_0, \ldots, x_r$. If f belongs to $\mathcal{C}^{r+1}[0, L]$ and has bounded derivatives, then [31],

$$\forall x \in \mathrm{supp}\,\varphi \;:\; \exists\, \xi(x) \in \mathrm{supp}\,\varphi \;:\; f(x) \;=\; P_r(x) + e_r(x)$$

with

$$e_r(x) \;=\; \frac{\Pi(x)\,(x \Leftrightarrow x_0)}{(r+1)!} f^{(r+1)}(\xi(x)).$$

Then

$$\begin{aligned} F_r[f(x)] \;&=\; F_r[P_r(x) + e_r(x)] \;=\; F_r[P_r(x)] + F_r[e_r(x)] \;=\; F_r[e_r(x)] \\ &=\; \langle\, \varphi, e_r \,\rangle \Leftrightarrow Q_r[e_r(x)] \\ &=\; \frac{1}{(r+1)!} \int_r \varphi(x)\,\Pi(x)\,(x \Leftrightarrow x_0)\, f^{(r+1)}(\xi(x))\,dx \end{aligned}$$

with $\xi(x) \in \mathrm{supp}\,\varphi$. This error estimate, however, is not very useful in practice. One usually does not know the $(r+1)$-th derivative of the function. Moreover, one has no control over the function $\xi(x)$. Deriving an upper bound usually leads to very pessimistic estimates.

However, our purpose is not so much to get accurate estimates on the error as to understand its asymptotic behavior. For the remainder of this section we switch back to the biorthogonal notation, i.e. the notation with tilde for the dual functions. Let $\lambda'_{n,l}$ be the computed approximation of the coefficients, where $\tilde{\varphi}$ is taken as weight function,

$$\lambda'_{n,l} \;=\; 2^{-n/2}\, Q[f(2^{-n}(x+l))].$$

It then follows that

$$\frac{\lambda_{n,l} \Leftrightarrow \lambda'_{n,l}}{\lambda_{n,l}} \;=\; \mathcal{O}(2^{-n(q+1)}). \tag{3.16}$$

If one now uses the fast wavelet transform to calculate the coefficients on the coarser levels, they will all have a relative error of order $2^{-n(q+1)}$. On the other hand, the use of the quadrature formula directly on the coarser levels j ($j < n$) will give an error of order $2^{-j(q+1)}$. This is precisely why we only use the quadrature formula on the finest level.

This analysis also helps us to choose the degree of accuracy. Remember from Chapter 2 that the multiresolution approximation of a function converges as

$$\|\mathcal{P}_n f \Leftrightarrow f\| \;=\; \mathcal{O}(h^N) \quad \text{with} \quad h = 2^{-n}.$$

This immediately shows that the degree of accuracy should at least be $N \Leftrightarrow 1$, otherwise the use of the quadrature formula ruins the convergence rate of the multiresolution approximation. In other words, let

$$\mathcal{P}'_n f(x) \;=\; \sum_l \lambda'_{n,l}\, \varphi_{n,l}(x),$$

then

$$\|\mathcal{P}'_n f \Leftrightarrow f\| \;=\; \mathcal{O}(h^N)$$

if $q \geqslant N \Leftrightarrow 1$. At this moment $q = N \Leftrightarrow 1$ seems the most natural choice. In Chapter 4 we show that there are situations in which one needs to choose $q > N \Leftrightarrow 1$.

Remember that the trapezoidal rule for $\widetilde{\varphi}(x)$ has a degree of accuracy equal to $\widetilde{N} \Leftrightarrow 1$. This means that it does not ruin the convergence rate of the multiresolution approximation in case $\widetilde{N} \geqslant N$. This is true in the orthogonal and semiorthogonal case.

Suppose we are only interested in one coefficient on a fixed level, let us say $\lambda_{0,0}$. One can then use a quadrature formula on a finer level n ($n > 0$) and use the fast wavelet transform to calculate $\lambda_{0,0}$. This scheme converges with an error of order h^{q+1}, with $h = 2^{-n}$. Using a generalization of Bernoulli polynomials it is possible to derive an asymptotic error expansion in powers of h for this scheme. This was done independently by Pierre Verlinden in [187] and Wolfgang Dahmen and Charles Micchelli in [61]. It can be seen as a generalization of the classic Euler-McLaurin formula to which it reduces in case φ is the box function. As a result it is possible to use convergence acceleration methods similar to Romberg integration.

Note: Different error estimates for quadrature formulae and related methods were proposed in [94]. It is shown here that the quadrature formula constructed in this chapter, except in an exotic case, give accurate approximations.

3.6 Fitting the formulae in the multiresolution analysis

3.6.1 Using a quadrature formula at the finest level

Suppose we have to calculate T coefficients $\lambda_{n,l}$ yielding a function v_n,

$$v_n(x) = \sum_{l=0}^{T-1} \lambda_{n,l}\, \varphi_{n,l}(x) \quad \text{with} \quad \lambda_{n,l} = \langle\, f, \widetilde{\varphi}_{n,l}\,\rangle\,.$$

The quadrature formula and error estimation yield

$$\lambda_{n,l} = \sqrt{h}\left[\sum_{k=1}^{r} w_k f\left(h\left((k \Leftrightarrow 1)2^s \Leftrightarrow \tau + l\right)\right) + \mathcal{O}(h^{r+1})\right] \quad \text{with} \quad h = 2^{-n}. \quad (3.17)$$

One usually wants to avoid evaluating (or "sampling") the function f at abscissae with spacing smaller than 2^{-n}. This means that $s \geqslant 0$. The total number of evaluations for the calculation of the T inner products is equal to $T + (r \Leftrightarrow 1)\, 2^s$. Note that this number is dominated by the first term and is only slightly dependent on r. As a result the one-point quadrature formula, which needs T evaluations, can be replaced with a quadrature formula with a higher degree of accuracy, which requires in total almost the same number of evaluations. The workload is equal to $T \cdot r$ multiplications and $T\,(r \Leftrightarrow 1)$ additions.

For a certain r, one usually wants to choose the maximal s to have abscissae spread out over the whole integration interval. The maximal s within the requirement that $(r \Leftrightarrow 1)2^s < L$, however, corresponds to the smallest admittance interval for τ. If for a formula with $s > 0$ no τ can be found, one can always try to find a formula with spacing 2^{s-1}.

3.6.2 Using a quadrature formula at the one but finest level

The restrictions $s \geqslant 0$ and $(r \Leftrightarrow 1)2^s < L$ imply that the degree of accuracy is limited to $q < L$. There is a way to get around this and achieve a higher degree of accuracy. We describe here a method to obtain a higher degree of accuracy in one part of the multiresolution tree at the cost of a lower degree of accuracy in the other part. The part of the tree, in which the degree of accuracy increases, is the subtree formed by the $\lambda_{n-1,l} \ldots \lambda_{0,l}$ and the $\gamma_{n-2,l} \ldots \gamma_{0,l}$. The part of the tree, in which the degree of accuracy decreases, is formed by the $\lambda_{n,l}$ and the $\gamma_{n-1,l}$.

The idea is to use function evaluations with spacing $h = 2^{-n}$ in a quadrature formula for the calculation of the coefficients at the one but finest level, namely the $\lambda_{n-1,l}$. This means that s has to be chosen equal to $\Leftrightarrow$1 whereas $r > L$, so that

$$\lambda_{n-1,l} = \sqrt{2\,h}\left[\sum_{k=1}^{r} w_k\, f\left(h\left((k \Leftrightarrow 1) \Leftrightarrow 2\tau + 2l\right)\right) + \mathcal{O}(h^{r+1})\right]. \quad (3.18)$$

The shift τ and the weights w_k are determined as described in Section 3.4. The number of evaluations for the calculation of T inner products is $2\,T + r \Leftrightarrow 2$. From these coefficients the multiresolution coefficients of the first part of the tree can be calculated using the decomposition scheme. The error on the coefficients of this part is $\mathcal{O}(h^{r+1})$.

For the calculation of the $\lambda_{n,l}$, we use a quadrature rule with $s = 0$ and $r = L$. The degree of accuracy is $L \Leftrightarrow 1$, since the value of the parameter τ is already determined by the first quadrature formula. The $\gamma_{n-1,l}$ can be calculated with one step of the decomposition scheme. The error of the coefficients of the second part of the tree is $\mathcal{O}(h^L)$.

This idea is particularly useful for applications like the one mentioned at the end of Section 3.5, where one was only interested in coefficients of the coarsest level. In fact, it is possible to "jump" over more than one level, i.e. use a quadrature formula with $s < \Leftrightarrow 1$ at the level $n + s$. This way one can achieve a very high degree of accuracy. Note that the error on the levels with $j > n + s$ still behaves like $\mathcal{O}(h^L)$.

A problem here is that as the number of abscissae grows large (> 30), the evaluation of the quadrature formula can become ill-conditioned. Also, it is not guaranteed that a higher degree of accuracy always gives a smaller error, even if the function is sufficiently smooth, because the constant in front of the error term can become very large. As a result we have a trade-off situation. One can experiment on how to choose the optimal s for a given scaling function.

3.7 Numerical results

We constructed quadrature formulae for the Daubechies orthogonal scaling functions and verified that the formulae Q_r with $2 \leqslant r \leqslant 2N \Leftrightarrow 1$ exist for $2 \leqslant N \leqslant 10$, i.e. the polynomial , (τ) has at least one root in the appropriate interval. For $r = 2$, one of the weights is always zero and we return to the one-point formula. In 9 of these 90 formulae, no τ could be found for the maximal value of s and s was taken one smaller. This was in most cases for $r = N$. For B-spline scaling functions, the quadrature formula Q_L exists for $2 \leqslant L \leqslant 10$ and Q_{2L} (in the sense of (3.18)) exists for $2 \leqslant L \leqslant 4$. The weights of these formulae can vary in sign, but their absolute value does not grow too large when the number of points increases. The evaluation of the quadrature formula is thus well-conditioned.

We compare now different quadrature formulae in a practical example. We construct several multiresolution trees, each with coarsest level 0 and finest level n, and this for several n. We compare each time $\nu_{0,0}$. Notice, however, that the error is of the same order in the whole multiresolution tree.

As an example we take for φ the Daubechies orthogonal scaling function with

Table 3.4: Error of the integration rules.

n	5.2^n	Trapezoidal rule	One-point formula	Q_5^*	Q_5	Q_{10}
0	5	7.08e-04	1.17e-02	6.13e-04	2.15e-03	-
1	10	4.17e-03	1.43e-03	9.78e-05	4.40e-05	1.03e-08
2	20	7.96e-04	1.76e-04	4.30e-06	6.51e-07	1.11e-12
3	40	1.15e-04	2.19e-05	1.52e-07	9.38e-09	4.21e-15
4	80	1.53e-05	2.74e-06	5.03e-09	1.38e-10	9.99e-16
5	160	1.98e-06	3.43e-07	1.61e-10	2.09e-12	-
6	320	2.50e-07	4.28e-08	5.10e-12	3.19e-14	-
7	640	3.15e-08	5.35e-09	1.60e-13	1.11e-16	-
8	1280	3.96e-09	6.69e-10	4.66e-15	-	-
9	2560	4.96e-10	8.37e-11	2.22e-16	-	-
10	5120	6.20e-11	1.04e-11	-	-	-

$N = 3$, $f(x) = \sin(x)$ and

$$\nu_{0,0} = \int_0^5 \varphi(x) \sin(x) dx \approx 0.741104421925905. \tag{3.19}$$

We compare the one-point formula, Q_5, Q_{10} (with $s = \Leftrightarrow 1$ applied at level $n \Leftrightarrow 1$), and the trapezoidal rule. The total number of evaluations is then, respectively, $5.2^n \Leftrightarrow 4$, 5.2^n, 5.2^n and $5.2^n \Leftrightarrow 1$. We also use a formula Q_5^* where τ was given a fixed value equal to $\Leftrightarrow 1/2$. The results are given in Table 3.4. They show that for sufficiently differentiable functions f, it is useful to search for the optimal value of the shift τ.

3.8 Related methods

Quadrature formulae are one possibility to approximate the multiresolution analysis coefficients from the function samples. In this section we discuss this problem in a more general setting and compare quadrature formulae with other solutions.

For notational simplicity we consider the case when $j = 0$. We also switch back to the notation with tilde for the dual functions. The problem can then be stated as: given the samples $d_k = f(k + \tau)$ of the function f, find a function in V_0 that, in some sense, approximates these samples. We write this function as

$$\sum_l \nu_l \, \varphi(x \Leftrightarrow l),$$

and refer to this problem as the *sampling problem*. Note that τ can be an unknown.

It is natural to assume that the coefficients ν_l are linear functionals of the samples d_k and that this relationship is invariant to integer translates. This implies that every solution can be written as a convolution. The general form of the solution thus involves a sequence $\{a_k\}$ so that

$$\{\nu_k\} \;=\; \{a_k\} * \{d_k\}.$$

This means that we have constructed a linear operator $\mathcal{S}$ from $\mathcal{C}^0$ to V_0, which can be written as

$$\mathcal{S}\, f(x) = \sum_l \nu_l\, \varphi(x \Leftrightarrow l),$$

which we refer to as the *sampling operator*.

Definition 3.5 *The sampling operator is local if a compact set S exists so that for all $f \in \mathcal{C}^0$ and $x \in \mathbf{R}$, $\mathcal{S}\, f(x)$ only depends on $f(y)$ for y in*

$$x + S \;=\; \{x + y \mid y \in S\}.$$

A local sampling operator is desirable because it is easy to implement. The following lemma gives a necessary condition for a sampling operator to be local.

Lemma 3.6 *A sampling operator is local if only a finite number of the coefficients a_k are non-zero and φ is compactly supported.*

In case of a quadrature formula it is immediately clear that the a_k are the weights. In the remainder of this section we study another solution, interpolation, in more detail and compare these solutions with ideas coming from a signal processing viewpoint.

3.8.1 Interpolation

The basic idea is to find a function of V_0 that interpolates the samples. We therefore introduce the following definition.

Definition 3.7 *A sampling operator $\mathcal{S}$ is interpolating if*

$$\mathcal{S}f(k+\tau) \;=\; d_k \;=\; f(k+\tau).$$

In this section we consider the case $\tau = 0$. We first consider the trivial case.

Definition 3.8 *A scaling function φ is interpolating if $\varphi(k) = \delta_k$.*

In this case the solution is immediately given by $\nu_l = f(l)$. The following lemma, which follows from the refinement relation, can help to characterize interpolating scaling functions.

Lemma 3.9 *If φ is interpolating, then $h_{2k} = \delta_k/2$.*

Note that the converse is not always true. The following function is a counterexample:

$$\varphi(x) \;=\; \begin{cases} 3+x & \text{for } x \in [\Leftrightarrow 3, 0) \\ 3 \Leftrightarrow x & \text{for } x \in [0, 3) \\ 0 & \text{elsewhere.} \end{cases}$$

It is a stretched hat function with $h_{-3} = h_3 = 1/4$, $h_0 = 1/2$ and the other h_k zero, and obviously is not interpolating. We will discuss how to construct interpolating scaling functions later in this section.

In general, the interpolation problem can be written as

$$d_k \;=\; \mathcal{S}\, f(k) \;=\; \sum_l \nu_l\, \varphi(k \Leftrightarrow l),$$

or

$$\{d_k\} \;=\; \{\nu_k\} * \{\varphi(k)\}. \tag{3.20}$$

This shows that the solution can by found by solving an inverse convolution problem. Algebraically it can be seen as inverting an infinite Toeplitz matrix with entries $\varphi(k \Leftrightarrow l)$. This matrix is banded in case the scaling function is compactly supported. Let $d(\omega)$, $\nu(\omega)$ and $p(\omega)$ respectively be the discrete Fourier transforms of the sequences $\{d_k\}$, $\{\nu_k\}$ and $\{\varphi(k)\}$. The relationship (3.20) can then be written as

$$d(\omega) \;=\; \nu(\omega)\, p(\omega).$$

This leads to the following result:

Lemma 3.10 *The interpolation problem (3.20) has a unique solution if $p(\omega)$ does not vanish.*

The solution is then given by

$$\{\nu_k\} \;=\; \{a_k\} * \{d_k\},$$

where

$$\sum_k a_k\, e^{-i\omega k} \;=\; \frac{1}{p(\omega)}.$$

This technique was studied in [4, 6, 194]. A problem here is that even if the scaling function is compactly supported, $1/p(\omega)$ in general is not a trigonometric polynomial. Hence, the a_k form an infinite sequence and the interpolating sampling operator is not local. In other words, each coefficient ν_l depends on all the data samples d_k. This is evidently not very useful computation-wise. It is, however, possible to show that the coefficients of $1/p(\omega)$ decay exponentially as $|k|$ tends to infinity. Consequently, if one sets forth a certain numerical accuracy, the infinite convolution can be broken off.

We can use this result to construct an interpolating scaling function whose integer translates span V_0 by letting

$$\widehat{\varphi}_{\text{interpol}}(\omega) \;=\; \frac{\widehat{\varphi}(\omega)}{p(\omega)}.$$

Again, even if φ is compactly supported, the interpolation function generally is not but instead it is exponentially decreasing. Typical examples are the cardinal spline interpolants of even order [158]. One exception here is the second order, where the B-spline itself (the hat function) is interpolating.

3.8.2 Shifted interpolation

In case $p(\omega)$ vanishes, an interpolating function does not exist. We can then, similar to the construction of the quadrature formulae, add some flexibility to the interpolation problem by not necessarily associating the data with the integer locations but allowing a shift $\tau \in (0,1)$. We formulate the problem now as

$$d_k \;=\; \sum_l \nu_l \, \varphi(k \Leftrightarrow l + \tau).$$

We have an extremely nice situation if shifting the scaling function yields an interpolating function. The following lemma then tells us what the shift should be.

Lemma 3.11 *If* $\varphi(\tau + k) \;=\; \delta_k$ *and* $N \geqslant 1$, *then* $\tau = \mathcal{M}_1$.

Proof : Follows immediately from the fact that

$$\sum_l (x \Leftrightarrow l)\, \varphi(x \Leftrightarrow l) \;=\; \mathcal{M}_1.$$

□

Note that this is exactly the same shift as in the case of the one-point quadrature formula. Unfortunately, this property is hardly ever satisfied. It was noted by Mary Ellen Bock and the author that in case of the orthogonal Daubechies scaling functions it is numerically almost satisfied. For the scaling function with $N = 2$, which has support $[0, 3]$, we checked that

$$\begin{aligned}
\mathcal{M}_1 &= (1+\sqrt{3})/4 \approx 0.683 \\
\varphi(\mathcal{M}_1) &\approx 1.00020859077 \\
\varphi(\mathcal{M}_1 + 1) &\approx \text{-4.17181539384e-04} \\
\varphi(\mathcal{M}_1 + 2) &\approx 2.08590769692\text{e-}04.
\end{aligned}$$

We calculated these values using the cascade algorithm, which allows us to determine the value of a scaling function locally. All the given digits are correct. One

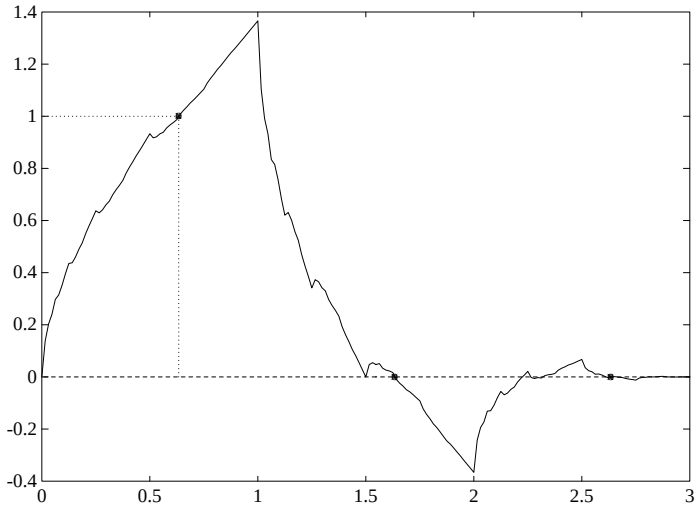

Figure 3.3: Daubechies' orthogonal scaling function with $N = 2$.

can use also Figure 3.3 for a graphical check. This means that letting $\nu_l = d_l$ gives a function that interpolates the data with a relative accuracy of 4.10^{-4}. In this case it is also possible to construct a very elegant geometric interpretation of the fast wavelet transform [26].

In the general case the solution to the shifted interpolating problem is given by

$$\nu(\omega) \;=\; \frac{d(\omega)}{\mathcal{Z}\varphi(\tau,\omega)},$$

provided that the denominator does not vanish [103]. Remember that $\mathcal{Z}$ stands for the Zak transform. It is almost always possible to find a τ so that $\mathcal{Z}\varphi(\tau,\omega)$ does not vanish. Note that $\mathcal{Z}\varphi(\tau,\omega)$ is a 2π-periodic function in ω.

The shifted interpolation is particularly useful when the scaling function is symmetric around a non-integer. Typical examples are the B-splines of odd order, which are symmetric around an integer + 1/2 such that $\tau = 1/2$ is a natural choice. In this case one cannot have $\tau = 0$, since $p(\omega)$ vanishes at $\omega = \pi$. Janssen studies more general criteria for the choice of τ in [103]. Unfortunately, this construction also almost always leads to a non-local sampling operator.

3.8.3 Quasi-interpolation

Because an interpolating sampling operator is generally a non-local sampling operator, we consider a less strict condition or so-called *quasi-interpolation*.

Definition 3.12 *A sampling operator is called a quasi-interpolating operator if it reproduces every polynomial of degree less than N and if it is local.*

Consequently, if f coincides with a polynomial of degree less than N on a bounded and closed interval I, then $\mathcal{S}\,f$ will also coincide with this polynomial on the interval I_o where

$$I_o \;=\; \{x \mid x + S \subset I\},$$

and S is the smallest set that can be used in Definition 3.5. This tells us that if f locally behaves like a polynomial of degree less than N, $\mathcal{S}\,f$ will locally coincide with this polynomial and thus interpolate f. This explains the term "quasi"-interpolation.

Quasi-interpolating operators for spaces of cardinal spline functions were studied in [34]. Completely analogous to the case of a quadrature formula it is now possible to derive an asymptotic error estimate. It should be no surprise that the error behaves like $\mathcal{O}(h^N)$.

It is immediately clear that a quadrature formula with degree of accuracy greater than or equal to N gives rise to a quasi-interpolating sampling operator, which is local in case φ is compactly supported. In fact, in case $q > N$, the quadrature formula does a little more, since the sampling operator constructed using it satisfies

$$\mathcal{P}_0\, x^p \;=\; \mathcal{S}\, x^p \quad \text{for} \quad N \leqslant p \leqslant q.$$

Note that these functions are no longer polynomials.

3.8.4 Compactly supported interpolating scaling functions

We present a different construction of an interpolating scaling function. It relies on the simple observation that any pair of biorthogonal scaling functions generates a new, interpolating, scaling function Φ by letting

$$\Phi(x) \;=\; \int_{-\infty}^{+\infty} \varphi(y + x)\, \overline{\widetilde{\varphi}(y)}\, dy.$$

The interpolation property immediately follows from the biorthogonality condition.

It is easy to see that the interpolating function satisfies a refinement relation with coefficients $H_k = \Phi(k/2)/2$ and where

$$H(\omega) \;=\; \sum_k H_k\, e^{-i\omega k} \;=\; h(\omega)\, \overline{\widetilde{h}(\omega)},$$

and

$$H(\omega) + H(\omega + \pi) \;=\; 1.$$

This interpolating scaling function has several nice properties. If the scaling function and its dual are compactly supported, so is the interpolation scaling function. In the case of an orthogonal scaling function, the interpolating function is just its autocorrelation function. It is smoother than φ and $\widetilde{\varphi}$, it is symmetric, and

it can reproduce the polynomials with degree less than $N + \widetilde{N}$. Note that the interpolating function does not generate the same multiresolution analysis as φ. One can build a new multiresolution analysis where the dual of the interpolating scaling function is formally the Dirac function such that $\widetilde{H}(\omega) = 1$. Following the construction of biorthogonal bases as described in Section 2.5, we see that a wavelet function that generates complementary spaces W_j can then be chosen as

$$\Psi(x) \;=\; \Phi(2x \Leftrightarrow 1).$$

The dual wavelet is then a linear combination of Dirac impulses and has $N + \widetilde{N}$ vanishing moments, or more precisely,

$$\hat{\widetilde{\Psi}}(\omega) = \Leftrightarrow e^{-i\omega}\,\overline{H(\omega + \pi)}.$$

Evidently this only yields a multiresolution analysis for function spaces where pointwise evaluation is a bounded operator. This means we need to impose some smoothness on the functions.

A fast wavelet transform with finite impulse response filters follows immediately from this construction. A disadvantage is that these filters introduce considerable aliasing in the fast wavelet transform. This is a typical example of a multiresolution analysis generated by a wavelet that does not have a vanishing moment, cf. the remark in Section 2.6. Note that the dual scaling function does not satisfy a partition of the unity. We will encounter some more examples in Chapter 6.

Recently Lemarié [124], Shensa [161], and Beylkin and Saito [157], noted that this construction, starting from the Daubechies orthogonal wavelets, yields a family of interpolating functions that were originally studied by Deslauriers and Dubuc in [75, 76]. They were also used for the characterization of function spaces in [80] and in signal processing applications in [157]. The interpolating scaling function constructed as the autocorrelation function of the Daubechies orthogonal scaling function with $N = 2$ is shown in Figure 3.4. In fact, one gets exactly the same function starting from any pair of compactly supported biorthogonal scaling functions with $N + \widetilde{N} = 4$.

These interpolation schemes are also closely related to stationary subdivision, see [32].

3.8.5 Signal processing approach

Researchers have also studied the sampling problem from a signal processing viewpoint [6, 161]. Their analysis is based on the fact that the functions of V_0 have most of their energy concentrated in the frequency band $[\Leftrightarrow\pi, \pi]$. This follows from the fact that $h(\omega)$ has most of its energy concentrated in the interval $[\Leftrightarrow\pi/2, \pi/2]$, see Section 2.7. In fact, in case the scaling function is taken to be the Shannon sampling function,

$$\varphi(x) = \frac{\sin(\pi x)}{\pi x},$$

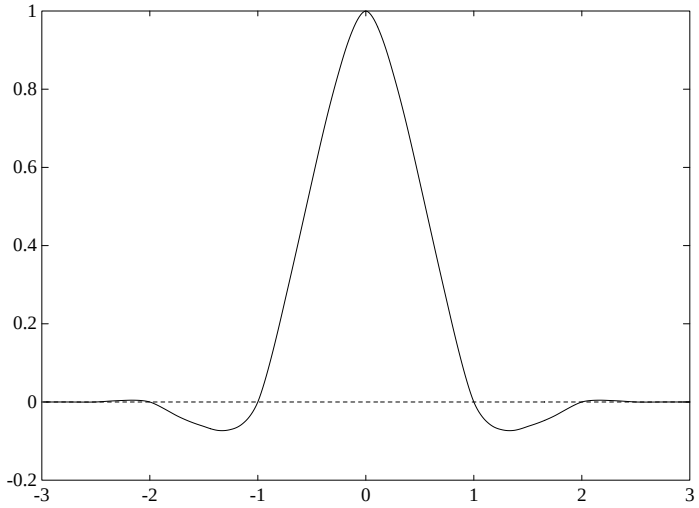

Figure 3.4: Deslauriers-Dubuc interpolating scaling function with $N + \widetilde{N} = 4$.

then all of the energy is concentrated exactly in these intervals. Note that this function is an orthogonal and interpolating scaling function, but that it dies off very slowly.

We continue the discussion on the Fourier transform side. Because of the shift invariance we can concentrate on the case $l = 0$:

$$\begin{aligned}
\nu_0 &= \sum_l a_{-l}\, f(l+\tau) \\
&= \frac{1}{2\pi} \sum_l a_{-l} \int_{-\infty}^{+\infty} e^{i\omega(l+\tau)}\, \widehat{f}(\omega)\, d\omega \\
&= \frac{1}{2\pi} \int_{-\infty}^{+\infty} \widehat{f}(\omega) \sum_l a_l e^{-i\omega l} e^{i\omega\tau}\, d\omega \\
&= \frac{1}{2\pi} \langle\, \widehat{f}, A \,\rangle\,,
\end{aligned}$$

where $A(\omega)$ is a 2π-periodic function,

$$A(\omega) \;=\; \overline{a(\omega)}\, e^{-i\omega\tau},$$

and $a(\omega)$ is de discrete Fourier transform of $\{a_k\}$. The Parseval identity gives the exact value

$$\lambda_0 \;=\; \frac{1}{2\pi} \langle\, \widehat{f}, \widehat{\widetilde{\varphi}} \,\rangle\,,$$

thus

$$\lambda_0 \Leftrightarrow \nu_0 \;=\; \frac{1}{2\pi} \langle\, \widehat{f}, \widehat{\widetilde{\varphi}} \Leftrightarrow \widehat{A} \,\rangle\,. \tag{3.21}$$

We can now formulate the problem as follows: find the unknowns τ and a_k so that $A(\omega)$ in some sense is a good approximation for $\widehat{\widetilde{\varphi}}(\omega)$ in the neighborhood of

the interval $[\Leftrightarrow\pi, \pi]$. Using this setting, we propose several solutions and compare them with the ones we already know.

1. The simplest solution just lets $a(\omega) = 1$, which corresponds to $\nu_l = a_l$. This means that $A(0) = \widehat{\widetilde{\varphi}}(0)$. To let the phase correspond at $\omega = 0$, one needs to choose $\tau = \widetilde{\mathcal{M}}_1$. Note that this is in fact a one-point quadrature formula.

2. The construction of a general quadrature formula lets the derivatives of $A(\omega)$ and $\widehat{\widetilde{\varphi}}(\omega)$ at $\omega = 0$ coincide up to the order equal to the degree of accuracy.

3. One can let $A(\omega)$ coincide with the 2π-periodization of $\widehat{\widetilde{\varphi}}(\omega)$,

$$A(\omega) \;=\; \sum_k \widehat{\widetilde{\varphi}}(\omega + k2\pi).$$

   Since the energy of $\widetilde{\varphi}(\omega)$ is concentrated in $[\Leftrightarrow\pi, \pi]$, this can be a reasonable approximation. Using the Poisson summation formula we see that

$$A(\omega) \;=\; \sum_l \widehat{\widetilde{\varphi}}(l)\, e^{-i\omega l}.$$

   This thus corresponds to the use of the trapezoidal rule as a quadrature formula.

4. One can let $A(\omega)$ coincide with $\widehat{\widetilde{\varphi}}(\omega)$ on the interval $[\Leftrightarrow\pi, \pi]$. This will lead to the exact result in case the function $f(x)$ is band-limited, with the band not exceeding $[\Leftrightarrow\pi, \pi]$. A sampling theorem similar to the classic Shannon sampling theorem can then be proven [5]. Note that $A(\omega)$ is not a trigonometric polynomial and the scheme thus is not local. The decay of the coefficients depends on the smoothness of $A(\omega)$ as a 2π-periodic function. The decay is here a little better than in the classic Shannon case because $\widehat{\widetilde{\varphi}}(\omega)$ has a root of multiplicity $\widetilde{N}$ at π and $\Leftrightarrow\pi$.

5. Remember that the interpolation solution corresponds to choosing

$$A(\omega) \;=\; 1/\overline{p(\omega)}.$$

   It is interesting to note that in the orthogonal case this is almost exactly the inverse of the solution given by the trapezoidal rule.

6. Evidently many other criteria and corresponding solutions can be suggested. One possibility is to fix the degree of a trigonometric polynomial $A(\omega)$ and then look for the L_∞ approximation of $\widehat{\widetilde{\varphi}}(\omega)$ in the interval $[\Leftrightarrow\pi, \pi]$.

This analysis shows the major difference between a numerical analysis and a signal processing point of view. In signal processing the sampling rate and thus the finest level of the multiresolution analysis are usually fixed. Moreover, one

has no a priori information about the smoothness of the data. Asymptotical error estimates, which typically rely on smoothness of the data and on the fact that one can increase the resolution, are thus useless. Therefore, solutions such as 4 and 6 are preferred. They try to minimize the error for all frequencies that are expected to be present in the signal.

In numerical analysis applications the situation is different. Here one tries to solve a mathematical problem that is formulated in terms of functions of a continuous variable, differential equations being a typical example. The solution is known to have some smoothness and one has control over the resolution. A quadrature formula here is the appropriate tool and the asymptotical error estimate predicts the behavior of the error in function of the resolution. On the Fourier transform side, increasing the resolution corresponds to "shrinking" the data towards zero in (3.21). Remember that the quadrature formula lets the derivatives of $A(\omega)$ and $\tilde{\varphi}(\omega)$ match at the origin.

3.9 Wavelets on an interval

So far our discussion only involved the case of the real line, which is invariant to integer shifts. Recently, several constructions of wavelets on an interval became available [12, 33, 46, 47, 141]. These constructions all have in common that the functions that are supported, in some sense, away from the endpoints correspond to the ones from the real line, while new basis functions are constructed near the endpoints (see also Section 2.9). One of the problems is that the shift invariance is lost at the boundary. Therefore, it is not immediately clear how the coefficients should correspond to the data. Fourier techniques cannot be used anymore.

One idea is to construct special quadrature formulae for the boundary scaling functions. We have done some experiments that show that this is feasible. One can derive a recursion formula to calculate the moments of the boundary functions and use them in an algorithm to find the weights. A disadvantage is that this method sometimes requires a non-uniform sampling. Several other solutions have been proposed.

In [47] a so-called *preconditioning* step is introduced. This is inspired by the fact that the coefficients of a polynomial in the V_0 space are not a polynomial sequence any more. The preconditioning step involves applying a linear transform to the data samples near the boundary such that in case of a polynomial data sequence one gets the coefficients of a polynomial in the V_0 space. This assures that smooth sequences will have small wavelet coefficients, which is one of the basic reasons why wavelets are suited for data compression.

In [12] so-called *recursive wavelets* are introduced. We explain this idea first on the real line. The V_0 space is here defined as the space of the functions that are piecewise constant on the intervals $[k, k+1)$. It is thus generated by "block" functions $\varphi_{0,l}$, which are orthogonal. For $j < 0$ we define the basis functions for

V_j and W_j recursively through the relations

$$\varphi_{j,k}(x) \;=\; \sum_k h_{m-2k}\,\varphi_{j+1,m}(x),$$

and

$$\psi_{j,k}(x) \;=\; \sum_k g_{m-2k}\,\varphi_{j+1,m}(x).$$

Here the sequences $\{h_k\}$ and $\{g_k\}$ come from a classic multiresolution analysis and correspond to basis functions φ and ψ. Similar relations define the dual functions. This assures that the wavelet coefficients can be calculated with the fast wavelet transform. Note that the wavelets here are not any longer the dilates and translates of one particular function. Note also that all basis and dual functions are piecewise constant on intervals of length one.

One assumes now that the samples d_l are inner products,

$$d_l \;=\; 2^{n/2}\,\langle\, f(x), \chi_{[0,1)}(x \Leftrightarrow k)\,\rangle\,.$$

One can then construct a recursive multiresolution scheme as follows:

$$\nu_{0,l} \;=\; 2^{-n/2}\,d_l \quad \text{and} \quad \nu_{j-1,l} \;=\; \sqrt{2}\sum_k h_{k-2l}\,\nu_{j,k}.$$

Here

$$\varphi_{j,l}(x) \;=\; 2^{j/2}\,\varphi^{(j)}(2^j x \Leftrightarrow l),$$

with $(j \leqslant 0)$

$$\varphi^{(j)}(x) \;=\; T^{-j}\,\chi_{[0,1)}(x),$$

and T an operator defined as

$$Tg(x) \;=\; 2\sum_k h_k\,g(2x \Leftrightarrow k).$$

It is proven in [63] that $\lim_{j\to-\infty} \varphi^{(j)}(x) \;=\; \varphi(x)$, where φ is the original scaling function corresponding with the sequence $\{h_k\}$. In fact, this scheme is applicable for a wide range of functions $\varphi^{(0)}$ with integral one. When applying the above construction for wavelets on a closed interval we obtain recursive wavelets for the interval. This construction has the advantage that sampling and preconditioning on the finest level become trivial.

3.10 Future research

Remember that the quadrature formulae and the related methods we discussed all use the samples of the function f. In [79] David Donoho considered the following

problem: suppose that not the samples of the function f are given but rather its inner products with the box functions or the so-called *boxcar coefficients*,

$$d_l \;=\; \int_l^{l+1} f(x)dx;$$

how to construct a multiresolution analysis? This problem is inspired by the fact that some physical devices such as a CDC camera rather generate boxcar coefficients than function samples. He gives a solution by constructing scaling functions, which are biorthogonal to the box functions so that the boxcar coefficients immediately are the coefficients of these scaling functions. A disadvantage of this solution is that one is restricted to this particular family of multiresolution analyses. Another solution, which does not have this disadvantage is the use of recursive wavelets as proposed in the previous section.

We here propose to use a generalization of quadrature formulae. We first state the problem in a more general setting. Suppose we are given the samples of the convolution of f with a "window" function g,

$$d_l \;=\; (f * g)(l) \;=\; \int_{-\infty}^{+\infty} f(x)\, g(l \Leftrightarrow x)\, dx.$$

Depending on the application, typical choices for g are box functions, Gaussians, B-splines, Dirac impulses, etc. We try to find the multiresolution analysis coefficient ν_0 of f as

$$\nu_0 \;=\; \sum_k w_k\, d_k \;=\; Q'[f(x)],$$

where Q' stands for a generalized quadrature formula. The weights w_k of the generalized quadrature formula can then be found by expressing that the quadrature formula gives the exact result in case f is a polynomial. This results in solving a linear system formed by the following equations:

$$\sum_k w_k \int_{-\infty}^{+\infty} x^p\, g(k \Leftrightarrow x)\, dx \;=\; \mathcal{M}_p \quad \text{for} \quad 0 \leqslant p < q.$$

This means that only the moments of the window function are needed in the construction. Is should be possible to derive an asymptotic error estimate similar to the one for a classic quadrature formula.

Another direction for future research is inspired by the work of Beylkin [23, 24]. He addresses the following problem: Given the projection of a function f in the space V_j, find a good approximation for the coefficients of $F(f)$, where F is a non-linear function. A canonical example is $F(u) \;=\; u^2$. One way to do this is to construct

$$\mathcal{P}_j F(\mathcal{P}_j f).$$

Here we again can use the same idea, i.e. construct a local scheme that is exact in the case of polynomials. In the case of the quadratic function this leads to the

evaluations of the coefficients

$$c_{k,l} = \int_{-\infty}^{+\infty} \varphi(x)\, \varphi(x \Leftrightarrow l)\, \varphi(x \Leftrightarrow k)\, dx.$$

These again can be found as solutions of an eigenvalue problem (see also [61]). One can expect that asymptotical error estimates similar to the ones for the quadrature formulae can be derived.

A third direction for further research is inspired by discussions with Pierre Verlinden. Instead of using quadrature formula for the calculation of wavelet coefficients, he wants to use wavelets to construct new quadrature formulae for general integrals. The basic idea is to construct a quadrature formula that integrates the wavelet approximation of the integrand. One of the advantages is that it would be very natural to build an adaptive integration scheme. Essentially, the wavelet coefficients will tell us if the function is locally well-behaved and thus if further refinement is needed.

Finally, in this chapter we always assumed that the function f is smooth. The study of the case where f has singularities opens another new direction for research. This involves the design of new recursion schemes for the moments and new construction algorithms for the quadrature formulae. We did some experiments for the case where the function has an algebraic singularity of the form x^α that show that this is feasible.

3.11 Concluding remarks

In this chapter we showed how one can construct quadrature formulae and when they can be used in a multiresolution analysis. In Chapter 4 we will study more carefully how the use of a quadrature formula affects the error of a multiresolution approximation. These quadrature formulae will be useful in the solution of ordinary differential equations in Chapter 6. The basic underlying idea of a quadrature formula, i.e. to construct a local scheme that is exact for polynomials, turns out to be extremely useful for the construction of wavelets biorthogonal with respect to a weight function in Chapter 5.

Chapter 4

Asymptotic Error Expansions for Wavelet Approximations

"It is one thing, to show a Man that he is in an Error,
and another, to put him in possession of Truth"
—John Locke, *An Essay Concerning Humane Understanding (1690).*

4.1 Introduction

For every classic approximation method used in numerical analysis, think of orthogonal polynomials or trigonometric functions, it is important to know and understand the behavior of the error. In this chapter we take a closer look at the error of a wavelet approximation. We therefore define the *error operator* as

$$\mathcal{E}_n \;=\; 1 \Leftrightarrow \mathcal{P}_n.$$

Again we concentrate on the case where f is a smooth function. We already know from Chapter 2 that the error asymptotically behaves as $\mathcal{O}(h^N)$, see (2.35). Here and throughout this chapter $h = 2^{-n}$. We will try to understand how the error spatially, i.e. in the variable x, depends on the behavior of f. We will do so by deriving an error expansion in powers of h, which is asymptotically valid for h approaching zero. This will lead to the definition of three new families of functions that are the building blocks of the expansion, and which we will refer to as *monowavelets*. We study their properties, and show how they can be used to understand the behavior of the error.

Using the error expansion, we will explain interpolating properties of the wavelet approximation and construct convergence acceleration algorithms. We also will show that the expansion is a well-suited tool for comparing approximation properties of different wavelet families. Therefore, we will need to study the dependencies in a multiresolution analysis. Finally, we will also try to understand how the use of a quadrature formula as described in Chapter 3 effects the error expansion.

4.2 Construction of the expansion

In this section we derive the asymptotical error expansion for $\mathcal{E}_n f$. Since we know that the error decays as $\mathcal{O}(h^N)$, we propose an expansion of the form

$$\mathcal{E}_n f(x) \;=\; \sum_{p=N}^{M} h^p \, U_{p-N}(x) + \mathcal{O}(h^{M+1}).$$

The idea is to first construct an expansion for $\mathcal{Q}_n$ and then use the fact that

$$\mathcal{E}_n \;=\; \sum_{j=n}^{\infty} \mathcal{Q}_j,$$

to find the expansion for $\mathcal{E}_n$. We want to work under very general conditions on the wavelet and dual wavelet. Therefore, we assume that $\psi, \tilde{\psi} \in \mathcal{D}^{M+1}$, $f \in \mathcal{C}^{M+1}$, and $f^{(l)}$ bounded for $l \leqslant M+1$. Recall now that

$$\mathcal{Q}_n f(x) \;=\; 2^n \sum_l \langle\, f(y), \tilde{\psi}(2^n y \Leftrightarrow l) \,\rangle \; \psi(2^n x \Leftrightarrow l) \;=\; \sum_l \gamma_{n,l} \, \psi(2^n x \Leftrightarrow l). \qquad (4.1)$$

We first construct an expansion for $\gamma_{n,l}$ and then use this equation to find the expansion for $\mathcal{Q}_n$. One way would be to use a Taylor expansion around $y = 2^{-n} l$. This, however, leads to an error expansion which is not very practical to work with [177]. Therefore, we start our construction by writing a Taylor formula around $y = x$ (where y is the integration variable in the inner products):

$$\begin{aligned}
\gamma_{n,l} &= 2^n \,\langle\, f(y), \tilde{\psi}(2^n y \Leftrightarrow l) \,\rangle \\
&= 2^n \,\langle \sum_{p=0}^{M} f^{(p)}(x) \, \frac{(y \Leftrightarrow x)^p}{p!} + f^{(M+1)}(\xi(x,y)) \frac{(y \Leftrightarrow x)^{M+1}}{(M+1)!}, \tilde{\psi}(2^n y \Leftrightarrow l) \,\rangle \\
&\quad \text{with } \xi(x,y) \text{ between } x \text{ and } y \\
&= \sum_{p=0}^{M} \frac{2^n \, f^{(p)}(x)}{p!} \,\langle\, (y \Leftrightarrow x)^p, \tilde{\psi}(2^n y \Leftrightarrow l) \,\rangle \; + \rho_{n,l}(x),
\end{aligned}$$

with

$$\rho_{n,l}(x) \;=\; 2^n \,\langle\, f^{(M+1)}(\xi(x,y)) \frac{(y \Leftrightarrow x)^{M+1}}{(M+1)!}, \tilde{\psi}(2^n y \Leftrightarrow l) \,\rangle \,.$$

Note that as the derivatives of f are bounded and $\tilde{\psi}$ belongs to $\mathcal{D}^{M+1}$, all the inner products are finite. The dual wavelet has N vanishing moments so that

$$2^n \,\langle\, (y \Leftrightarrow x)^p, \tilde{\psi}(2^n y \Leftrightarrow l) \,\rangle \;=\; 0 \quad \text{for} \quad 0 \leqslant p < N,$$

and thus the first N terms of the summation over p vanish. For $N \leqslant p \leqslant M$ we have, using the transformation $z = 2^n y \Leftrightarrow l$, that

$$2^n \,\langle\, (y \Leftrightarrow x)^p, \tilde{\psi}(2^n y \Leftrightarrow l) \,\rangle \;=\; \langle\, (hz + hl \Leftrightarrow x)^p, \tilde{\psi}(z) \,\rangle$$

$$\begin{aligned} &= h^p \langle \sum_{s=0}^{p} \binom{p}{s} z^{p-s} (l \Leftrightarrow 2^n x)^s, \tilde{\psi}(z) \rangle \\ &= h^p \sum_{s=0}^{p} \binom{p}{s} \widetilde{\mathcal{N}}_{p-s} (l \Leftrightarrow 2^n x)^s. \end{aligned}$$

The last N terms of this sum again vanish, so the upper bound of the summation over s can be $p \Leftrightarrow N$. Thus,

$$\gamma_{n,l} = \sum_{p=N}^{M} \frac{h^p f^{(p)}(x)}{p!} \sum_{s=0}^{p-N} \binom{p}{s} \widetilde{\mathcal{N}}_{p-s} (l \Leftrightarrow 2^n x)^s + \rho_{n,l}(x).$$

Combining this expansion and (4.1) yields that

$$\mathcal{Q}_n f(x) = \sum_{p=N}^{M} \frac{h^p f^{(p)}(x)}{p!} \sum_{s=0}^{p-N} \binom{p}{s} \widetilde{\mathcal{N}}_{p-s} (\Leftrightarrow 1)^s \sigma_s(2^n x) + K_n(x).$$

Here σ_p is the first monowavelet, which is defined as

$$\sigma_p(x) = \sum_l (x \Leftrightarrow l)^p \psi(x \Leftrightarrow l) \quad \text{for} \quad 0 \leqslant p < M \Leftrightarrow N,$$

and K_n is given by

$$K_n(x) = \sum_l \rho_{n,l}(x) \psi(2^n x \Leftrightarrow l).$$

Next we need to show that K_n behaves like $\mathcal{O}(h^N)$:

$$\begin{aligned} |K_n(x)| &\leqslant \sum_l |\rho_{n,l}(x)| \, |\psi(2^n x \Leftrightarrow l)| \\ &\leqslant \frac{\|f^{(M+1)}\|_\infty}{(M+1)!} 2^n \sum_l \langle \, |y \Leftrightarrow x|^{M+1}, |\tilde{\psi}(2^n y \Leftrightarrow l)| \, \rangle \, |\psi(2^n x \Leftrightarrow l)| \\ &= h^{M+1} \frac{\|f^{(M+1)}\|_\infty}{(M+1)!} \sum_l \langle \, |z + l \Leftrightarrow x/h|^{M+1}, |\tilde{\psi}(z)| \, \rangle \, |\psi(x/h \Leftrightarrow l)| \\ &\leqslant h^{M+1} \frac{\|f^{(M+1)}\|_\infty}{(M+1)!} \cdot \\ &\quad \sum_l \left[\sum_{j=0}^{M+1} m_j \, |x/h \Leftrightarrow l|^j \binom{M+1}{j} \right] |\psi(x/h \Leftrightarrow l)| \\ &\quad \text{with } m_j = \langle \, |z|^{M+1-j}, |\tilde{\psi}(z)| \, \rangle \ \text{(finite since } \tilde{\psi} \in \mathcal{D}^{M+1}) \\ &\leqslant h^{M+1} \frac{\|f^{(M+1)}\|_\infty}{(M+1)!} \max_{0 \leqslant j \leqslant M+1} m_j \cdot \\ &\quad \sum_l \left[\sum_{j=0}^{M+1} |x/h \Leftrightarrow l|^j \binom{M+1}{j} \right] |\psi(x/h \Leftrightarrow l)| \end{aligned}$$

$$= h^{M+1} \frac{\|f^{(M+1)}\|_\infty}{(M+1)!} \max_{0 \leqslant j \leqslant M+1} m_j \cdot \sum_l (|x/h \Leftrightarrow l| + 1)^{M+1} |\psi(x/h \Leftrightarrow l)|.$$

Since this last summation over l can be bounded independently of x and h, it holds that $|K_n(x)| \leqslant C\, h^{M+1}$ with C independent of x and n. Now we need to combine the error expansions for $\mathcal{Q}_n$ into one for $\mathcal{E}_n$. Therefore, we define a new monowavelet σ_p^* and write

$$\mathcal{Q}_n f(x) = \sum_{p=N}^{M} \frac{h^p\, f^{(p)}(x)}{p!}\, \sigma^*_{p-N}(2^n x) + K_n(x)\,, \tag{4.2}$$

with

$$\sigma_p^*(x) = \sum_{s=0}^{p} \binom{N+p}{s} \widetilde{\mathcal{N}}_{N+p-s}\, (\Leftrightarrow 1)^s\, \sigma_s(x). \tag{4.3}$$

So

$$\mathcal{Q}_{n+j} f(x) = \sum_{p=N}^{M} \frac{h^p\, f^{(p)}(x)}{p!\, 2^{jp}}\, \sigma^*_{p-N}(2^{n+j} x) + K_{n+j}(x).$$

Finally, adding the projections $\mathcal{Q}_{n+j}\, f$ yields the desired expansion,

$$\mathcal{E}_n f(x) = \sum_{p=N}^{M} \frac{h^p\, f^{(p)}(x)}{p!}\, \tau_{p-N}(2^n x) + \mathcal{O}(h^{M+1}). \tag{4.4}$$

Here τ_p is the third monowavelet which is defined as

$$\tau_p(x) = \sum_{j=0}^{\infty} \frac{\sigma_p^*(2^j x)}{2^{j(p+N)}}.$$

The error is still $\mathcal{O}(h^N)$ because

$$|K_{n+j}(x)| \leqslant C'\, h^{M+1}/2^{j(M+1)},$$

and thus

$$\sum_{j=n}^{\infty} |K_{n+j}(x)| \leqslant C\ h^{M+1}.$$

We conclude by saying that the general term of the expansions consists of: a power of h, the same order of derivative of f, and a monowavelet. We can look at the monowavelet as the "oscillating" part and at the derivative as the "modulating" part.

4.3 Properties of monowavelets

4.3.1 Definition

Recall that the monowavelet σ_p is defined as

$$\sigma_p(x) \;=\; \sum_l (x \Leftrightarrow l)^p \, \psi(x \Leftrightarrow l). \tag{4.5}$$

It is the one-periodization of $x^p \, \psi(x)$. If $\psi \in \mathcal{D}^p$ then the series (4.5) converges uniformly on $[0,1]$ and σ_p is bounded. This can be seen using the Weierstrass M-test combined with the fact that $x^p \psi(x) \in \mathcal{D}^0$. One can check that this condition was always satisfied in the previous section. From the definition it follows that

$$\int_0^1 \sigma_p(x)\, dx \;=\; \mathcal{N}_p \quad \text{for} \quad p \in \mathbf{N},$$

and thus the first $\widetilde{N}$ monowavelets have a vanishing mean.

The monowavelet τ_p is defined as

$$\tau_p(x) \;=\; \sum_{j=0}^{\infty} \frac{\sigma_p^*(2^j x)}{2^{j(p+N)}}. \tag{4.6}$$

The series (4.6) converges uniformly on $[0,1]$, and the monowavelet is periodic with period one. Also,

$$\sigma_p^*(x) \;=\; \tau_p(x) \Leftrightarrow \frac{\tau_p(2x)}{2^{N+p}}, \tag{4.7}$$

and

$$\begin{aligned}
\int_0^1 \tau_p(x)\, dx \;&=\; \frac{2^{(N+p)}}{2^{(N+p)} \Leftrightarrow 1} \int_0^1 \sigma_p^*(x)\, dx \\
&=\; \frac{2^{(N+p)}}{2^{(N+p)} \Leftrightarrow 1} \sum_{s=0}^{p} \binom{N+p}{s} (\Leftrightarrow 1)^s \, \widetilde{\mathcal{N}}_{N+p-s} \, \mathcal{N}_s \\
&=\; 0 \quad \text{if} \quad p < \widetilde{N}.
\end{aligned}$$

Again the first $\widetilde{N}$ monowavelets have a vanishing mean.

4.3.2 Invariance

There are obviously many possible choices for the wavelet ψ whose translates and dilates generate the same subspaces W_j. A trivial alternative would just be $\psi(x \Leftrightarrow 1)$. From its definition we see that the function σ_p depends on the particular

choice for ψ. This is not true for σ_p^* and τ_p. Writing (4.2) with $n = 0$ in case f is a monomial x^p with $p > N$ yields

$$\mathcal{Q}_0\, x^p \;=\; \sum_{s=N}^{p} \binom{p}{s}\, x^{p-s}\, \sigma^*_{s-N}(x). \tag{4.8}$$

It follows that σ_p^* only depends on the multiresolution analysis subspaces W_j and not on which particular functions $\psi_{j,k}$ generate it. So σ_p^* is more characteristic for a multiresolution analysis than σ_p. The same is true for τ_p as

$$\mathcal{E}_0\, x^p \;=\; (1 \Leftrightarrow \mathcal{P}_0)\, x^p \;=\; \sum_{s=N}^{p} \binom{p}{s}\, x^{p-s}\, \tau_{s-N}(x). \tag{4.9}$$

These dependencies are studied in more detail in Section 4.8. Note that

$$\tau_0(x) \;=\; x^N \Leftrightarrow \mathcal{P}_0\, x^N.$$

This is the error of the approximation of the lowest degree monomial that cannot be approximated exactly. Equation (4.9) generalizes this to higher degree monomials.

This also explains the name "monowavelets". The monowavelets come from periodizing a *mono*mial multiplied with a wavelet or from projecting down *mono*mials. These techniques are also used in spline theory where the resulting functions are called *monosplines* [159].

4.3.3 Fourier series

Write the Fourier series of σ_p as

$$\sigma_p(x) \;=\; \sum_k s_{p,k}\, e_k(x),$$

with

$$e_k(x) \;=\; \exp(2\pi i k x),$$

and

$$s_{p,k} \;=\; \int_0^1 \sigma_p(x)\, \overline{e_k(x)}\, dx.$$

Poisson's summation formula yields

$$s_{p,k} \;=\; i^p\, \widehat{\psi}^{(p)}(2k\pi).$$

The coefficients with even index can be written as

$$\begin{aligned} s_{p,2k} \;=\; i^p\, \widehat{\psi}^{(p)}(4k\pi) \;&=\; i^p\, \frac{d^p}{d\omega^p} \Big[g(\omega/2)\, \widehat{\varphi}(\omega/2) \Big]_{\omega = 4k\pi} \\ &=\; (i/2)^p\, \frac{d^p}{d\omega^p} \Big[g(\omega)\, \widehat{\varphi}(\omega) \Big]_{\omega = 2k\pi}. \end{aligned} \tag{4.10}$$

If $p < \widetilde{N}$, these terms vanish, since $2k\pi$ is a root of order $\widetilde{N}$ of $g(\omega)$. So only the odd index terms, which are antisymmetric around 1/2, remain and thus

$$\sigma_p(x + 1/2) \;=\; \Leftrightarrow\sigma_p(x), \tag{4.11}$$

or

$$\sum_l (x + l/2)^p \, \psi(x + l/2) \;=\; 0 \quad \text{for} \quad p < \widetilde{N}.$$

So the 1/2-periodization of $x^p \, \psi(x)$ is exactly zero if $p < \widetilde{N}$. We also know that $2k\pi$, $k \neq 0$ is a root of order N of $\widehat{\varphi}(\omega)$. This, together with (4.10), yields that if $\widetilde{N} \leqslant p < N_{\text{tot}} = \widetilde{N} + N$, the $s_{p,2k}$ with $k \neq 0$ vanish, or

$$s_{p,2k} \;=\; i^p \widehat{\psi}^{(p)}(0) \, \delta_k \;=\; \mathcal{N}_p \, \delta_k.$$

So

$$\sigma_p(x + 1/2) + \sigma_p(x) \;=\; 2\,\mathcal{N}_p, \tag{4.12}$$

or

$$\sum_l (x + l/2)^p \, \psi(x + l/2) \;=\; 2\,\mathcal{N}_p \quad \text{for} \quad \widetilde{N} \leqslant p < N_{\text{tot}}.$$

The σ_p^* have Fourier series

$$\sigma_p^*(x) \;=\; \sum_k s_{p,k}^* \, e_k(x),$$

with

$$s_{p,k}^* \;=\; \sum_{j=0}^{p} \binom{N+p}{j} \widetilde{\mathcal{N}}_{N+p-j} \, (\Leftrightarrow 1)^j \, s_{j,k}.$$

Since they are defined as finite linear combinations of the σ_p, the σ_p^* with $p < \widetilde{N}$ also have vanishing even coefficients in their Fourier series and thus satisfy

$$\sigma_p^*(x + 1/2) \;=\; \Leftrightarrow\sigma_p^*(x) \quad \text{for} \quad p < \widetilde{N}. \tag{4.13}$$

The τ_p have Fourier series

$$\tau_p(x) \;=\; \sum_k t_{p,k} \, e_k(x). \tag{4.14}$$

Writing the Fourier series of both sides of (4.7) yields

$$t_{p,2k+1} \;=\; s_{p,2k+1}^* \quad \text{and} \quad t_{p,2k} \;=\; s_{p,2k}^* + \frac{t_{p,k}}{2^{N+p}},$$

or, if $p < \widetilde{N}$ and thus $s_{p,2k}^* = 0$,

$$t_{p,k} \;=\; \frac{s_{p,2l+1}^*}{2^{m(N+p)}} \quad \text{with} \quad k = 2^m(2l+1). \tag{4.15}$$

The transition from σ_p to τ_p apparently corresponds to filling in the gaps at the even indices in the Fourier spectrum.

4.3.4 Zeros

Lemma 4.1 *If ψ is continuous and in $\mathcal{D}^p$ with $p < \widetilde{N}$, the monowavelets σ_p and σ_p^* have at least two zeros in the interval $[0, 1)$.*

Proof : If ψ is in $\mathcal{D}^p$, then the series (4.5) converges uniformly and σ_p is continuous. Since σ_p^* is a finite linear combination, it is continuous too. The proof then immediately follows from (4.11) and (4.13). Also, if x_0 is a root in $[0, 1)$, so is $(x_0 + 1/2) \bmod 1$. □

Lemma 4.2 *If ψ is continuous and in $\mathcal{D}_p$ with $p < \widetilde{N}$ and $N > 1$, the monowavelet τ_p has at least two zeros in the interval $[0, 1)$.*

Proof : The function τ_p is defined as the limit of a uniformly convergent series of continuous functions, so it is continuous. We have, using (4.13) and (4.14),

$$\tau_p(0) \;=\; \tau_p(1) \;=\; \frac{2^{(N+p)}}{2^{(N+p)} \Leftrightarrow 1}\sigma_p^*(0) \quad \text{and} \quad \tau_p(1/2) \;=\; \Leftrightarrow\frac{2^{(N+p)} \Leftrightarrow 2}{2^{(N+p)} \Leftrightarrow 1}\sigma_p^*(0).$$

This means we have at least two changes of sign. □

4.3.5 Symmetry

If the wavelet is even or odd,

$$\psi(\Leftrightarrow x) \;=\; (\Leftrightarrow 1)^m\, \psi(x),$$

so are σ_p and σ_p^* and, more precisely,

$$\sigma_p(\Leftrightarrow x) \;=\; (\Leftrightarrow 1)^{m+p}\, \sigma_p(x) \quad \text{and} \quad \sigma_p^*(\Leftrightarrow x) \;=\; (\Leftrightarrow 1)^{m+p}\, \sigma_p^*(x).$$

More generally, if the wavelet is (anti)symmetric around an integer k,

$$\psi(2k \Leftrightarrow x) \;=\; (\Leftrightarrow 1)^m\, \psi(x),$$

then σ_p^* is (anti)symmetric, or

$$\sigma_p^*(\Leftrightarrow x) \;=\; (\Leftrightarrow 1)^{m+p}\, \sigma_p^*(x).$$

This is true because the function $\psi(x \Leftrightarrow k)$ generates the same space W_j and thus gives rise to the same σ_p^* function while $\psi(x \Leftrightarrow k)$ is even or odd. Note that we cannot make a simple statement about σ_p. The following statements regarding the zeros of σ_p (and consequently of σ_p^*) and τ_p, for $p < \widetilde{N}$ hold:

- If σ_p is odd, it has zeros at the integers because of $\sigma_p(0) = 0$ and the periodicity. It then also has zeros at the integers + 1/2 because of (4.11).
- If σ_p is even, this combined with (4.11) yields $\sigma_p(x) = \Leftrightarrow\sigma_p(1/2 \Leftrightarrow x)$. It thus has zeros at the integers + 1/4, and again because of (4.11) also at the integers + 3/4.
- If τ_p is odd, it has zeros at the half integers; if it is even we cannot tell more about the position of its zeros this easily.

4.3.6 Connection with scaling function

The relationship between σ_p for $p < N_{\rm tot}$ and the scaling function can be written more explicitly using the Zak transform. Remember that the Zak transform of a function $f \in \mathrm{L}_2(\mathbf{R})$ is defined as [103, 104]:

$$(\mathcal{Z}f)(x,\omega) \;=\; \sum_l e^{-i\omega l}\, f(x+l) \qquad \text{for} \qquad x,\omega \in \mathbf{R},$$

and satisfies

$$(\mathcal{Z}f)(x,\omega) \;=\; \sum_k \widehat{f}(\omega+2\pi k)\, e^{i(\omega+2\pi k)x}.$$

Define now:

$$\gamma_p \;=\; i^p\, g^{(p)}(\pi) \;=\; \sum_k k^p\, (\Leftrightarrow 1)^k\, g_k.$$

Then

$$\begin{aligned} s_{p,2k+1} &= i^p \frac{d^p}{d\omega^p}\Big[g(\omega/2)\,\widehat{\varphi}(\omega/2)\Big]_{\omega=(2k+1)2\pi} \\ &= \left(\frac{i}{2}\right)^p \sum_{s=0}^{p} \binom{p}{s} g^{(p-s)}(\pi)\,\widehat{\varphi}^{(s)}((2k+1)\pi) \\ &= 2^{-p} \sum_{s=0}^{p} \binom{p}{s} \gamma_{p-s}\, i^s\, \widehat{\varphi}^{(s)}((2k+1)\pi). \end{aligned}$$

Now, for $p < \widetilde{N}$,

$$\begin{aligned} \sigma_p(x) &= (\mathcal{Z}\, x^p\, \psi)(x,0) \\ &= \sum_k s_{p,2k+1}\, e_{2k+1}(x) \\ &= 2^{-p} \sum_{s=0}^{p} \binom{p}{s} \gamma_{p-s} \sum_k i^s\, \widehat{\varphi}^{(s)}((2k+1)\pi)\, e_{2k+1}(x) \\ &= 2^{-p} \sum_{s=0}^{p} \binom{p}{s} \gamma_{p-s}\, (\mathcal{Z}\, x^s \varphi)(2x,\pi) \\ &= 2^{-p} \sum_{s=0}^{p} \binom{p}{s} \gamma_{p-s} \sum_l (\Leftrightarrow 1)^l\, (2x \Leftrightarrow l)^s\, \varphi(2x \Leftrightarrow l). \end{aligned}$$

For $\widetilde{N} \leqslant p < N_{\rm tot}$ one has that

$$\sigma_p(x) \;=\; \mathcal{N}_p + 2^{-p} \sum_{s=0}^{p} \binom{p}{s} \gamma_{p-s} \sum_l (\Leftrightarrow 1)^l\, (2x \Leftrightarrow l)^s\, \varphi(2x \Leftrightarrow l).$$

In case $p = 0$ we have that

$$s_{0,2k+1} \;=\; g(\pi)\,\widehat{\varphi}((2k+1)\pi) \;=\; \widehat{\varphi}((2k+1)\pi), \tag{4.16}$$

and

$$\sigma_0(x) \;=\; \sum_l (\Leftrightarrow 1)^l \varphi(2x \Leftrightarrow l) \;=\; 2\sum_l \varphi(2x \Leftrightarrow 2l) \Leftrightarrow 1. \tag{4.17}$$

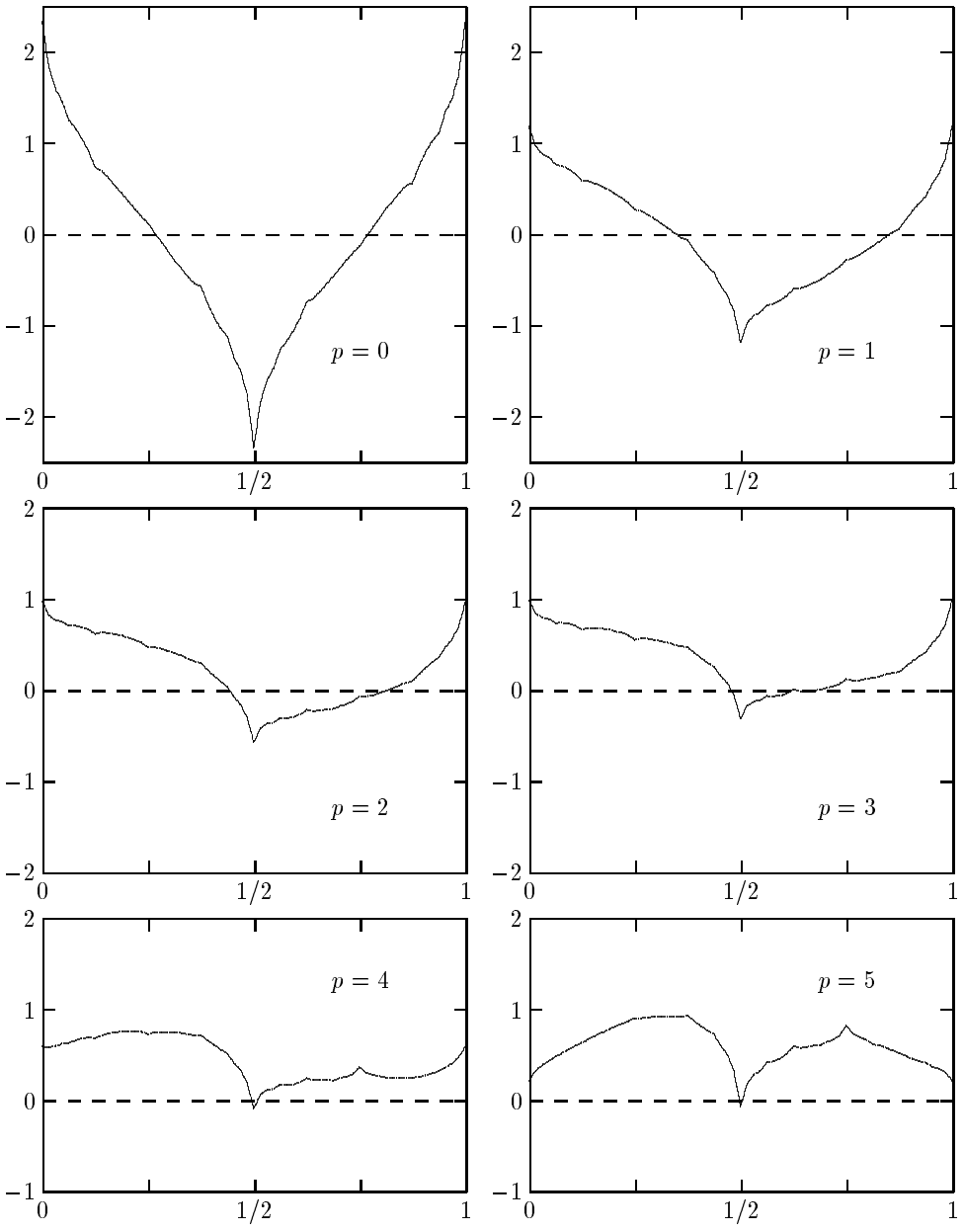

Figure 4.1: σ_p for coiflet with $N = 2$.

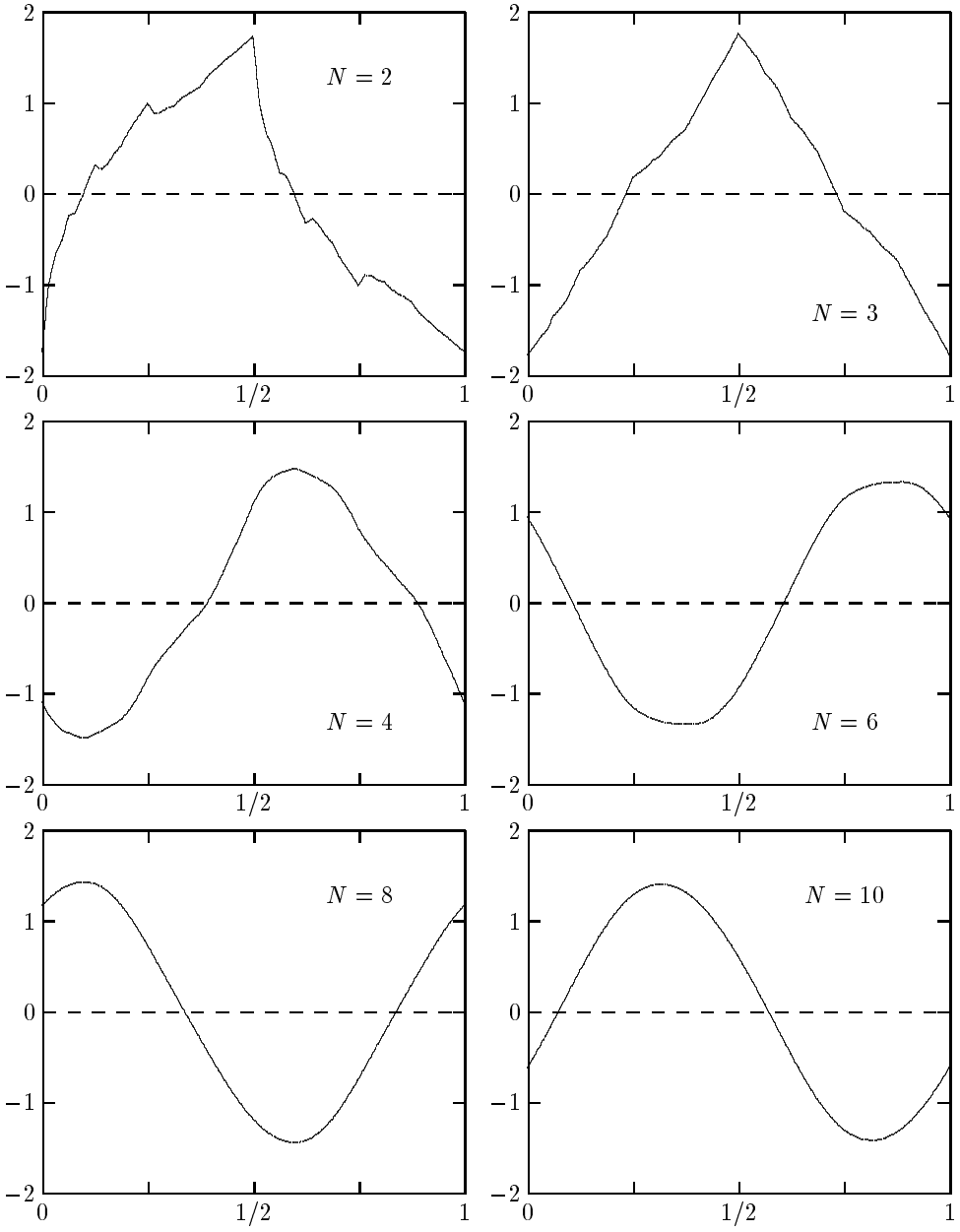

Figure 4.2: σ_0 for Daubechies' wavelets with $N = 2, 3, 4, 6, 8, 10$.

4.4 Examples of monowavelets

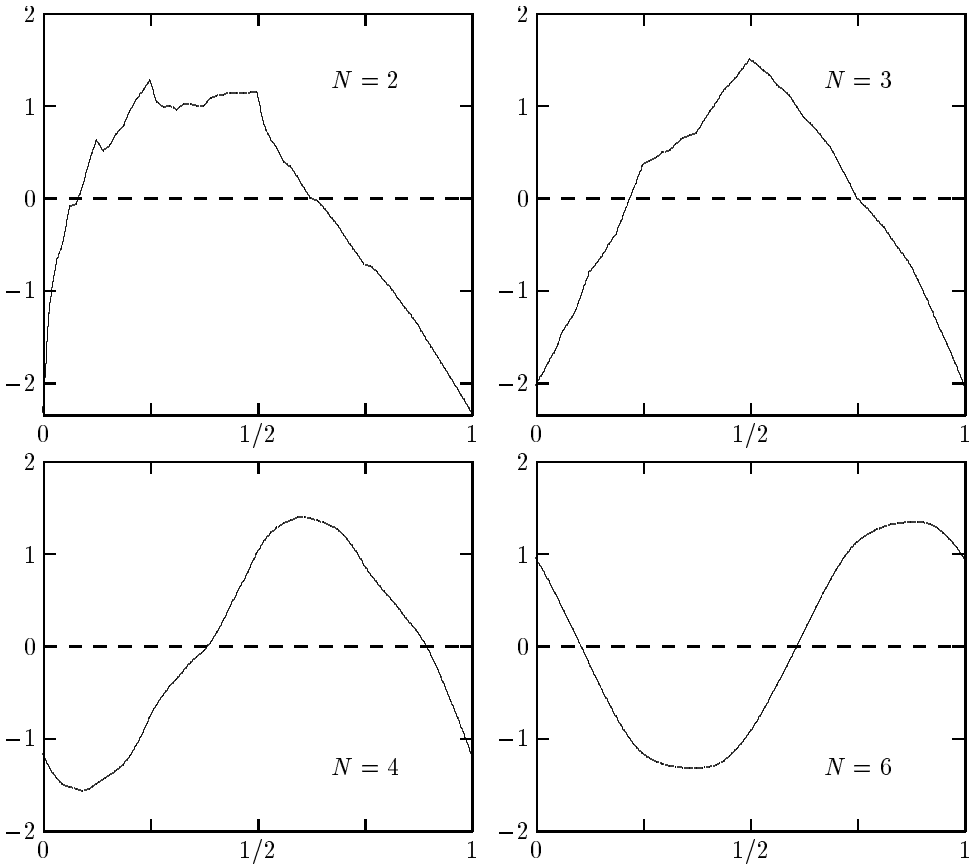

Figure 4.3: τ_0 for Daubechies' wavelets with $N = 2, 3, 4, 6$.

In the Haar case, $\sigma_0\,\chi_{[0,1]}$ is the Haar wavelet and τ_0 is a sawtooth function with $\tau_0(x) = x \Leftrightarrow 1/2$ for $0 \leqslant x < 1$. The $\sigma_0(2^j x)$ functions with $j \geqslant 0$ are then called the Rademacher functions, which are well-known in probability theory.

However, it is not always possible to derive an analytic expression for the monowavelets. We therefore wrote a computer program that allows us to numerically evaluate them. It first constructs the wavelet and then just uses the definition of the monowavelets. It uses ten terms in the definition of τ_p. Using this program we constructed the following graphs.

Figure 4.1 shows the first six σ_p monowavelets in case ψ is the coiflet with $N = 2$. One can graphically check that the 1/2-periodization of the first two is zero and of the following two a constant. The next examples concentrate on the leading terms ($p = 0$).

Figures 4.2 and 4.3 show σ_0 and $\tau_0/\mathcal{N}_N$ in the case of the orthogonal Daubechies wavelets. One can understand that σ_0 and $\tau_0/\mathcal{N}_N$ cannot be distinguished graphically for N bigger than 6. Note that for $N = 10$, the monowavelet looks like a shifted sine function.

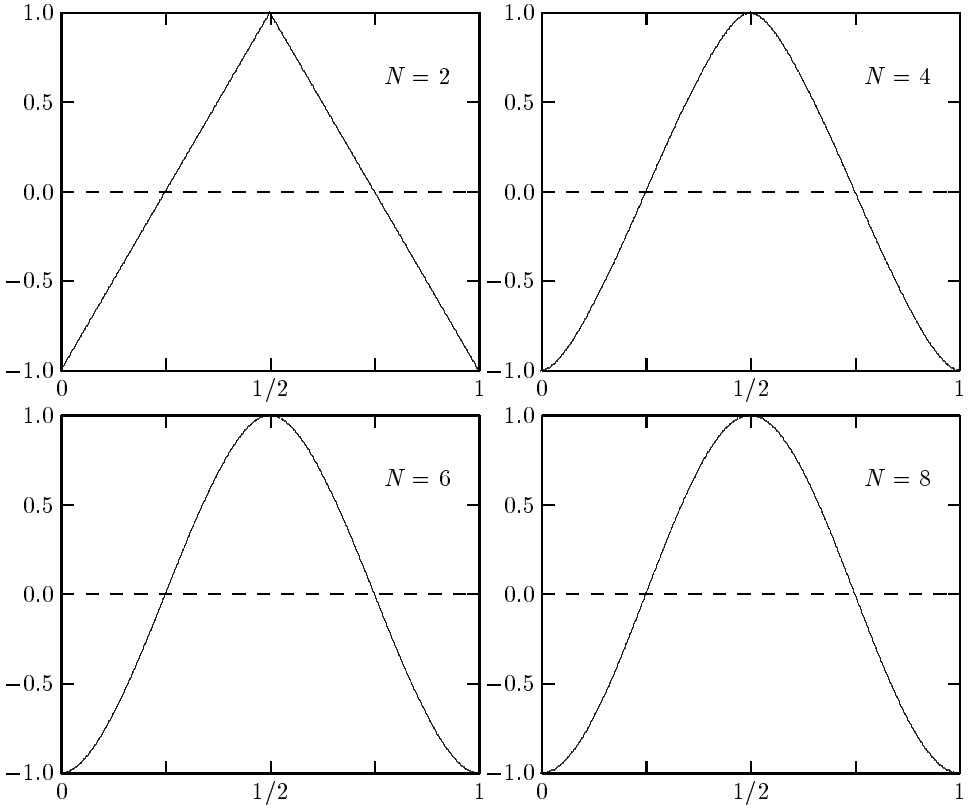

Figure 4.4: σ_0 for Deslauriers-Dubuc with $N = 2, 4, 6, 8$.

Monowavelets built from Deslauriers-Dubuc wavelets are shown in Figures 4.4 and 4.5. The properties described in Section 4.3 can now easily be verified graphically.

The Meyer wavelet [140] has $\operatorname{supp} \widehat{\psi}(\omega) = [\Leftrightarrow 8\pi/3, \Leftrightarrow 2\pi/3] \cup [2\pi/3, 8\pi/3]$, so $\widehat{\psi}(2k\pi)$ is identically zero for $|k| \neq 1$. We have $\widehat{\psi}(\pm 2\pi) = \Leftrightarrow \sqrt{2}/2$, and thus

$$\sigma_{0,\mathrm{Meyer}}(x) \;=\; \Leftrightarrow \sqrt{2}\,\cos(2\pi x).$$

The Meyer wavelet has faster than polynomial decay and thus all the functions σ_p are defined. They are all of the form

$$\sigma_{p,\mathrm{Meyer}}(x) \;=\; a_p\,\sin(2\pi x \Leftrightarrow b_p).$$

The asymptotical error expansion, however, does not make sense in this case, since all the moments of this wavelet vanish ($N = \infty$).

For the Shannon wavelet, where

$$\psi_{\mathrm{Shannon}}(x) \;=\; \frac{\sin(2\pi x) \Leftrightarrow \sin(\pi x)}{\pi x},$$

one can see that

$$\sum_l \psi_{\mathrm{Shannon}}(x \Leftrightarrow l)$$

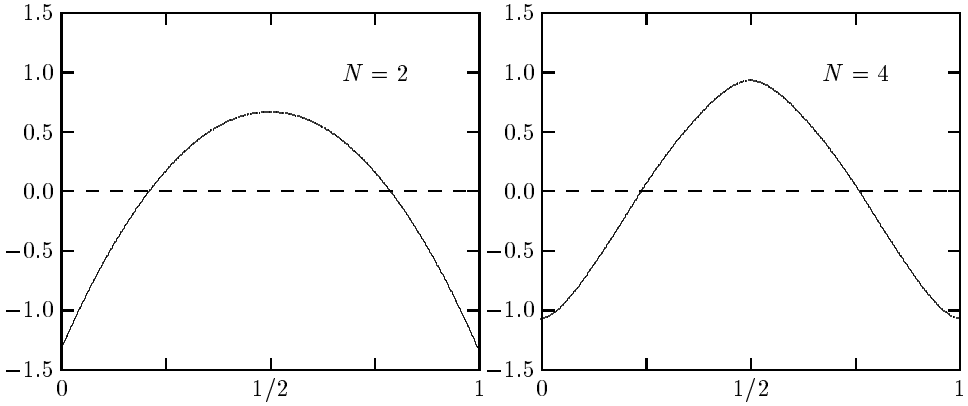

Figure 4.5: τ_0 for Deslauriers-Dubuc with $N = 2, 4$.

diverges. Note that the Shannon wavelet is not in $\mathcal{D}^0$.

For the Meyer wavelet, which has infinitely many vanishing moments, the σ_p function is a sine. For the different Daubechies families, the monowavelets seem to converge to a sine function as N goes to infinity. So it seems that as the number of vanishing moments goes to infinity, the monowavelets converge to sine functions. This makes sense because the smoother ψ, the faster the decay of $\widehat{\psi}$ and the more σ_0 looks like its fundamental frequency component. There is, however, no general proof of this phenomenon to our knowledge.

4.5 Spline monowavelets

In this section we study monowavelets in case the scaling function and wavelet are spline functions. Therefore, we first introduce Euler and Bernoulli polynomials and splines.

4.5.1 Euler and Bernoulli polynomials and splines

A sequence of polynomials V_m, $m \in \mathbf{N}$, is an *Appell sequence* if V_m is a polynomial of strict degree m and if

$$V_m' \;=\; m\, V_{m-1}.$$

The Euler and Bernoulli polynomials are two Appell sequences, denoted, respectively, with $E_m(x)$ and $B_m(x)$, which satisfy [1, 83]:

$$\frac{2\, e^{xz}}{e^z + 1} \;=\; \sum_n \frac{E_m(x)}{m!} z^m \quad \text{for} \quad |z| < 2\pi,$$

and

$$\frac{z\, e^{xz}}{e^z \Leftrightarrow 1} \;=\; \sum_n \frac{B_m(x)}{m!} z^m \quad \text{for} \quad |z| < \pi.$$

The first elements of the sequences are

$$\begin{aligned} E_0(x) &= 1 \\ E_1(x) &= x \Leftrightarrow 1/2 \\ E_2(x) &= x^2 \Leftrightarrow x \\ E_3(x) &= x^3 \Leftrightarrow 3/2\, x^2 + 1/4 \\ E_4(x) &= x^4 \Leftrightarrow 2\, x^3 + x, \end{aligned} \qquad \text{and} \qquad \begin{aligned} B_0(x) &= 1 \\ B_1(x) &= x \Leftrightarrow 1/2 \\ B_2(x) &= x^2 \Leftrightarrow x + 1/6 \\ B_3(x) &= x^3 \Leftrightarrow 3/2\, x^2 + 1/2\, x. \\ B_4(x) &= x^4 \Leftrightarrow 2\, x^3 + x^2 \Leftrightarrow 1/30. \end{aligned}$$

The Bernoulli polynomials satisfy

$$B_m(x+1) \Leftrightarrow B_m(x) \;=\; m\, x^{m-1},$$

and consequently

$$B_m^{(p)}(0) \;=\; B_m^{(p)}(1) \quad \text{for} \quad 0 \leqslant p \leqslant m \Leftrightarrow 2.$$

This means that, if we define a one-periodic function $\mathbf{B}_m$ as

$$\mathbf{B}_m(x) \;=\; B_m(x \Leftrightarrow \lfloor x \rfloor),$$

then this is a periodic spline of order $m+1$ with integer knots that belongs to $\mathcal{C}^{m-2}$. It is called the *Bernoulli periodic spline* [120]. The first four are shown in Figure 4.6. Its Fourier series for $m \geqslant 1$ is

$$\mathbf{B}_m(x) \;=\; \Leftrightarrow \frac{m!}{(2\pi i)^m} \sum_k{}' \frac{e_k(x)}{k^m}.$$

The prime indicates that the term with $k = 0$ is omitted. The function $\mathbf{B}_m$ has the same parity (even/odd) as m.

Something similar is possible for the Euler polynomials. They satisfy

$$E_m(x+1) + E_m(x) \;=\; 2\, x^m,$$

and consequently

$$E_m^{(p)}(0) \;=\; \Leftrightarrow E_m^{(p)}(1) \quad \text{for} \quad 0 \leqslant p \leqslant m \Leftrightarrow 1.$$

This means that if we define a two-periodic function $\mathbf{E}_m$ as

$$\mathbf{E}_m(x) \;=\; (\Leftrightarrow 1)^{\lfloor x \rfloor}\, E_m(x \Leftrightarrow [x]),$$

then this is a periodic spline of order $m+1$ with integer knots that belongs to $\mathcal{C}^{m-1}$. It is called the *Euler periodic spline* [120, 159]. The first four are shown in Figure 4.7. The function $\mathbf{E}_m$ has the opposite parity (even/odd) as m. Also,

$$\mathbf{E}_m(j + m/2) \;=\; (\Leftrightarrow 1)^j\, \epsilon_m.$$

It is sometimes normalized so that $\epsilon_m = 1$, see [27, 159]. It is then the cardinal spline interpolant of the sequence $y_k = (\Leftrightarrow 1)^k$. Its Fourier series for $m > 1$ is

$$\mathbf{E}_m(x) \;=\; \frac{2m!}{(\pi i)^{m+1}} \sum_k \frac{e_{2k+1}(x/2)}{(2k+1)^{m+1}}.$$

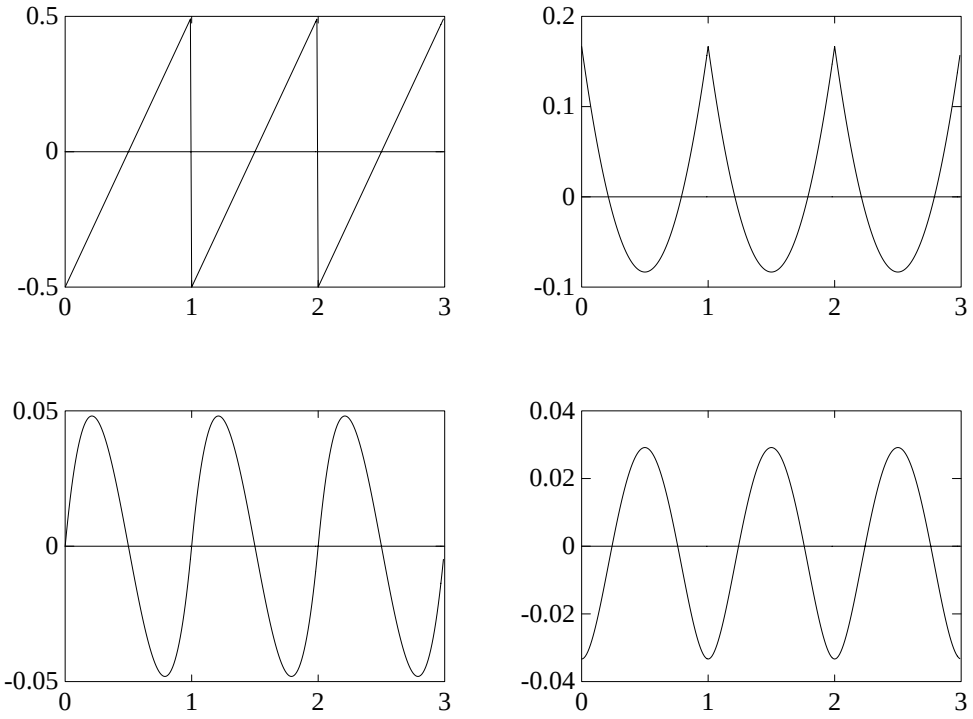

Figure 4.6: $\mathbf{B}_m$ for $m = 1, 2, 3, 4$.

4.5.2 Spline wavelets

We consider the multiresolution analysis where V_0 is the space of piecewise polynomials of degree $m \Leftrightarrow 1$ with integer knots that belong to $\mathcal{C}^{m-2}$. Note that this implies that $N = m$. From the dual multiresolution analysis we only require that $\widetilde{N} > 0$. The dual scaling function and wavelet need not be splines. We supply the notation with an extra superscript (m) (not to be confused with the derivative of a function). A possible choice of scaling function is $\varphi^{(m)} = N_m$, the cardinal B–spline of order m. We know that

$$\widehat{N}_m(\omega) \;=\; \left(\frac{1 \Leftrightarrow e^{-i\omega}}{i\omega}\right)^m,$$

so that (from (4.15) and (4.16))

$$s^{(m)}_{0,2k+1} \;=\; \left(\frac{2}{i\pi}\right)^m \frac{1}{(2k+1)^m} \quad \text{and} \quad t^{(m)}_{0,k} \;=\; \widetilde{\mathcal{N}}^{(m)}_m \left(\frac{2}{i\pi}\right)^m \frac{1}{k^m}.$$

We will show in Section 4.8 that, for all possible dual wavelets,

$$\widetilde{\mathcal{N}}^{(m)}_m \;=\; \Leftrightarrow\frac{m!}{2^{2m}}.$$

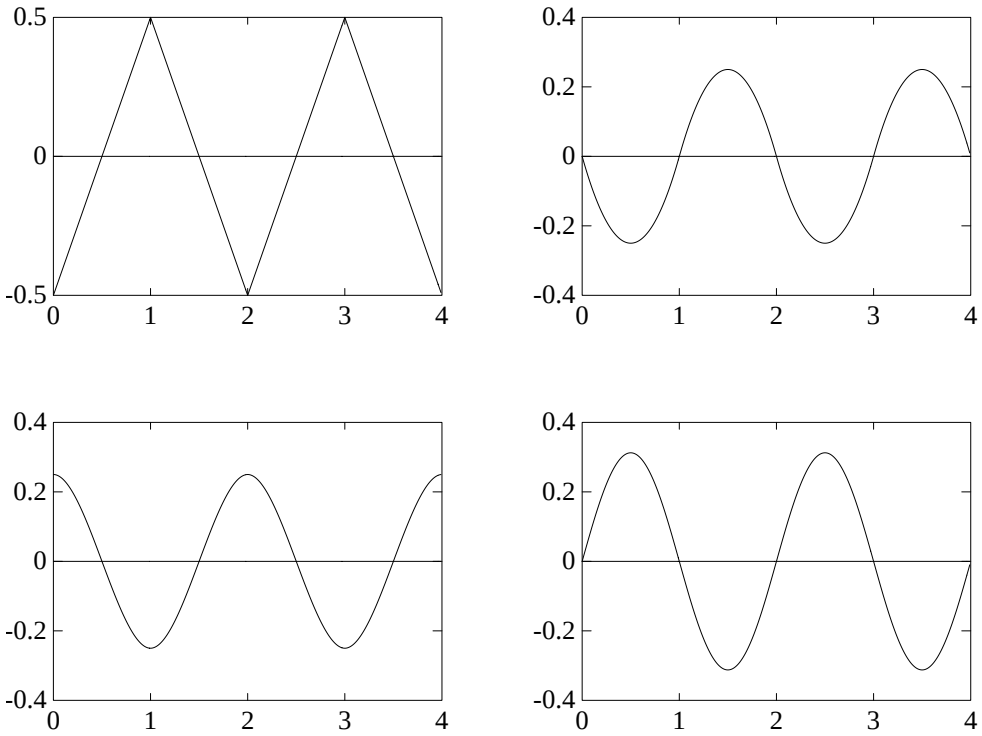

Figure 4.7: $\mathbf{E}_m$ for $m = 1, 2, 3, 4$.

This implies that

$$\sigma_0^{(m)}(x) \;=\; \frac{2^{m-1}}{(m \Leftrightarrow 1)!}\,\mathbf{E}_{m-1}(2x) \quad \text{and} \quad \tau_0^{(m)}(x) \;=\; \mathbf{B}_m(x).$$

The $\sigma_p(x)$ and $\tau_p(x)$ functions with $p > 0$ here are also one-periodic splines of order m. Analytic expressions can be obtained, but the derivation is quite technical and is omitted here. We refer to [171] for details. We only include the main result, which states that

$$\tau_p^{(m)}(x) \;=\; (\Leftrightarrow 1)^p \binom{m}{p} \mathbf{B}_{m+p}(x) \quad \text{for} \quad p < \widetilde{N}.$$

The fact that we have an analytic expression for the monowavelets here is extremely useful in convergence acceleration algorithms as described in Section 4.7.

Remarks

- Equation (4.6) for $p = 0$ yields the following relationship between Euler and Bernoulli splines:

$$\mathbf{B}_m(x) \;=\; \Leftrightarrow \frac{m}{2^{m+1}} \sum_{i=0}^{\infty} \frac{\mathbf{E}_{m-1}(2^{i+1}x)}{2^{mi}}.$$

Something similar is also true for the polynomials, as

$$B_m(x) \;=\; \Leftrightarrow\frac{m}{2^{m+1}} \sum_{i=0}^{k} \frac{E_{m-1}(2^{i+1}x)}{2^{mi}} + \frac{B_m(2^{k+1}x)}{2^{m(k+1)}},$$

and this is just an iterated version of [1, equation (23.1.27)].

- In [167, pages 147–151], Strang and Fix construct an asymptotical error analysis for the projection in the space spanned by piecewise linear finite elements. This result coincides with the one presented here in case $m = 2$ and $M = 2$.

4.6 Interpolation

The leading term of the expansion (4.4) looks like

$$h^N \, \tau_0(x/h) \, f^{(N)}(x)/N!.$$

Remember that we stated that it consists of an oscillating and a modulating part. The modulating part is given by the envelopes

$$h^N \, f^{(N)}(x) \, \tau_{\max} \quad \text{and} \quad h^N \, f^{(N)}(x) \, \tau_{\min}, \tag{4.18}$$

where

$$\tau_{\max} \;=\; \max_{x\in[0,1]} \tau_0(x)/N! \quad \text{and} \quad \tau_{\min} \;=\; \min_{x\in[0,1]} \tau_0(x)/N!.$$

The first term oscillates between these two envelopes. As a result of Lemma 4.2 this function has at least 2^{n+1} zeros per unit length. This leads to the following theorem:

Theorem 4.3 *If f is sufficiently smooth, the approximation $\mathcal{P}_n f$ asymptotically interpolates f in at least 2^{n+1} points per unit length.*

The "asymptotically" here means that one can always find a large enough n so that the interpolating properties hold. Essentially, one needs to take n so that the remaining terms do not influence the zeros of the first term. The examples of Section 4.10 will show that n need not be very large. Note that the number of interpolation points is twice the number of basis functions. The interpolation points z_k, where $\mathcal{P}_n f(z_k) = f(z_k)$, satisfy

$$z_{2k} \;=\; (x_1 + k)\, h + \mathcal{O}(h^2) \quad \text{and} \quad z_{2k+1} \;=\; (x_2 + k)\, h + \mathcal{O}(h^2),$$

where x_1 and x_2 are zeros of $\tau_0(x)$ in $[0, 1)$.

4.7 Numerical extrapolation

One of the most powerful applications of error expansions are extrapolation algorithms, cf. [186]. The general idea is to use approximations at different resolutions to identify, estimate, and eliminate the most relevant components of the expansion and thus obtain convergence acceleration. The easiest case is if the expansion consists of powers of the scale parameter h. Then, classic methods, such as Richardson extrapolation, can be used.

The multiresolution scheme naturally gives us a number of approximations at different levels, which can be used for convergence acceleration. We assume that x is a point so that $2^m x \in \mathbf{Z}$, where m is the coarsest level of the multiresolution scheme. Because of the periodicity of the monowavelets, the asymptotic error expansion at x then consists of powers of h.

Table 4.1 shows the results of a numerical experiment where

$$f(x) = \exp(\Leftrightarrow 20\,(x \Leftrightarrow 0.5)^2),$$

and the wavelet used is the Daubechies wavelet with $N = 2$. The first column shows the relative error of the approximation at $x = 1/4$ for levels n from 2 to 9. The other columns are the relative errors of the values obtained with the following Richardson extrapolation scheme:

$$r_{n,1} \;=\; \mathcal{P}_n f(1/4) \quad \text{for} \quad 2 \leqslant n \leqslant 9,$$

and

$$r_{n,j} \;=\; \frac{2^{N+j-2}\, r_{n,j-1} \Leftrightarrow r_{n-1,j-1}}{2^{N+j-2} \Leftrightarrow 1} \quad \text{for} \quad 3 \leqslant n \leqslant 9 \quad \text{and} \quad n \leqslant j \leqslant 9.$$

The first column shows a convergence of h^2, i.e. on each level the error is roughly divided by 4. Subsequent columns correspond to eliminating one term of the expansion. We see that in this case simple linear combinations result in an increase from 5 to almost 13 accurate digits.

This simple algorithm, however, only works for the points with $2^m x \in \mathbf{Z}$. In case this condition is not satisfied it cannot be used any more. It is then necessary to numerically evaluate the monowavelets so that the coefficients $\tau_p(x/h)$ can be taken into account in the algorithm. In this case an analytical expression for the monowavelets, such as the one derived for the spline case, is extremely useful.

In a more general setting, it is also possible to consider f as a distribution. Note that a distribution f is characterized by inner products $\langle\, f, \eta \,\rangle$ where η belongs to the class of test functions. If f is a distribution, we can approximate the inner products $\langle\, f, \eta \,\rangle$ by $\langle\, \mathcal{P}_n f, \eta \,\rangle$. We can use the asymptotic error expansion of the wavelet approximation to get an asymptotic error expansion for these inner products. For a certain class of test functions this error expansion consists

Table 4.1: The Richardson extrapolation table for x = 1/4.

n	$\lvert\frac{\mathcal{E}_n f(1/4)}{f(1/4)}\rvert$							
2	1.2e-01	-	-	-	-	-	-	-
3	5.6e-02	3.3e-02	-	-	-	-	-	-
4	1.7e-02	3.5e-03	6.8e-04	-	-	-	-	-
5	4.3e-03	1.9e-04	2.8e-04	2.5e-04	-	-	-	-
6	1.1e-03	2.5e-06	3.0e-05	1.3e-05	5.5e-06	-	-	-
7	2.7e-04	2.3e-06	2.3e-06	4.3e-07	1.4e-08	7.3e-08	-	-
8	6.6e-05	4.2e-07	1.5e-07	1.2e-08	1.3e-09	1.6e-09	9.9e-10	-
9	1.6e-05	6.2e-08	9.9e-09	3.4e-10	3.6e-11	1.6e-11	3.6e-12	3.1e-13

of powers of h whose coefficients are independent of h. This makes Richardson extrapolation applicable again.

An important implication of this algorithm is the following: Traditionally, when using wavelet approximation, the only way to achieve $\mathcal{O}(h^N)$ convergence is to use a dual wavelet with N vanishing moments. This can have disadvantages such as long filters or extra problems when working on an interval. Now, it is possible to use wavelets with a low number of vanishing moments (such as Haar), but still get the fast $\mathcal{O}(h^N)$ convergence using the extrapolation algorithm. This is especially useful in applications where one naturally works with boxcar coefficients, cf. the discussion in Section 3.10.

4.8 Comparison of multiresolution analyses

The error expansion can be used to compare different multiresolution analyses. The error decays as $\mathcal{O}(h^N)$, and the constant in front of this factor is given by

$$U_0(x) \;=\; \tau_0(x/h)\,\frac{f^{(N)}(x)}{N!}$$

For sufficiently small h the leading term of the expansion provides a sufficiently accurate approximation of the error. To compare different multiresolution analyses we therefore just look at

$$A_N \;=\; \|\tau_0\|_\infty,$$

because

$$\|\mathcal{E}_n\, f\|_p \approx h^N\, \|U_0\|_p \;\leqslant\; \frac{A_N\, h^N}{N!}\, |f|_{N,p}.$$

Here the Sobolev seminorm is defined as

$$|f|_{N,p} \;=\; \|f^{(N)}\|_p.$$

Table 4.2: A_N for different wavelet families.

N	extremal-phase	closest-to-linear-phase	coiflet	spline	Deslauriers-Dubuc
1	0.500	0.500		0.5000	
2	0.500	0.500	0.641	0.1667	0.1667
3	0.597	0.597		0.0481	
4	0.865	0.915	0.856	0.0333	0.3000
5	1.904	1.918		0.0244	
6	5.109	5.701	4.899	0.0238	1.7857
7	18.169	18.044		0.0261	
8	70.927	71.865	59.436	0.0333	21.6176
9	310.398	303.921		0.0476	

4.8.1 Different multiresolution analyses

A first possibility is to compare different multiresolution analyses that have the same N. The order of convergence of the wavelet approximation evidently is the same. Therefore, we compare the numerical value of the constant A_N. Table 4.2 gives A_N as a function of N for the Daubechies orthogonal wavelets, spline wavelets, and Deslauriers-Dubuc wavelets. As we will see in the next section, A_m is the same for all spline wavelets (orthogonal, biorthogonal or semiorthogonal) of order m. The spline wavelets have by far the smallest constants.

The ratio between A_N for Daubechies' orthogonal wavelets and spline wavelets behaves roughly like 2^N. Consequently, an approximation using splines at a certain level n is yields roughly the same error as an approximation using Daubechies' orthogonal scaling functions at level $n + 1$. In other words, in order to obtain an approximation with a certain numerical error, one needs, in general, one more level with Daubechies' orthogonal wavelets than with spline wavelets. Remember that one extra level involves double the amount of work.
Note: The fact that the first non-vanishing dual wavelet moment plays a role in comparing errors in the discrete case was also pointed out in [116].

4.8.2 Fixed multiresolution analysis

Another idea is to compare A_N for a multiresolution analysis with fixed V_j subspaces, but different W_j subspaces and consequently different projection operators. Typically, we want to compare biorthogonal bases with (semi-)orthogonal ones. Recall that in the latter case the projection operators are orthogonal and yield best approximations in the L_2 sense.

In order to compare the error expansion for different families of wavelets, we

first need to study some dependencies in a multiresolution analysis more carefully. Here we always use the normalizations

$$\widehat{\varphi}(0) \;=\; 1 \quad \text{and} \quad g(\pi) \;=\; 1,$$

to avoid non-uniqueness. The following dependencies now hold in any multiresolution analysis:

- Given a scaling function, the subspaces V_j are uniquely determined by

  $$V_j \;=\; \text{clos span}\{\varphi_{j,l} \mid l \in \mathbf{Z}\}.$$

  On the other hand, given the subspaces V_j, infinitely many scaling functions exist that generate these spaces. More precisely, if φ is a such a function, any function φ^* with

  $$\widehat{\varphi}^*(\omega) \;=\; a(\omega)\,\widehat{\varphi}(\omega), \tag{4.19}$$

  where $a(\omega)$ is a bounded 2π-periodic function that does not vanish and $a(0) = 1$, generates the same subspaces V_j. Moreover, any function generating the same subspaces is of this form.

- A similar statement holds for the wavelet ψ and the subspaces W_j.

- Given the subspaces W_j, the subspaces V_j are uniquely determined by

  $$V_j \;=\; \bigoplus_{i=-\infty}^{j-1} W_i.$$

  On the other hand, if the V_j are given, infinitely many choices for complementary spaces W_j are possible, one choice being the orthogonal complements.

- Given the spaces W_j, the $\widetilde{W}_j$ are uniquely determined by the fact that

  $$W_j \perp \widetilde{W}_{j'} \quad \text{if} \quad j \neq j' \quad \text{and} \quad \bigoplus_j \widetilde{W}_j = \mathrm{L}_2(\mathbf{R}).$$

Figure 4.8 shows these dependencies in a graph.

For each characteristic of a multiresolution analysis we now can define its dependency: V-dependent, φ-dependent, W-dependent, or ψ-dependent. Something is called φ-dependent if it depends on the specific choice of scaling function. The other dependencies are defined similarly. We always use the most characteristic dependency. By looking at the dependency graph we see that something that is V-dependent is also φ-dependent, but we only use the term φ-dependent for something that is not V-dependent. In other words, something that is V-dependent does not change if the scaling function φ is replaced with a function φ^* that generates the same V_j spaces. In order to become more familiar with this terminology, we illustrate it with some examples.

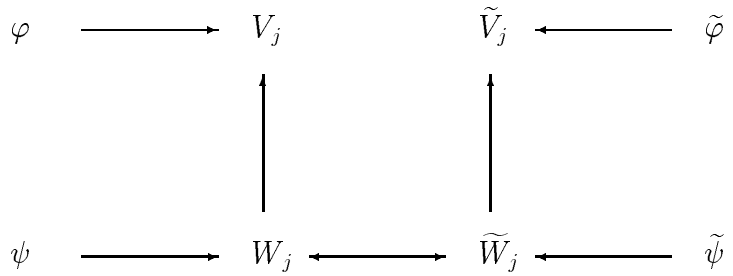

Figure 4.8: Dependencies in a multiresolution analysis.

- Anything that is V-dependent is also W-dependent.
- The projection $\mathcal{Q}_n f$ is W-dependent.
- The projection $\mathcal{P}_n f$ is W-dependent too, as

$$\mathcal{P}_n \;=\; \sum_{j=-\infty}^{n-1} \mathcal{Q}_j.$$

- The number of vanishing dual wavelet moments (N) is V-dependent.
- The σ_0 monowavelet is φ-dependent because of (4.17).
- The σ_p^* and τ_p monowavelets are, in general, W-dependent because of (4.8) and (4.9).
- Anything that is φ-dependent and W-dependent at the same time is V-dependent.

We extend these properties with the following lemmas:

Lemma 4.4 *The first non-vanishing dual wavelet moment is V-dependent and, more precisely,*

$$\widetilde{\mathcal{N}}_N \;=\; \Leftrightarrow (i/2)^N \, \overline{h^{(N)}(\pi)} \;=\; (\Leftrightarrow 1/2)^N \sum_k (\Leftrightarrow 1)^k \, k^N \, h_k.$$

Proof : We first prove that it is φ-dependent. Pick a scaling function φ that generates the V_j. Recall that (2.30)

$$\widetilde{g}(\omega) \;=\; \Leftrightarrow \frac{\overline{h(\omega+\pi)}}{\overline{\delta(\omega)}},$$

with

$$\delta(\omega) \;=\; h(\omega)\, g(\omega+\pi) \Leftrightarrow h(\omega+\pi)\, g(\omega).$$

Now,

$$\widetilde{\mathcal{N}}_N \;=\; i^N \widehat{\widetilde{\psi}}^{(N)}(0) \;=\; i^N \frac{d^N}{d\omega^N}\left[\widetilde{g}(\omega/2)\,\widehat{\widetilde{\varphi}}(\omega/2)\right]_{\omega=0},$$

and, since $\omega = 0$ is a root of multiplicity N of $\widetilde{g}(\omega)$,

$$\widetilde{\mathcal{N}}_N \;=\; (i/2)^N\,\widetilde{g}^{(N)}(0).$$

Since $\omega = \pi$ is a root of multiplicity N of $h(\omega)$, it holds that

$$\widetilde{g}^{(N)}(0) \;=\; \Leftrightarrow\frac{\overline{h^{(N)}(\pi)}}{\overline{\delta(0)}}.$$

The fact that $\delta(0) = 1$ now yields the φ-dependency.

To prove that $\widetilde{\mathcal{N}}_N$ is V-dependent, take a different scaling function,

$$\widehat{\varphi}^*(\omega) \;=\; a(\omega)\,\widehat{\varphi}(\omega),$$

which has

$$h^*(\omega) \;=\; \frac{a(2\omega)}{a(\omega)}\,h(\omega).$$

As $a(0) \;=\; 1$, and π is a zero of order N of $h(\omega)$, it holds that

$$h^{*(N)}(\pi) \;=\; h^{(N)}(\pi)/a(\pi),$$

which yields the same first non-vanishing dual wavelet moment. □

Lemma 4.5 *The leading term of the error expansion is V-dependent.*

Proof : The function σ_0 and $\widetilde{\mathcal{N}}_N$ are φ-dependent, so $\sigma_0^* = \widetilde{\mathcal{N}}_N\,\sigma_0$ is φ-dependent too. As σ_0^* was already W-dependent, it is thus V-dependent. So also τ_0 and consequently the leading term of the expansion are V-dependent. □

Note: If the subspace V_0 is the space of the piecewise polynomials of degree $m \Leftrightarrow 1$ with integer knots that belong to $\mathcal{C}^{m-2}$, we can take $\varphi^{(m)}$ to be the B-spline N_m, so

$$h(\omega) \;=\; \left(\frac{1+e^{-i\omega}}{2}\right)^m,$$

and thus

$$\widetilde{\mathcal{N}}_m^{(m)} \;=\; \Leftrightarrow\frac{m!}{2^{2m}},$$

and

$$A_m \;=\; \|\mathbf{B}_m\|_\infty.$$

In the case m is even we have a simple expression since

$$A_{2n} \;=\; \|\mathbf{B}_{2n}(x)\|_\infty \;=\; |B_{2n}|,$$

where B_{2n} is the $2n$th Bernoulli number. The leading term of the expansion is exactly the same for Battle-Lemarié orthogonal spline wavelets [19, 125], Cohen-Daubechies-Feauveau biorthogonal spline wavelets [45], and Chui-Wang semiorthogonal spline wavelets, [34, 39, 37].

The dependency of the higher order terms is studied in the following lemmas.

Lemma 4.6 *Given $\widetilde{N}$, the first $N_{\rm tot} = N+\widetilde{N}$ moments of the dual scaling function are φ-dependent.*

Proof : Since $\tilde{\varphi}$ is a dual function, it holds that

$$\sum_k \widehat{\tilde{\varphi}}(\omega + k2\pi)\overline{\widehat{\varphi}(\omega + k2\pi)} \;=\; 1.$$

Taking the pth derivative of this expression at $\omega = 0$ yields

$$\sum_k \sum_{s=0}^{p} \binom{p}{s} \widehat{\tilde{\varphi}}^{(s)}(k2\pi)\overline{\widehat{\varphi}^{(p-s)}(k2\pi)} \;=\; 0. \tag{4.20}$$

Now, since

$$i^p\, \widehat{\varphi}^{(p)}(2k\pi) \;=\; \mathcal{M}_p\, \delta_k \quad \text{for} \quad 0 \leqslant p < N,$$

and

$$i^p\, \widehat{\tilde{\varphi}}^{(p)}(2k\pi) \;=\; \widetilde{\mathcal{M}}_p\, \delta_k \quad \text{for} \quad 0 \leqslant p < \widetilde{N},$$

it holds that

$$\sum_k \widehat{\tilde{\varphi}}^{(l)}(k2\pi)\overline{\widehat{\varphi}^{(m)}(k2\pi)} \;=\; \widehat{\tilde{\varphi}}^{(l)}(0)\overline{\widehat{\varphi}^{(m)}(0)}$$

for $0 \leqslant l < \widetilde{N}$ or $0 \leqslant m < N$. The terms for $k \neq 0$ in (4.20) thus vanish if $s < \widetilde{N}$ or $p \Leftrightarrow s < N$. Consequently, if $p < N_{\rm tot}$, then

$$\sum_{s=0}^{p} \binom{p}{s} (\Leftrightarrow 1)^s\, \widetilde{\mathcal{M}}_s\, \mathcal{M}_{p-s} \;=\; 0 \quad \text{for} \quad 0 < p < N_{\rm tot}.$$

These relations show that the first $N_{\rm tot}$ moments of the dual scaling function only depend on the first $N_{\rm tot}$ moments of the scaling function. □

Note: a similar statement holds also for the first $N_{\rm tot}$ discrete moments of the sequence $\{h_k\}$.

Lemma 4.7 *The functions $\mathcal{Q}_n\, x^p$ and $\mathcal{E}_n\, x^p$ are V-dependent if $p < N_{\rm tot}$.*

Proof : It follows from Lemma 4.6 that the function $\mathcal{P}_n\, x^p$ is V-dependent. The proof then immediately follows from the fact that $\mathcal{E}_n = 1 \Leftrightarrow \mathcal{P}_n$ and $\mathcal{Q}_n = \mathcal{P}_{n+1} \Leftrightarrow \mathcal{P}_n$. □

Lemma 4.8 *The functions $\sigma_p^*(x)$ and $\tau_p(x)$ are V-dependent if $p < \widetilde{N}$.*

Proof : From Lemma 4.7, (4.8), and (4.9). □

These lemmas can be combined into the following theorem:

Theorem 4.9 *The first $\widetilde{N}$ terms of the error expansion are V-dependent.*

So we can conclude that for the approximation of a smooth function on a small scale it does not really matter how the spaces W_j are chosen. The outcome that orthogonal and biorthogonal projections almost give the same result might look surprising at first sight, but one has to keep in mind that it only holds for smooth functions.

4.9 Quadrature formulae and error expansions

The coefficients of the basis functions in the projection $\mathcal{P}_n f$ are inner products with the dual functions. As we saw in Chapter 3, these inner products

$$\lambda_{n,l} \;=\; \langle\, f, \widetilde{\varphi}_{n,l} \,\rangle \;=\; 2^{-n/2}\, \langle\, f(2^{-n}(y+l)), \widetilde{\varphi}(y) \,\rangle\,,$$

usually cannot be calculated exactly, but are approximated by a quadrature formula. We define for $\widetilde{\varphi} \in \mathcal{D}^p$,

$$f_p \;=\; F_r[y^p] \;=\; \widetilde{\mathcal{M}}_p \Leftrightarrow \sum_{k=1}^{r} w_k\, x_k^p,$$

and remember that

$$f_p \;=\; 0 \quad \text{for} \quad 0 \leqslant p \leqslant q, \tag{4.21}$$

where q is the degree of accuracy. The definition of F_r is given in Section 3.5. Recall that the projection operator $\mathcal{P}_n$ is defined as

$$\mathcal{P}_n f(x) \;=\; \sum_l \langle\, f(h(y+l)), \widetilde{\varphi}(y) \,\rangle\; \varphi(2^n x \Leftrightarrow l).$$

The use of a quadrature formula results in an approximation of this operator given by

$$\mathcal{P}'_n f(x) \;=\; \sum_l Q_r[f(h(y+l))]\, \varphi(2^n x \Leftrightarrow l).$$

The error of this "approximation of the approximation" is now,

$$\mathcal{E}'_n f \;=\; f \Leftrightarrow \mathcal{P}'_n f \;=\; \mathcal{E}_n f + \mathcal{P}_n f \Leftrightarrow \mathcal{P}'_n f.$$

We try to understand how the quadrature formula affects the asymptotic error expansion. In other words, we derive an expansion for $\mathcal{P}_n \Leftrightarrow \mathcal{P}'_n$. We assume that

φ, $\widetilde{\varphi} \in \mathcal{D}^{M+1}$, $f \in \mathcal{C}^{M+1}$, and $f^{(p)}$ bounded for $p \leqslant M+1$ and write the first M terms of a Taylor series around x ($M > q$).

$$\begin{aligned}
\mathcal{P}_n f(x) \Leftrightarrow \mathcal{P}'_n f(x) &= \sum_l F_r[f(h(y+l))]\,\varphi(2^n x \Leftrightarrow l) \\
&= \sum_l F_r\left[\sum_{p=0}^{M} \frac{f^{(p)}(x)}{p!}\,(hy+hl \Leftrightarrow x)^p + \right. \\
&\qquad \left. \frac{f^{(M+1)}(\xi(x,y))}{(M+1)!}\,(hy+hl \Leftrightarrow x)^{M+1}\right] \cdot \varphi(2^n x \Leftrightarrow l) \\
&= \sum_l \sum_{p=0}^{M} \frac{h^p f^{(p)}(x)}{p!}\,F_r\left[(y+l \Leftrightarrow 2^n x)^p\right]\,\varphi(2^n x \Leftrightarrow l) \\
&\quad + h^{M+1} L(2^n x) \\
&= \sum_{p=0}^{M} \frac{h^p f^{(p)}(x)}{p!} \sum_{s=0}^{p} \binom{p}{s} f_{p-s}\,(\Leftrightarrow 1)^s\,\rho_s(2^n x) + h^{M+1} L(2^n x)
\end{aligned}$$

with

$$\rho_p(x) \;=\; \sum_l (x \Leftrightarrow l)^p\,\varphi(x \Leftrightarrow l),$$

and

$$L(x) \;=\; \sum_l F_r\left[\frac{f^{(M+1)}(\xi(x,y))}{(M+1)!}\,(y+l \Leftrightarrow x)^{M+1}\right]\varphi(x \Leftrightarrow l).$$

We now need to bound $L(x)$:

$$\begin{aligned}
|L(x)| &\leqslant \frac{\|f^{(M+1)}\|_\infty}{(M+1)!} \sum_l F_r\left[|y+l \Leftrightarrow x|^{M+1}\right]|\varphi(x \Leftrightarrow l)| \\
&\leqslant \frac{\|f^{(M+1)}\|_\infty}{(M+1)!} \sum_l \Big(\langle\, |y+l \Leftrightarrow x|^{M+1}, \widetilde{\varphi}(y)\,\rangle + \\
&\qquad\qquad Q_r\left[|y+l \Leftrightarrow x|^{M+1}\right]\Big)\,|\varphi(x \Leftrightarrow l)| \\
&\leqslant \frac{\|f^{(M+1)}\|_\infty}{(M+1)!}\,(I+II),
\end{aligned}$$

with

$$\begin{aligned}
I &\leqslant \sum_l \langle\, |y+l \Leftrightarrow x|^{M+1}, |\widetilde{\varphi}(y)|\,\rangle\;|\varphi(x \Leftrightarrow l)| \\
&\leqslant \sum_l \left[\sum_{j=0}^{M+1} m_j\,|x \Leftrightarrow l|^j \binom{M+1}{j}\right]|\varphi(x \Leftrightarrow l)| \\
&\qquad \text{with } m_j \;=\; \langle\, |z|^{M+1-j}, |\widetilde{\varphi}(z)|\,\rangle \;\text{ (finite since } \widetilde{\varphi} \in \mathcal{D}^{M+1})
\end{aligned}$$

$$\begin{aligned}
&\leqslant \max_{0\leqslant j\leqslant M+1} m_j \sum_l \left[\sum_{j=0}^{M+1} |x \Leftrightarrow l|^j \binom{M+1}{j}\right] |\varphi(x \Leftrightarrow l)| \\
&= \max_{0\leqslant j\leqslant M+1} m_j \sum_l (|x/h \Leftrightarrow l| + 1)^{M+1} |\varphi(x \Leftrightarrow l)| \\
&\leqslant C',
\end{aligned}$$

where C' is independent of n and x as $\varphi \in \mathcal{D}^{M+1}$, and

$$\begin{aligned}
II &\leqslant \sum_l Q_r[|y + l \Leftrightarrow x|^{M+1}] \; |\varphi(x \Leftrightarrow l)| \\
&\leqslant \sum_l \sum_{k=1}^{r} |w_k| |x_k + l \Leftrightarrow x|^{M+1} \; |\varphi(x \Leftrightarrow l)| \\
&\leqslant \max_{1\leqslant k\leqslant r} |w_k| \; (1 + |x_k|^{M+1}) \sum_l (|x \Leftrightarrow l| + 1)^{M+1} |\varphi(x \Leftrightarrow l)| \\
&\leqslant C'',
\end{aligned}$$

where C'' also is independent of n and x. The first $q+1$ terms of the expansion vanish because of (4.21) so that

$$\mathcal{P}_n f(x) \Leftrightarrow \mathcal{P}'_n f(x) = \sum_{p=q+1}^{M} \frac{h^p f^{(p)}(x)}{p!} \sum_{s=0}^{p-q-1} \binom{p}{s} f_{p-s} \, (\Leftrightarrow 1)^s \, \rho_s(2^n x) + h^{M+1} L(2^n x).$$

Recall that

$$\rho_p(x) \;=\; \mathcal{M}_p \quad \text{for} \quad 0 \leqslant p < N,$$

so if we let $M = N + q$, we have

$$\mathcal{P}_n f(x) \Leftrightarrow \mathcal{P}'_n f(x) \;=\; \sum_{p=q+1}^{N+q} \frac{h^p f^{(p)}(x)}{p!} \, \kappa_p + \mathcal{O}(h^{N+q+1})$$

with

$$\kappa_p \;=\; \sum_{s=0}^{p-q-1} \binom{p}{s} f_{p-s} \, (\Leftrightarrow 1)^s \, \mathcal{M}_s.$$

Formally, this expansion is similar to the expansion of the error of the wavelet approximation, the only difference being that instead of oscillating functions there are constants in each term. We can get an error expansion for $\mathcal{E}'_n$ by adding the expansions for $\mathcal{E}_n$ and $\mathcal{P}'_n \Leftrightarrow \mathcal{P}'_n$. In order not to ruin the convergence speed of $\mathcal{O}(h^N)$ of the wavelet approximation, we obviously have to take $q \geqslant N \Leftrightarrow 1$. At first sight is seems most logical to choose $q = N \Leftrightarrow 1$. The leading term of the combined expansion is then

$$\frac{h^N f^{(N)}(x)}{N!} \, (\tau_0(x) + f_N).$$

This immediately leads to the following theorem.

Theorem 4.10 *If*

$$\min_x |\tau_0(x) + f_N| > 0,$$

then the asymptotic interpolating properties of the wavelet approximation as described in Theorem 4.3 are lost.

In this case it is actually the quadrature formulae that contributes the most to the error. So here one might want to use a quadrature formula with degree of accuracy $q \geqslant N$.

The next theorem follows immediately from Lemma 4.6 and the construction of the quadrature formula.

Theorem 4.11 *Any quadrature formula with degree of accuracy less than $N_{\rm tot}$ is V-dependent.*

This implies the remarkable result that $\mathcal{P}'_n f$ does not depend on the dual scaling function as long as $q < N_{\rm tot}$. In other words, for fixed $\widetilde{N}$, no matter how the subspaces W_j are chosen, $\mathcal{P}'_n$ will be the same. If the wavelet subspaces W_j are chosen as the orthogonal complement, we know that $\widetilde{N} = N$. We denote the approximate projection in this case as $\mathcal{P}_n^{\rm orth\prime}$. Consequently, if the dual scaling function has $\widetilde{N} \geqslant N$ and if $q < 2N$, then the operator $\mathcal{P}'_n$ coincides with $\mathcal{P}_n^{\rm orth\prime}$.

4.10 Numerical examples

We implemented a computer program that calculates the operator $\mathcal{P}'_n f$ from the samples $f(k/2^n)$ of a function f. We again consider the function

$$f(x) \;=\; \exp(\Leftrightarrow 20(x \Leftrightarrow 1/2)^2),$$

and calculate $\mathcal{P}'_n f(x)$ for $x \in [0,1]$.

Figure 4.9 shows the original function and its approximation $\mathcal{P}'_4 f(x)$ in case the scaling function and the dual scaling function both are the box function ($N = \widetilde{N} = 1$). The quadrature formula used here has $r = 2$, $x_1 = 0$, $x_2 = 1$, $w_1 = w_2 = 1/2$, and $q = 0$.

Figure 4.10 shows $\mathcal{P}'_4 f(x)$ where the scaling function is again the box function, but the dual scaling function is the one with $\widetilde{N} = 3$ constructed in [45]. The quadrature formula used here has $r = 4$, $x_1 \ldots x_4 = \Leftrightarrow 1 \ldots 2$, $w_1 = w_4 = \Leftrightarrow 1/12$, $w_2 = w_3 = 7/12$, and $q = 3$. Note the difference with the previous figure at the top of the Gaussian.

Figure 4.11 shows the error $\mathcal{E}'_5 f(x)$ in the case of the orthogonal Daubechies scaling function with $N = 2$. The quadrature formula used here has $r = 4$, $x_1, \ldots, x_4 = 0, \ldots, 3$, and $q = 3$. One easily identifies the modulating and oscillating part. The dotted lines are the envelopes of the leading term of the error

Table 4.3: Comparison of the error for Daubechies' and spline wavelets ($N = 4$).

level	Daubechies'	spline
1	7.53e-01	3.05e-01
2	4.01e-01	5.07e-02
3	5.00e-02	7.53e-03
4	3.16e-03	3.13e-04
5	1.52e-04	1.01e-05
6	9.45e-06	4.24e-07
7	6.06e-07	2.37e-08
8	3.82e-08	1.53e-09
9	2.39e-09	9.67e-11
10	1.50e-10	6.06e-12

expansion. At this level the leading term already provides a reasonable approximation of the error. Note also that the interpolation properties as described in Theorem 4.3 hold.

Figure 4.12 shows the error $\mathcal{E}_5' f(x)$ in case the scaling function is the orthogonal Daubechies function with $N = 2$. The quadrature formula used here is the trapezoidal rule with $r = 2$, $x_1 = 1$, $x_2 = 2$, $w_k = \varphi(k)$, and $q = 1$. The situation described in Theorem 4.10 occurs here: $f_2 = 1/2$ and $\min_x \tau(x) = \Leftrightarrow 0.279$. We see that the interpolating properties are lost. The error is about twice as big as in the previous case. Again the dotted lines are the envelopes of the leading term of the combined error expansion.

Figure 4.13 shows the error $\mathcal{E}_5' f(x)$ in case the scaling function is the B-spline of order $N = 2$, and the dual scaling function is the one with $\widetilde{N} = 2$ constructed in [45]. The quadrature formula has $r = 3$, $x_1 \ldots x_3 = \Leftrightarrow 1 \ldots 1$, $w_1 = w_3 = \Leftrightarrow 1/12$, $w_2 = 14/12$, and $q = 2$. Again the dotted lines are the envelopes of the leading term of the error expansion. One can clearly distinguish the shape of the Bernoulli spline of degree 2.

Table 4.3 compares $\max_x |\mathcal{E}_n' f(x)|$ on different levels in two cases. On the first one, the scaling function is the orthogonal Daubechies scaling function with $N = 4$, and the quadrature formula has $q = 5$. The second one corresponds to the biorthogonal case with $\varphi = N_4$ ($N = 4$), and $\widetilde{N} = 6$. The quadrature formula has $r = 3$ and $q = 4$. The order of convergence is $\mathcal{O}(h^4)$ in both cases. The influence of the quadrature formulae is negligible. On the finer levels, the error is indeed divided by 16 each time. This also confirms what was predicted in Section 4.8: the approximation using splines at a certain level yields roughly the same error as an approximation using Daubechies' wavelets with the same N at the next (finer) level.

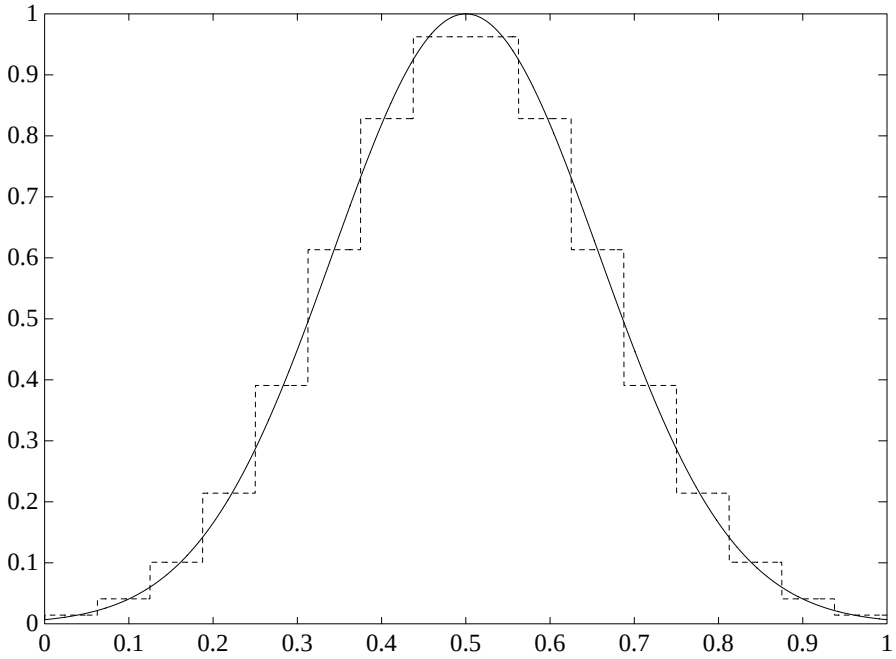

Figure 4.9: $\mathcal{P}_4' f$ in the Haar case and with $q = 0$.

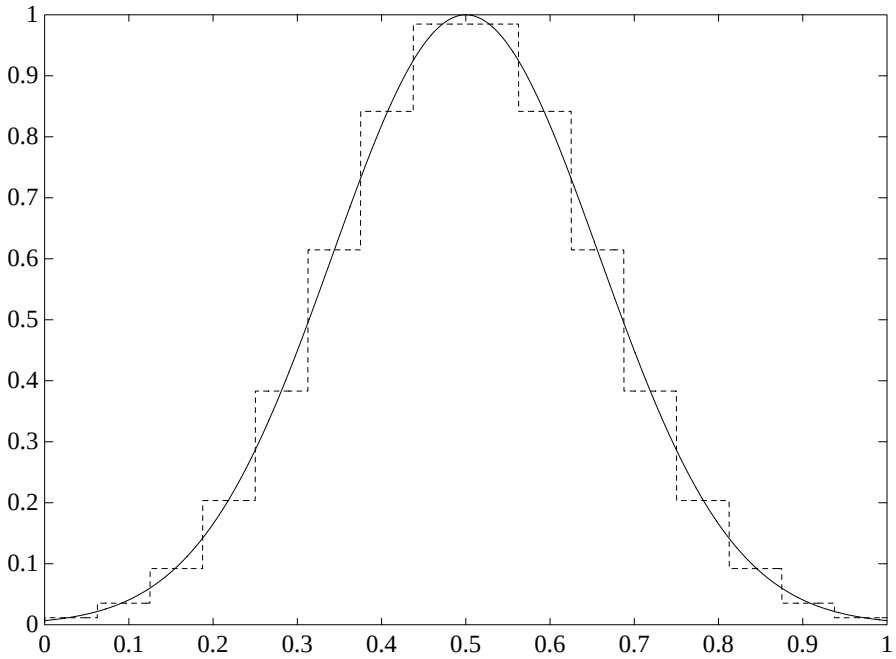

Figure 4.10: $\mathcal{P}_4' f$ in the Haar case and with $q = 3$.

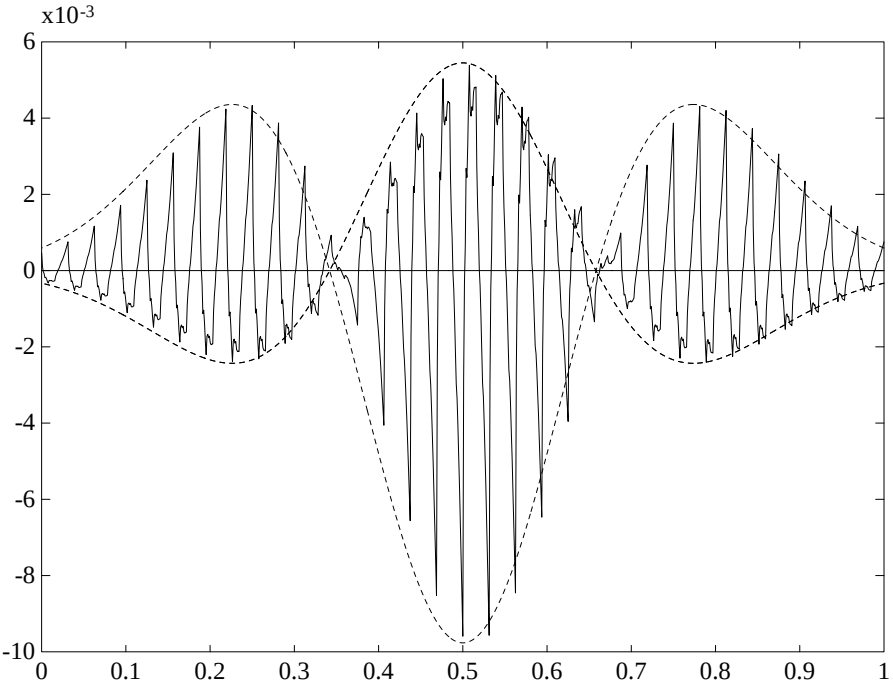

Figure 4.11: $\mathcal{E}_5'$ for the Daubechies orthogonal wavelet $(N = 2,\ q = 3)$.

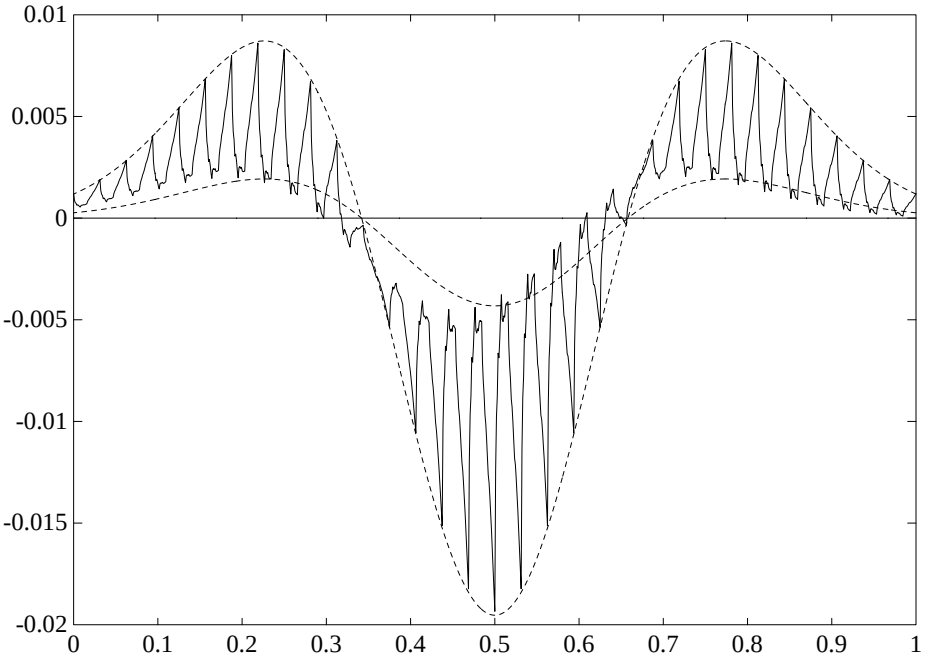

Figure 4.12: $\mathcal{E}_5'$ for the Daubechies orthogonal wavelet $(N = 2,\ q = 1)$.

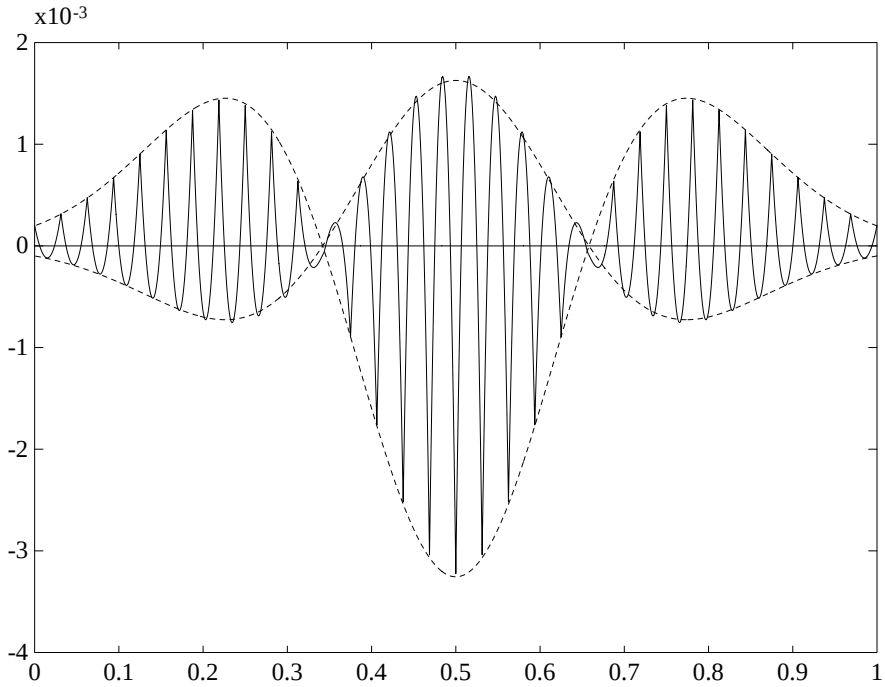

Figure 4.13: $\mathcal{E}_5'$ for the biorthogonal wavelet ($N = \widetilde{N} = q = 2$).

4.11 A more accurate first term

If we compare Figures 4.11 and 4.13, it looks like the first term is a better approximation in the spline case than in the Daubechies case. It seems that the error in the Daubechies case is a little shifted compared to the envelopes. In this section we study and explain this phenomenon. It will lead to a more accurate term. Since it occurs in the orthogonal case we assume that the wavelets are orthogonal.

4.11.1 Approximation of x^{N+1}

We first consider the case of approximating $f(x) = x^{N+1}$ at level 0. Equation (4.2) yields

$$\mathcal{Q}_0\, x^{N+1} \;=\; (N+1)\,\mathcal{N}_N\, x\,\sigma_0(x) + \mathcal{N}_{N+1}\,\sigma_0(x) \Leftrightarrow (N+1)\,\mathcal{N}_N\,\sigma_1(x).$$

The left-hand side of this formula evidently belongs to W_0. (This is not entirely correct unless we see W_0 as a subspace of $\mathrm{L}_\infty(\mathbf{R})$ instead of $\mathrm{L}_2(\mathbf{R})$.) On the right-hand side it is hard to distinguish the components of each subspace. Only the second term belongs clearly to W_0. Therefore, we try to rewrite the right-hand side and isolate the component of each term in W_0. We group the first two terms,

$$\mathcal{Q}_0\, x^{N+1} \;=\; (N+1)\,\mathcal{N}_N\,(x+\alpha)\,\sigma_0(x) \Leftrightarrow (N+1)\,\mathcal{N}_N\,\sigma_1(x),$$

with

$$\alpha \;=\; \frac{\mathcal{N}_{N+1}}{(N+1)\,\mathcal{N}_N}.$$

Secondly, we isolate the component of $\sigma_1(x)$ in W_0, by letting

$$\begin{aligned}
\mathcal{Q}_0\,\sigma_1(x) &= \sum_k \langle\, \sigma_1(y)), \psi(y \Leftrightarrow k) \,\rangle\, \psi(x \Leftrightarrow k) \\
&= \sum_k \langle\, \sigma_1(y), \psi(y) \,\rangle\, \psi(x \Leftrightarrow k) \\
&= \langle\, \sigma_1(y), \psi(y) \,\rangle\, \sigma_0(x) \\
&= \beta\,\sigma_0(x),
\end{aligned}$$

where

$$\beta \;=\; \langle\, \sigma_1, \psi \,\rangle \;=\; \int_0^1 \sigma_0(y)\,\sigma_1(y)\,dy \;=\; \int_{-\infty}^{+\infty} y\,\psi(y)\,\sigma_0(y)\,dy.$$

In the next section an algorithm to calculate β is described. Define now

$$\bar{\sigma}_1(x) \;=\; \beta\,\sigma_0(x) \Leftrightarrow \sigma_1(x),$$

so that σ_0 and $\bar{\sigma}_1$ are orthogonal. Then,

$$\mathcal{Q}_0\, x^{N+1} \;=\; (N+1)\,\mathcal{N}_N\,(x \Leftrightarrow \gamma)\,\sigma_0(x) + (N+1)\,\mathcal{N}_N\,\bar{\sigma}_1(x),$$

with

$$\gamma \;=\; \beta \Leftrightarrow \alpha.$$

In this formula the second term has no component in W_0. We see that in order to isolate the component in W_0, the modulating function (in this case x) has to be shifted over a distance γ.

4.11.2 Calculation of β

In this section an algorithm to calculate β accurately is described. We can write

$$\begin{aligned}
\beta &= \langle\, x, \sigma_0(x)\, \psi(x) \,\rangle \\
&= 2 \sum_k g_k \,\langle\, x, \sigma_0(x)\, \varphi(2x \Leftrightarrow k) \,\rangle \\
&= 1/2 \sum_k g_k \,\langle\, x, \sigma_0((x+k)/2)\, \varphi(x) \,\rangle \;+ 1/2 \sum_k g_k\, k \,\langle\, \sigma_0((x+k)/2), \varphi(x) \,\rangle \\
&= 1/2\,(p_1 + p_2),
\end{aligned}$$

with

$$p_2 \;=\; \sum_k g_k\, k \,\langle\, \sigma_0((x+k)/2), \varphi(x) \,\rangle \;=\; \sum_k h_{1-k}\, k \;\langle\, \sigma_0(x/2), \varphi(x) \,\rangle \;=\; 1 \Leftrightarrow \mathcal{M}_1,$$

and

$$p_1 \;=\; \sum_k g_k \,\langle\, x, \sigma_0((x+k)/2)\, \varphi(x) \,\rangle \;=\; \langle\, x, \sigma_0(x/2)\, \varphi(x) \,\rangle \;=\; \sum_l (\Leftrightarrow 1)^l \kappa_l,$$

where

$$\kappa_l \;=\; \langle\, x, \varphi(x \Leftrightarrow l)\, \varphi(x) \,\rangle\,.$$

The κ_l can be found by solving the linear system

$$\kappa_m = \sum_l a_{l-2m}\, \kappa_l + b_{2m},$$

with

$$a_i \;=\; \sum_k h_k\, h_{k+i} \quad \text{and} \quad b_i \;=\; \sum_k k\, h_k\, h_{k+i}.$$

4.11.3 General construction

For the general construction we first need a theorem and some lemmas.

Theorem 4.12 *If ζ is a periodic function with period one, and r is the number of vanishing moments of $\zeta\,\psi$, and $g \in \mathcal{C}^r$, then $\mathcal{Q}_n\,[g(x)\,\zeta(2^n x)] \;=\; \mathcal{O}(h^r)$ where $h = 2^{-n}$.*

Proof : We know that

$$\mathcal{Q}_n[g(x)\,\zeta(2^n x)] \;=\; \sum_l \gamma_{n,l}\,\psi(2^n x \Leftrightarrow l),$$

with

$$\gamma_{n,l} \;=\; 2^n\;\langle\, g(y), \zeta(2^n y)\,\psi(2^n y \Leftrightarrow l)\,\rangle \;\;=\;\; \langle\, g(h\,(y+l)), \zeta(y)\,\psi(y)\,\rangle\,.$$

The proof now follows from the Taylor formula. □

Lemma 4.13 *The function $\sigma_j\,\varphi$ has N vanishing moments if $0 \leqslant j < N$ and no vanishing moments if $N \leqslant j$.*

Proof :

$$\begin{aligned}\langle\, x^p, \sigma_j(x)\,\varphi(x)\,\rangle &= \sum_l \langle\, (x+l)^p, x^j\,\psi(x)\,\varphi(x+l)\,\rangle \\ &= \mathcal{M}_p\mathcal{N}_j \qquad\qquad \text{if } 0 \leqslant p < N.\end{aligned}$$

This is zero if $0 \leqslant j < N$ and non-zero if $p = 0$ and $j = N$. □

Lemma 4.14 *The following functions also have N vanishing moments if $0 \leqslant j < N$, and no vanishing moments if $N \leqslant j$: $\sigma_j(x)\,\varphi(x \Leftrightarrow l)$ with $l \in \mathbf{Z}$, $\sigma_j(2^i x)\,\varphi(x)$ with $i \in \mathbf{N}$, $\sigma_j^*\,\varphi$, and $\tau_j\,\varphi$.*

Lemma 4.15 *The function $\sigma_j(2^i x)\,\psi(x)$ with $i > 0$ has $2N$ vanishing moments if $0 \leqslant j < N$, and N vanishing moments if $N \leqslant j$.*

Proof : For $i = 1$,

$$\begin{aligned}\langle\, x^p, \psi(x)\,\sigma_j(2x)\,\rangle &= 2\sum_l g_l\;\langle\, x^p, \varphi(2x \Leftrightarrow l)\,\sigma_j(2x)\,\rangle \\ &= 2^{-p}\sum_l g_l\;\langle\, x^p, \varphi(x \Leftrightarrow l)\,\sigma_j(x)\,\rangle \\ &= 2^{-p}\sum_l g_l\;\langle\, (x+l)^p, \varphi(x)\,\sigma_j(x)\,\rangle\,.\end{aligned}$$

If $0 \leqslant p < N$, this is zero for all j because of Lemma 4.13. If $N \leqslant p < 2N$, the inner product in the summation is a polynomial of degree $p \Leftrightarrow N < N$. To prove the lemma for $i > 1$, we can follow the same reasoning as above and use Lemma 4.14. □

We assume that $f \in \mathcal{C}^{N+2}$. Then, from (4.2):

$$\mathcal{Q}_n f(x) \;=\; q_0(x)\,h^N + q_1(x)\,h^{N+1} + \mathcal{O}(h^{N+2}),$$

where

$$q_0(x) \;=\; f^{(N)}(x)\,\sigma_0^*(2^n x) \quad \text{and} \quad q_1(x) \;=\; f^{(N+1)}(x)\,\sigma_1^*(2^n x),$$

and, from (4.3):

$$\begin{aligned}
\sigma_0^*(2^n x) &= \frac{\mathcal{N}_N}{N!}\sigma_0(2^n x) \\
\sigma_1^*(2^n x) &= \frac{\mathcal{N}_{N+1}}{(N+1)!}\sigma_0(2^n x) \Leftrightarrow \frac{\mathcal{N}_N}{N!}\sigma_1(2^n x) \\
&= \frac{\mathcal{N}_N}{N!}\left[\alpha\,\sigma_0(2^n x) \Leftrightarrow \sigma_1(2^n x)\right] \\
&= \frac{\mathcal{N}_N}{N!}\left[\Leftrightarrow\gamma\,\sigma_0(2^n x) + \bar{\sigma}_1(2^n x)\right].
\end{aligned}$$

The problem here is that both $q_0(x)$ and $q_1(x)$ have a component in W_n. Indeed, since

$$\int_{-\infty}^{+\infty} \sigma_0(x)\,\psi(x)\,dx \;=\; 1,$$

Theorem 4.12 says that $\mathcal{Q}_n q_0(x) = \mathcal{O}(h^0)$ and $\mathcal{Q}_n q_1(x) = \mathcal{O}(h^0)$. We can solve this problem by grouping the components in W_n,

$$\begin{aligned}
\mathcal{Q}_n f(x) &= \frac{h^N \mathcal{N}_N}{N!}\Big[\Big(f^{(N)}(x) \Leftrightarrow h\,\gamma\,f^{(N+1)}(x)\Big)\,\sigma_0(2^n x) + h\,f^{(N+1)}(x)\,\bar{\sigma}_1(2^n x) + \\
&\qquad\qquad \mathcal{O}(h^2)\Big] \\
&= r_{0,0}(x)\,h^N + r_{0,1}(x)\,h^{N+1} + \mathcal{O}(h^{N+2}),
\end{aligned}$$

where

$$r_{0,0}(x) \;=\; \frac{\mathcal{N}_N}{N!} f^{(N)}(x \Leftrightarrow h\gamma)\,\sigma_0(2^n x) \quad \text{and} \quad r_{0,1}(x) \;=\; \frac{\mathcal{N}_N}{N!} f^{(N+1)}(x)\,\bar{\sigma}_1(2^n x).$$

In Figure 4.14, σ_1 and $\bar{\sigma}_1$ are shown for the orthogonal Daubechies wavelets (note the different scale of the vertical axis left and right). As a consequence of the orthogonalization, $\bar{\sigma}_1$ is about four times as small as σ_1, and the new first term is thus more accurate. From Theorem 4.12 with σ_0 and $\bar{\sigma}_1$ as ζ, respectively, we see that $\mathcal{Q}_n r_{0,0} = \mathcal{O}(h^0)$ and $\mathcal{Q}_n r_{0,1} = \mathcal{O}(h^1)$. This means the component in W_n of the $\mathcal{O}(h^{N+1})$ term tends to zero if $h \to 0$. Now, for $i > 0$,

$$\begin{aligned}
\mathcal{Q}_{n+i} f(x) &= \frac{h^N f^{(N)}(x)}{2^{iN}}\sigma_0^*(2^n x) + \frac{h^{N+1} f^{(N+1)}(x)}{2^{i(N+1)}}\sigma_1^*(2^n x) + \mathcal{O}(h^{N+2}) \\
&= \frac{h^N \mathcal{N}_N}{N!\,2^{iN}}\Bigg[f^{(N)}(x)\,\sigma_0(2^{n+i}x) + \\
&\qquad\qquad h\frac{f^{(N+1)}(x)}{2^i}\Big(\Leftrightarrow\gamma\,\sigma_0(2^{n+i}x) + \bar{\sigma}_1(2^{n+i}x)\Big) + \mathcal{O}(h^2)\Bigg]
\end{aligned}$$

$$
\begin{aligned}
&= \frac{h^N \mathcal{N}_N}{N!\, 2^{iN}} \Bigg[f^{(N)}(x \Leftrightarrow \gamma h)\, \sigma_0(2^{n+i}x) + \\
&\qquad\qquad h \frac{f^{(N+1)}(x)}{2^i} \left((2^i \Leftrightarrow 1)\gamma\, \sigma_0(2^{n+i}x) + \bar{\sigma}_1(2^{n+i}x) \right) + \mathcal{O}(h^2) \Bigg] \\
&= r_{i,0}(x)\, h^N + r_{i,1}(x)\, h^{N+1} + \mathcal{O}(h^{N+2}),
\end{aligned}
$$

where

$$
r_{i,0}(x) \;=\; \frac{\mathcal{N}_N}{N!\, 2^{iN}} f^{(N)}(x \Leftrightarrow \gamma h)\, \sigma_0(2^{n+i}x),
$$

and

$$
r_{i,1}(x) \;=\; \frac{\mathcal{N}_N}{N!\, 2^{i(N+1)}} f^{(N+1)}(x) \left((2^i \Leftrightarrow 1)\gamma\, \sigma_0(2^{n+i}x) + \bar{\sigma}_1(2^{n+i}x) \right).
$$

Lemma 4.15 states that $\sigma_j(2^i x)\, \psi(x)$ has $2N$ vanishing moments if $0 \leqslant j < N$. Thus, using Theorem 4.12 with $\sigma_0(2^i x)$ and $\sigma_1(2^i x)$ as $\zeta(x)$, yields, respectively, $\mathcal{Q}_n r_{i,0}(x) = \mathcal{O}(h^{2N})$ and $\mathcal{Q}_n r_{i,1}(x) = \mathcal{O}(h^{2N})$. So the only term with a component in W_n that is independent of h is $r_{0,0}(x)$. This means we have proven the following theorem:

Theorem 4.16 *Shifting the modulating function of the first term of the error expansion (4.4), and thus letting*

$$
\mathcal{E}_n f(x) \;=\; \frac{h^N \mathcal{N}_N}{N!} f^{(N)}(x \Leftrightarrow h\gamma)\, \tau(2^n x) + \mathcal{O}(h^{N+1}), \tag{4.22}
$$

yields an $\mathcal{O}(h^{N+1})$ term with a component in W_n that is $\mathcal{O}(h)$ and consequently a more accurate first term.

It is easy to see that the shift γ is zero for (anti-)symmetric wavelets. This shift can be seen as a measure for the symmetry of the wavelet. In Tables 4.4 and 4.5 we give numerical values for γ both for the original Daubechies wavelets and for the closest-to-linear-phase ones. For the original Daubechies, the absolute value of the shift seems to be increasing linearly with N. As could be expected, the shift is smaller for the closest-to-linear-phase ones.

4.11.4 Examples

In this section we consider the same example where

$$
f(x) \;=\; \exp(\Leftrightarrow 20\,(x \Leftrightarrow 0.5)^2).
$$

Figure 4.15 shows the error $\mathcal{E}_6 f(x)$. The wavelet used is the Daubechies orthogonal wavelet with $N = 3$. On the same figure, the envelopes of the first term of the right-hand side of (4.4) are drawn dashed and the envelopes of the first term of the right-hand side of (4.22) are drawn solid. Figure 4.16 shows the error $\mathcal{E}_6 f(x)$ using the

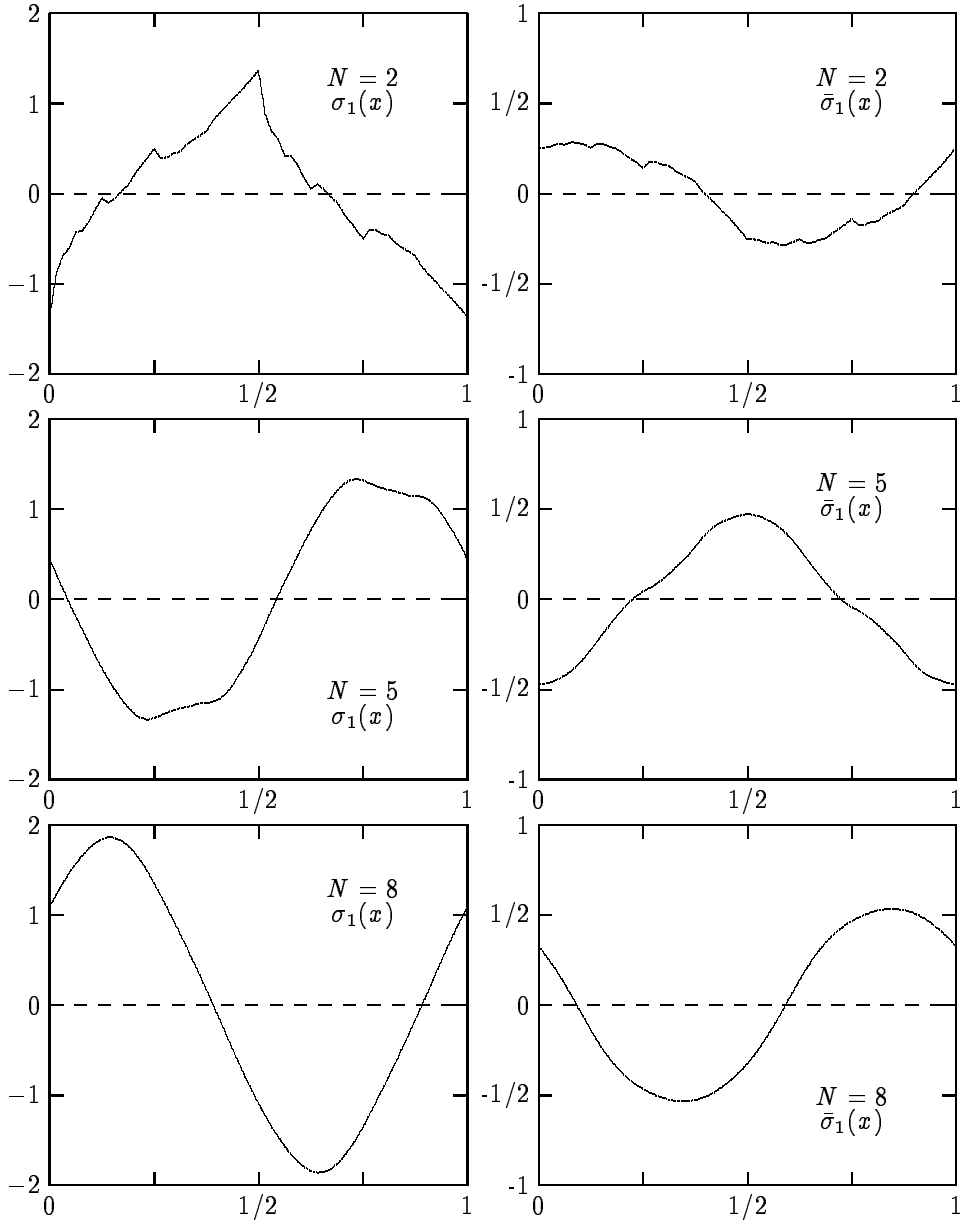

Figure 4.14: $\sigma_1(x)$ and $\bar{\sigma}_1(x)$ for Daubechies' wavelets with $N = 2, 5, 8$.

Table 4.4: The shift for the original Daubechies wavelets.

N	γ	N	γ	N	γ	N	γ
1	0	6	-1.9942	11	-3.9906	16	-6.0034
2	-0.4301	7	-2.3909	12	-4.3924	17	-6.4068
3	-0.8187	8	-2.7892	13	-4.7947	18	-6.8104
4	-1.2077	9	-3.1888	14	-5.1973	19	-7.2142
5	-1.5996	10	-3.5893	15	-5.6003	20	-7.6179

Table 4.5: The shift for the closest-to-linear-phase Daubechies wavelets.

N	γ	N	γ
2	-0.4301	6	-0.0189
3	-0.8187	7	0.0683
4	0.0860	8	-0.0741
5	0.6132	9	0.3431

Daubechies orthogonal wavelet with $N = 9$ and the same envelopes. We see that the first term of the error expansion already gives a reasonable approximation of the actual error and, secondly, that shifting the modulating function indeed yields more accurate results. For both these examples the inner products $\langle f, \varphi_{n,l} \rangle$ were calculated by the program using a quadrature formula with $q = 2N \Leftrightarrow 1$, so this influence can be neglected.

4.12 Future research

One possible direction for future research is inspired by discussions with Richard Gundy. Remember that the $\sigma_0(2^j x)$ monowavelets built from the Haar wavelet are called the Rademacher functions. The study of series of Rademacher functions is of great importance in functional analysis and probability theory [30]. A classic result here is Khintchine's inequality [123]. On the other hand, lacunary trigonometric series, e.g. series of the form

$$\sum_{k=0}^{\infty} a_k e^{i2^k x},$$

have been studied extensively [201]. Remember that we pointed out that the monowavelet series seem to converge to these lacunary trigonometric series if N tends to infinity. So monowavelets series form a natural bridge between Rade-

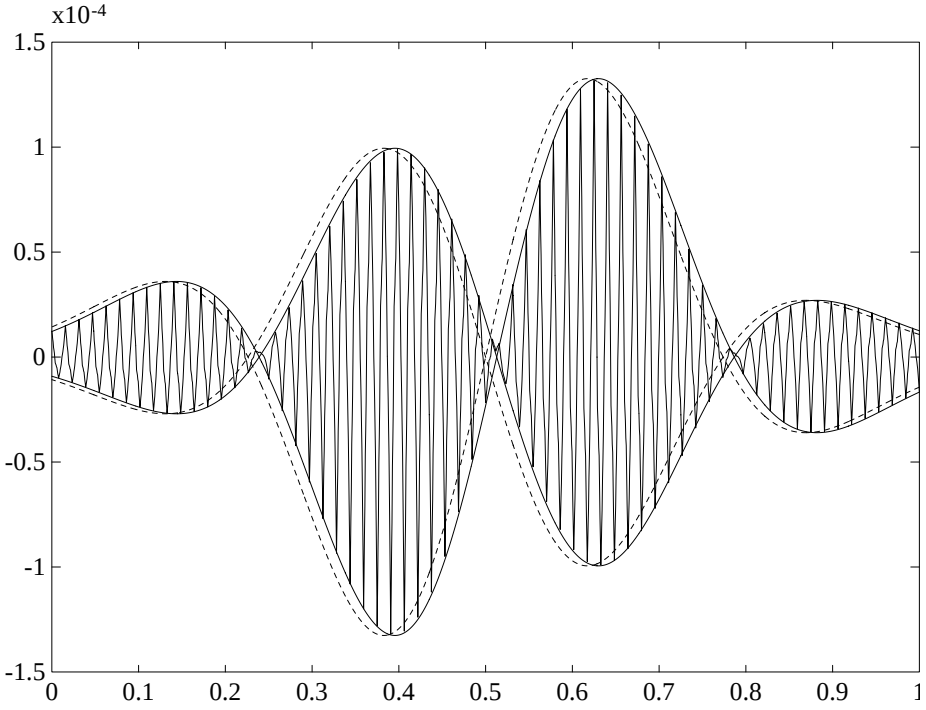

Figure 4.15: The error and envelopes for $N = 3$ and $n = 6$.

macher series and lacunary trigonometric series. It would be fascinating to study how much of the properties of Rademacher and trigonometric series carry over to monowavelet series. Another interesting question is if the monowavelets can be used to build martingales.

For the case of non-smooth functions many results concerning convergence speed have already been established. Classically, if f has Hölder regularity α with $\alpha < N$, then the wavelet approximation converges like $\mathcal{O}(h^\alpha)$. Convergence for even more general functions was proven by Mark Kon and Louise Raphael in [117]. An interesting question now, is if one can construct an error expansion for these functions and characterize the dependence on f as explicitly as in the case of smooth functions. We have done some experiments for the case where f has an isolated algebraic singularity of the form x_+^α that indicate that this might be possible.

The construction of an expansion for wavelet approximation on the interval is another direction. One can understand that most of the construction of the real line carries over. The only difference is that the monowavelets need to be adapted at the boundary.

All the expansions derived in this chapter are for the case when the error is measured as the difference between the function and the approximation. In many applications, such as image compression, the error is often measured in a smoothness norm, such as a Sobolev norm. This means that one essentially checks

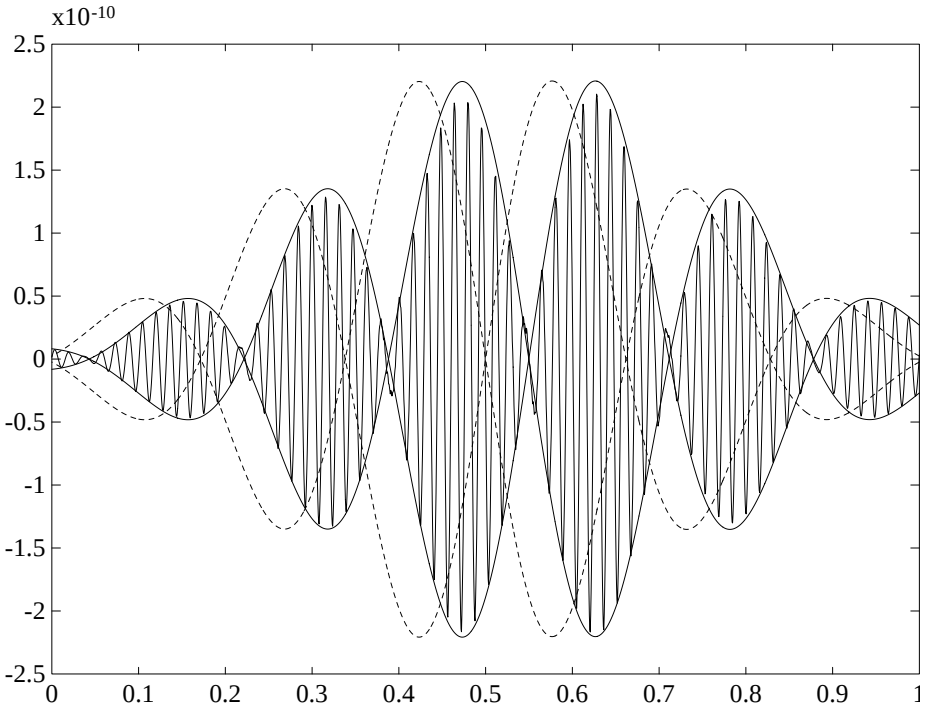

Figure 4.16: The error and envelopes for $N = 9$ and $n = 6$.

how close the derivatives of the approximation are to the derivatives of the function. Using the techniques of this chapter, it should be possible to derive pointwise error expansions for the derivatives of the error. One thing to keep in mind here, is that as the order of derivative increases, the convergence speed decreases. Typically,

$$D^p(\mathcal{E}_n f(x)) \;=\; \mathcal{O}(h^{N-p}),$$

at least if one assumes extra smoothness of the basis functions.

In Section 4.7, we described a simple extrapolation algorithm that worked in certain dyadic points. We mentioned how more general algorithms could be constructed. It would be interesting to implement these and perform some numerical experiments, especially in the case of spline wavelets where analytic expression for the monowavelets are available.

4.13 Conclusion

In this chapter we derived an asymptotical error expansion for wavelet approximation of smooth functions. It is valid for a wide class of wavelets, which includes all well-known examples. The construction lead to the definition of a new family of functions, monowavelets, whose properties were studied. These properties allowed us to understand the behavior of the error. We were able to explain the interpo-

lating properties of the wavelet approximation and showed how the expansion can be used to obtain convergence acceleration.

The leading term of the expansion was used to compare different multiresolution analyses. An important result is that it does not depend on the choice of complementary space W_j. In other words, the projection of a smooth function onto a sufficiently fine scale space of a multiresolution analysis does not essentially depend on whether this projection is orthogonal or not. We also showed that the error in the case of Daubechies' orthogonal wavelets is roughly 2^N times larger than in the case of spline wavelets. Thus, in order to obtain an approximation with a certain accuracy, one needs, in general, one more level with Daubechies' orthogonal wavelets than with spline wavelets. We also constructed an error expansion in case the quadrature formulae from Chapter 3 are used to calculate the projections.

Chapter 5

The Construction of Weighted Wavelets

"In things that are tender and unpleasing,
it is good to break the ice by some whose words are of less weight,
and to reserve the more weighty voice to come in as by chance."
—Francis Bacon, *Of Cumming.*

5.1 Introduction

In the previous chapters we studied wavelets and multiresolution analysis in detail. It is characteristic that the wavelets are translates and dilates of one function and that they are (bi)orthogonal with respect to the L_2 inner product.

In this chapter we show how wavelets can be adapted to a weighted inner product. We call such wavelets *weighted wavelets*. In some sense this parallels the transition from Legendre polynomials to Chebyshev or Jacobi polynomials. In the next chapter we will use these wavelets to construct fast algorithms for the numerical solution of differential equations.

The chapter is organized as follows. We discuss how to adapt the definition of multiresolution analysis to a weighted inner product in Section 5.2. As we will see, one can no longer use the translates and dilates of one function to form a basis. Consequently, the Fourier transform cannot be used in the construction. We therefore use an alternative, the Donoho average-interpolation scheme, a construction based on subdivision. We first consider a simple example of the subdivision scheme (Section 5.3). In Section 5.4, we discuss the construction in general. We study the convergence of the subdivision scheme and the regularity of the solution in Section 5.5, and the properties of the basis in Section 5.6. Finally, we give some examples and discuss ideas for future research.

5.2 Weighted multiresolution analysis

Consider a locally integrable function w so that $0 < w(x)$ a.e., and the weighted inner product

$$\langle f, g \rangle_w = \int_{-\infty}^{+\infty} w(x)\, f(x)\, \overline{g(x)}\, dx.$$

The associated norm is

$$\|f\|_w = \sqrt{\langle f, f \rangle_w},$$

and we define the space L_2^w as

$$\mathrm{L}_2^w = \{f \mid \|f\|_w < \infty\}.$$

Remember that the basic idea of wavelets so far was to use the dyadic translates and dilates of one function as basis functions. This can be seen as an algebraic structure since translation and dilation become algebraic operations on the Fourier transform side. We call such wavelets *algebraic wavelets*.

It is easy to see that in the case of weighted wavelets, the algebraic structure cannot be used any longer. We thus need to adapt the definition of multiresolution analysis. We first concentrate on which properties we want the weighted wavelets to have. Essentially they are the following:

1. They can be constructed from scaling functions.
2. Simple explicit expression for their coordinate functionals are available (the dual wavelets).
3. They have compact support.
4. They have cancellation properties (vanishing moments).
5. They are smooth.

We will see that properties (1)–(3) result in a fast wavelet transform. Because of (3) and (4), the wavelets are localized in space and frequency. Consequently, the wavelet coefficients decay rapidly where the function is smooth. One can thus represent a function within a certain accuracy with only a few wavelet coefficients. As we saw in Chapter 2, compact representation is the key to applications in data compression and numerical analysis. Property (5) is of importance in applications where one needs to take derivatives and when the result is judged visually, cf. the discussion in Section 2.8.

We define a multiresolution analysis as a sequence of closed subspaces V_j so that

1. $V_j \subset V_{j+1}$,
2. $\bigcup_{j=-\infty}^{+\infty} V_j$ is dense in L_2^w and $\bigcap_{j=-\infty}^{+\infty} V_j = \{\mathbf{0}\}$,

3. Scaling functions $\varphi_{j,k}$ exist so that $\{\varphi_{j,k}\}_k$ is a Riesz basis of V_j.

This implies that for every scaling function $\varphi_{j,k}$, coefficients $\{h_{j,k,l}\}$ exist so that

$$\varphi_{j,k} \;=\; \sum_l h_{j,k,l}\, \varphi_{j+1,2k+l}. \tag{5.1}$$

Each scaling function satisfies a different refinement relation. The dual multiresolution analysis consists of spaces $\widetilde{V}_j$ with bases generated by dual scaling functions $\widetilde{\varphi}_{j,k}$ that are biorthogonal with the scaling functions,

$$\langle\, \varphi_{j,k}, \widetilde{\varphi}_{j,k'} \,\rangle_w \;=\; \delta_{k-k'}. \tag{5.2}$$

The dual scaling functions satisfy refinement relations with coefficients $\{\widetilde{h}_{j,k,l}\}$. Note that the coefficients of the refinement relation can be written as

$$h_{j,k,l} \;=\; \langle\, \varphi_{j,k}, \widetilde{\varphi}_{j+1,2k+l} \,\rangle_w .$$

Let $N \Leftrightarrow 1$ be the highest degree of polynomials that can be represented as a linear combination of the $\{\varphi_{j,k}\}_k$, and similarly for $\widetilde{N}$.

We define the space W_j to be a complement of V_j in V_{j+1}, and let $\{\psi_{j,k}\}_k$ be a Riesz basis for this space. We have that

$$\psi_{j,k} \;=\; \sum_l g_{j,k,l}\, \varphi_{j+1,2k+l}, \tag{5.3}$$

and similarly for the dual wavelets. The dual wavelets $\widetilde{\psi}_{j,k}$ are biorthogonal to the wavelets, or

$$\langle\, \psi_{j,k}, \widetilde{\psi}_{j',k'} \,\rangle_w \;=\; \delta_{k-k'}\, \delta_{j-j'}.$$

The wavelets $\psi_{j,k}$ have $\widetilde{N}$ vanishing weighted moments,

$$\int_{-\infty}^{+\infty} w(x)\, x^p \psi_{j,k} \;=\; 0 \quad \text{for} \quad 0 \leqslant p < \widetilde{N}.$$

We want to construct compactly supported basis functions and dual functions, where the index j somehow corresponds to scale and the index k somehow to location. This is true if a closed interval $I \subset \mathbf{R}$ exists such that

$$\operatorname{supp} \varphi_{j,k} \;\subset\; 2^{-j}\,(I + k),$$

and similarly for the wavelets and dual functions. This implies that the coefficient sequences of the refinement relations are finite.

A fast wavelet transform is now given by

$$\begin{aligned} \lambda_{j,k} &= \sum_l \widetilde{h}_{j,k,k-2l}\, \lambda_{j+1,l} \\ \gamma_{j,k} &= \sum_l \widetilde{g}_{j,k,k-2l}\, \lambda_{j+1,l}, \end{aligned}$$

and

$$\lambda_{j+1,k} \;=\; \sum_l h_{j,l,k-2l}\,\lambda_{j,l} + \sum_l g_{j,l,k-2l}\,\gamma_{j,l}.$$

All filters are finite. The only difference with the algebraic fast wavelet transform is that the filter coefficients are different for every coefficient. Note that with this setting, we can satisfy properties (1)–(5) without the use of the algebraic structure.

We restrict our attention in this chapter to a more specific case. First of all, we use a more restrictive condition on the support in the sense that

$$\operatorname{supp}\varphi_{j,k} \;=\; 2^{-j}(\operatorname{supp}\varphi_{0,0} + k), \tag{5.4}$$

and similarly for the wavelets and dual functions. This implies that the index range (over l) of the non-zero coefficients in $\{h_{j,k,l}\}_l$ is independent of j and k.

Secondly, we fix the dual scaling functions and construct scaling functions, wavelets and dual wavelets with the desired properties. More precisely, we let the dual scaling functions be the indicator functions on the dyadic intervals,

$$\widetilde{\varphi}_{j,k} \;=\; \chi_I \quad \text{with} \quad I \;=\; [k2^{-j}, (k+1)2^{-j}).$$

Consequently,

$$\int_{-\infty}^{+\infty} \widetilde{\varphi}_{j,k}(x)\,dx \;=\; 2^{-j},$$

and

$$\sum_k \widetilde{\varphi}_{j,k} \;=\; 1. \tag{5.5}$$

The refinement relation for the dual scaling functions is

$$\widetilde{\varphi}_{j,k} \;=\; \widetilde{\varphi}_{j+1,2k} + \widetilde{\varphi}_{j+1,2k+1}. \tag{5.6}$$

The normalization might look strange at this moment but it will help simplify the notation of the construction. Because of (5.5) and the biorthogonality (5.2), the scaling functions satisfy

$$\int_{-\infty}^{+\infty} w(x)\,\varphi_{j,k}(x)\,dx \;=\; 1,$$

and thus

$$\sum_l h_{j,k,l} \;=\; 1.$$

From the refinement relations (5.1) and (5.6) and the biorthogonality (5.2) it follows that

$$h_{j,k,2l} + h_{j,k,2l+1} \;=\; \delta_l. \tag{5.7}$$

In this chapter we focus on the construction of the basis functions and do not go into the functional analytic point of view. For example, we do not discuss the issue of when the weighted wavelets form a Riesz basis for L_2^w, as it would lead us too far.

Example: The unbalanced Haar wavelets. A simple example of wavelets that are orthogonal with respect to a weighted inner product exists. It is a variant of the Haar wavelets. Consider the multiresolution analysis where V_j is the space of functions that are piecewise constant on dyadic intervals of length 2^{-j}. Obviously, the set $\{\varphi_{j,k}\}_k$ with $\varphi_{j,k} = \chi_I$ and $I = [k2^{-j}, (k+1)2^{-j})$ forms an orthogonal basis of V_j for any weight function. Let $\psi_{j,k}$ be a function that is piecewise constant on the intervals $[2^{-j}k, 2^{-j-1}(2k+1))$ and $[2^{-j-1}(2k+1), 2^{-j}(k+1))$, zero elsewhere, and that satisfies

$$\int_{-\infty}^{+\infty} w(x)\, \psi_{j,k}(x)\, dx \;=\; 0.$$

It then follows immediately that the set $\{\psi_{j,k}\}$ is orthogonal with respect to the weighted inner product. Since the wavelet is not symmetric, we call it the *unbalanced Haar wavelet*. These functions have the disadvantage that they only have one vanishing moment and that they are non-smooth. In the remainder of this chapter we construct smooth weighted wavelets with more vanishing moments.

Several other constructions of weighted wavelets already exist, see [15, 16, 140, 180]. They, however, have the disadvantage that the wavelets are not compactly supported.

5.3 Simple example

The construction of algebraic wavelets relies heavily on the Fourier transform. In the case of weighted wavelets, one can no longer use the Fourier transform. Instead, we use *average-interpolation*, an idea of David Donoho to construct biorthogonal wavelets [79]. His construction builds upon a subdivision scheme and does not make use of the Fourier transform.

In this section, we consider a simple example on how to adapt the Donoho average-interpolation scheme for the construction of weighted wavelets. The reasoning goes as follows. First we outline the properties we want the scaling functions to have. We show what the subdivision scheme looks like, assuming the scaling functions satisfy those properties. In the next section, we reverse the reasoning. We give the subdivision scheme for the general case and prove that it indeed yields scaling functions with the desired properties.

In our example we assume that the scaling functions satisfy the following properties:

1. $\varphi_{j,k}$ is continuous,
2. $\langle\, \varphi_{j,k}, \widetilde{\varphi}_{j,k'} \,\rangle_w \;= \delta_{k-k'}$,
3. $\operatorname{supp} \varphi_{j,k} = [(k \Leftrightarrow 2)\, 2^{-j}, (k+3)\, 2^{-j}]$,
4. $N = 2$.

The relationship between the width of the support and N is inspired by the biorthogonal algebraic wavelets of [45]. We show how to calculate the coefficients

$h_{j,k,l}$. For notational simplicity we consider the case $j = k = 0$ and omit these indices in the notation of the coefficients. The function $\varphi_{0,0}$ is supported on $[\Leftrightarrow 2, 3]$ and we look for 6 coefficients $h_{-2}, \ldots, h_3$ so that

$$\varphi_{0,0} \;=\; \sum_{l=-2}^{3} h_l \, \varphi_{1,l},$$

where we know that

$$h_l \;=\; \langle \, \varphi_{0,0}, \widetilde{\varphi}_{1,l} \, \rangle_w \qquad \text{for} \qquad \Leftrightarrow 2 \leqslant l \leqslant 3.$$

Construct a second degree polynomial $P(x)$ so that

$$\begin{aligned} \langle \, P, \widetilde{\varphi}_{0,-1} \, \rangle_w &= 0 \\ \langle \, P, \widetilde{\varphi}_{0,0} \, \rangle_w &= 1 \\ \langle \, P, \widetilde{\varphi}_{0,1} \, \rangle_w &= 0. \end{aligned}$$

This polynomial exists in general, as it is determined by three linear equations. By assumption it can be written as a linear combination of the $\varphi_{0,k}$. Because of the compact support (5.4), this summation only involves five terms on the interval $[0, 1]$,

$$P(x) \;=\; \sum_{k=-2}^{2} a_k \, \varphi_{0,k}(x) \qquad \text{for} \qquad x \in [0, 1].$$

By construction $a_{-1} = a_1 = 0$ and $a_0 = 1$. Again because of the compact support, we have

$$\begin{aligned} \langle \, P, \widetilde{\varphi}_{1,0} \, \rangle_w &= a_{-2} \, \langle \, \varphi_{0,-2}, \widetilde{\varphi}_{1,0} \, \rangle_w \, + \, \langle \, \varphi_{0,0}, \widetilde{\varphi}_{1,0} \, \rangle_w \, + a_2 \, \langle \, \varphi_{0,2}, \widetilde{\varphi}_{1,0} \, \rangle_w \\ &= \langle \, \varphi_{0,0}, \widetilde{\varphi}_{1,0} \, \rangle_w \\ &= h_0, \end{aligned}$$

and similarly

$$h_1 \;=\; \langle \, P, \widetilde{\varphi}_{1,1} \, \rangle_w .$$

The coefficients h_{-2} and h_{-1} can be found by constructing $P(x)$ so that

$$\begin{aligned} \langle \, P, \widetilde{\varphi}_{0,-2} \, \rangle_w &= 0 \\ \langle \, P, \widetilde{\varphi}_{0,-1} \, \rangle_w &= 0 \\ \langle \, P, \widetilde{\varphi}_{0,0} \, \rangle_w &= 1, \end{aligned}$$

and then taking the inner products with $\widetilde{\varphi}_{1,-2}$ and $\widetilde{\varphi}_{1,-1}$. The construction for h_2 and h_3 again is similar. By repeating this construction for all j and k we can calculate all coefficients $h_{j,k,l}$. These coefficients can then be used to construct the scaling functions in a subdivision scheme.

Note: This construction in case $w \equiv 1$, results in a biorthogonal algebraic scaling function from [45]. Its coefficients $\{h_k\}$ are equal to

$$\{\Leftrightarrow 1/16, \; 1/16, \; 1/2, \; 1/2, \; 1/16, \; \Leftrightarrow 1/16\}.$$

We refer to the case $w \equiv 1$ as the unweighted case.

5.4 The subdivision scheme

We describe the subdivision scheme for general N. From symmetry arguments one can understand that N has to be odd. We let $N = 2D + 1$. Suppose one wants to synthesize a function of V_i,

$$f \;=\; \sum_k \lambda_{i,k}\, \varphi_{i,k}. \tag{5.8}$$

The idea of a subdivision scheme is to write this function in the basis of a finer scale space V_j with $j > i$,

$$f \;=\; \sum_k \lambda_{j,k}\, \varphi_{j,k},$$

and to let j tend to infinity. Note that these two formulas at this moment only hold formally, as we haven't given a meaning to the symbol $\varphi_{j,k}$ yet. The only thing we know is that we are looking for an f with

$$\lambda_{i,k} \;=\; \langle\, f, \widetilde{\varphi}_{i,k} \,\rangle_w .$$

One step of the subdivision scheme consists of, given the coefficients $\lambda_{j,k}$ on one level, calculating the coefficients on the next finer level $\lambda_{j+1,k}$. For each group of N coefficients $\{\lambda_{j,k-D}, \ldots, \lambda_{j,k}, \ldots, \lambda_{j,k+D}\}$, it involves two steps:

1. Construct a polynomial P of degree N so that

$$\langle\, P, \widetilde{\varphi}_{j,k+l} \,\rangle_w \;=\; \lambda_{j,k+l} \quad \text{for} \quad \Leftrightarrow D \leqslant l \leqslant D.$$

2. Calculate two coefficients on the next finer level as

$$\lambda_{j+1,2k} \;=\; \langle\, P, \widetilde{\varphi}_{j+1,2k} \,\rangle_w \quad \text{and} \quad \lambda_{j+1,2k+1} \;=\; \langle\, P, \widetilde{\varphi}_{j+1,2k+1} \,\rangle_w .$$

To write the subdivision scheme in detail, we define the *local moments* as

$$M^p_{j,k} \;=\; \langle\, x^p, \widetilde{\varphi}_{j,k} \,\rangle_w \quad \text{for} \quad 0 \leqslant p < N.$$

Note that

$$M^p_{j,k} \;=\; M^p_{j+1,2k} + M^p_{j+1,2k+1}.$$

Define the $2 \times N$ matrix

$$H_{j,k} \;=\; \begin{bmatrix} M^0_{j+1,2k} & \cdots & M^{N-1}_{j+1,2k} \\ M^0_{j+1,2k+1} & \cdots & M^{N-1}_{j+1,2k+1} \end{bmatrix} \begin{bmatrix} M^0_{j,k-D} & M^1_{j,k-D} & \cdots & M^{N-1}_{j,k-D} \\ \cdots & \cdots & \cdots & \cdots \\ M^0_{j,k} & M^1_{j,k} & \cdots & M^{N-1}_{j,k} \\ \cdots & \cdots & \cdots & \cdots \\ M^0_{j,k+D} & M^1_{j,k+D} & \cdots & M^{N-1}_{j,k+D} \end{bmatrix}^{-1},$$

where we assume that the second matrix is invertible. It follows that the subdivision scheme can be written as

$$\forall k \in \mathbf{Z}: \quad \begin{bmatrix} \lambda_{j+1,2k} \\ \lambda_{j+1,2k+1} \end{bmatrix} = H_{j,k} \begin{bmatrix} \lambda_{j,k-D} \\ \vdots \\ \lambda_{j,k+D} \end{bmatrix}.$$

This defines a subdivision operator $\mathcal{U}_j$ so that

$$\{\lambda_{j+1,k}\}_k = \mathcal{U}_j \{\lambda_{j,k}\}_k.$$

It also immediately follows that

$$\lambda_{j,k} = \lambda_{j+1,2k} + \lambda_{j+1,2k+1},$$

and thus

$$\sum_k \lambda_{j,k} = \sum_k \lambda_{j+1,k}. \tag{5.9}$$

One now finds the coefficients $h_{j,k,l}$ by letting one of the $\lambda_{j,k}$ be equal to one and all others equal to zero, or, more precisely

$$\{h_{j,k,l}\}_l = \mathcal{U}_j \{\delta_{k-l}\}_l.$$

For each j and k, there are $2N$ non-zero coefficients $h_{j,k,l}$, namely the ones with $l = \Leftrightarrow N+1, \ldots, N$. The relationship between the $H_{j,k}$ and $h_{j,k,l}$ is given by

$$H_{j,k} = \begin{bmatrix} h_{j,k-D,2D} & \cdots & h_{j,k-1,2} & h_{j,k,0} & h_{j,k+1,-2} & \cdots & h_{j,k+D,-2D} \\ h_{j,k-D,2D+1} & \cdots & h_{j,k-1,3} & h_{j,k,1} & h_{j,k+1,-1} & \cdots & h_{j,k+D,-2D+1} \end{bmatrix}.$$

In order to synthesize f, define the following series of functions ($j \geqslant i$),

$$f^{(j)} = \sum_k \lambda_{j,k} \widetilde{\varphi}_{j,k} / M^0_{j,k}.$$

The superscript is not to be confused with the derivative. Note that

$$\lambda_{i,k} = \langle f^{(i)}, \widetilde{\varphi}_{i,k} \rangle_w,$$

and also that

$$\begin{aligned} \langle f^{(j)}, \widetilde{\varphi}_{i,k} \rangle_w &= \sum_l \lambda_{j,l} \langle \widetilde{\varphi}_{j,l}, \widetilde{\varphi}_{i,k} \rangle_w / M^0_{j,l} \\ &= \sum_{l=k2^{j-i}}^{(k+1)2^{j-1}} \lambda_{j,l} \\ &= \lambda_{i,k}. \end{aligned} \tag{5.10}$$

In the next section we study when

$$\lim_{j\to\infty} f^{(j)}$$

converges uniformly. We then define f to be the limit function. In order to construct the scaling function $\varphi_{i,k}$, we start from the Kronecker sequence $\{\lambda_{i,l}\}_l = \{\delta_{k-l}\}_l$. This way we give meaning to the formal expression (5.8).

Note: In the unweighted case, the matrix $H_{j,k}$ is independent of j and k. Such a subdivision scheme is called a *stationary subdivision*, see [32], or a *cascade algorithm*, see [65]. The scheme then converges to an algebraic biorthogonal scaling function from the family described in [45]. The convergence of the subdivision scheme and the regularity of the solution is already studied extensively in case of stationary subdivision, see Section 2.6. In the next section we use some of these techniques to show the properties of the non-stationary scheme.

5.5 Convergence and regularity

5.5.1 Definitions

In this section we study the convergence of the subdivision scheme and the regularity of the limit function. Define

$$\mu_{j,k} \;=\; \frac{\lambda_{j,k}}{M^0_{j,k}}$$

so that

$$f^{(j)} \;=\; \sum_l \mu_{j,k}\,\widetilde{\varphi}_{j,k}.$$

Since

$$\lambda_{j,k} \;=\; \langle\, f, \widetilde{\varphi}_{j,k}\,\rangle_w\,,$$

we see that if f is continuous, then

$$\lim_{j\to\infty} \mu_{j,k\,2^{j-m}+l} \;=\; f(k\,2^{-m}),$$

for any fixed l. On the other hand, if for all k,

$$\lim_{j\to\infty} \mu_{j,k} \;=\; \lim_{j\to\infty} \mu_{j,k+l}$$

with l arbitrary but fixed, then a continuous limit function exists.

In order to study the convergence, we write the subdivision scheme in terms of the $\mu_{j,k}$. Define therefore the *normalized local moments* as

$$N^p_{j,k} \;=\; \frac{M^p_{j,k}}{M^0_{j,k}},$$

so that

$$\begin{bmatrix} \mu_{j+1,2k} \\ \mu_{j+1,2k+1} \end{bmatrix} = C_{j,k} \begin{bmatrix} \mu_{j,k-D} \\ \vdots \\ \mu_{j,k+D} \end{bmatrix},$$

where

$$C_{j,k} = \begin{bmatrix} 1 & N^1_{j+1,2k} & \cdots & N^{N-1}_{j+1,2k} \\ 1 & N^1_{j+1,2k+1} & \cdots & N^{N-1}_{j+1,2k+1} \end{bmatrix} \begin{bmatrix} 1 & N^1_{j,k-D} & \cdots & N^{N-1}_{j,k-D} \\ \cdots & \cdots & \cdots & \cdots \\ 1 & N^1_{j,k} & \cdots & N^{N-1}_{j,k} \\ \cdots & \cdots & \cdots & \cdots \\ 1 & N^1_{j,k+D} & \cdots & N^{N-1}_{j,k+D} \end{bmatrix}^{-1}.$$

Note that

$$C_{j,k} \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}. \tag{5.11}$$

This follows from the fact that if on level j all coefficients $\mu_{j,k}$ are equal to one, the polynomial $P(x)$ in the subdivision is always a constant equal to one.

Concentrate on what happens in the neighborhood of a point $x \in \mathbf{R}$. Define a series of vectors, each with $2N \Leftrightarrow 1$ components,

$$\mu_j = [\mu_{j,k-2D} \ \cdots \ \mu_{j,k} \ \cdots \ \mu_{j,k+2D}]^t \in \mathbf{C}^{2N-1},$$

where k depends on j in the following way:

$$k = k(j) = \lfloor 2^j x \rfloor \quad \text{such that} \quad x \in [k2^{-j}, (k+1)2^{-j}).$$

It is then easy to see that

$$\mu_{j+1} = T_j \mu_j,$$

where T_j is a $(2N \Leftrightarrow 1) \times (2N \Leftrightarrow 1)$ matrix that can be found as follows. In case $k(j+1) = 2k(j)$, T_j contains $C_{j,k-D}$, ..., $C_{j,k+D-1}$ and the top row of $C_{j,k+D}$, see Figure 5.1 (left). In case $k(j+1) = 2k(j)+1$, T_j consists of the bottom row of $C_{j,k-D}$, and $C_{j,k-D+1}$, ..., $C_{j,k+D}$, see Figure 5.1 (right). It follows from (5.11) that T_j has an eigenvalue equal to one with the corresponding eigenvector a constant vector. The function f is continuous at x in case the μ_j converge to a constant vector μ.

5.5.2 Unweighted case

In the unweighted case we use similar notation, but just switch to lowercase (i.e. n, c, h and t, with the proper sub- and superscripts). We immediately see that

$$n^p_{j,k} = 2^{-jp}((k+1)^{(p+1)} \Leftrightarrow k^{(p+1)})/(p+1).$$

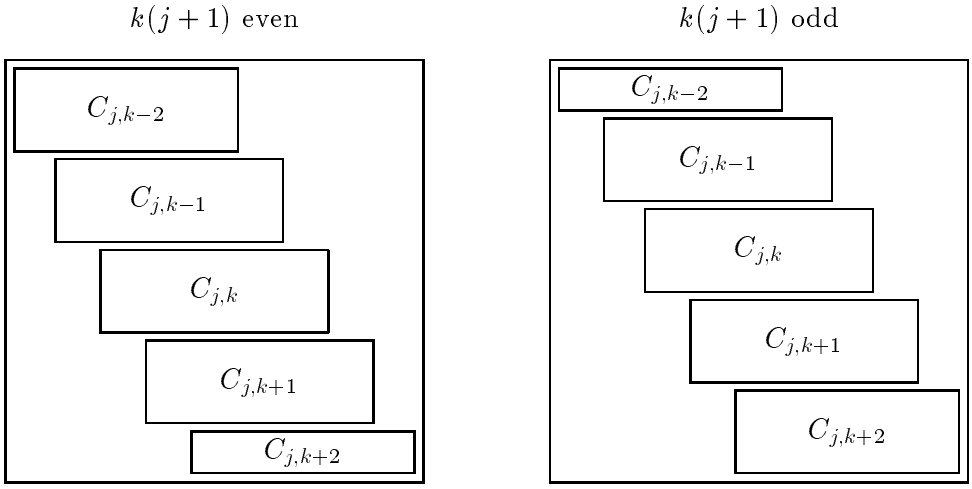

Figure 5.1: Structure of T_j in case $D = 2$.

One can understand that the matrix $c_{j,k}$ is independent of j and k. For example, if $D = 1$, we have

$$c_{j,k} = \begin{bmatrix} 1/16 & 1/2 & \Leftrightarrow 1/16 \\ \Leftrightarrow 1/16 & 1/2 & 1/16 \end{bmatrix}.$$

As we already pointed out, also the coefficients $h_{j,k,l}$ are independent of j and k. For the matrix t_j we only have two possibilities, t^0 and t^1, depending on the parity of $k(j)$. We state the main result from [79].

Theorem 5.1 (Donoho, 1993) *Let N be an odd integer greater than one. Start the subdivision scheme from the Kronecker sequence $\{\lambda_{i,l}\}_l = \{\delta_{k-l}\}_l$. Then the functions $f^{(j)}$ converge uniformly to a function $\varphi_{i,k}$. This function is uniformly continuous, compactly supported, and Hölder continuous of order R, with $R = R(N) > 0$. For an arbitrary initial sequence $\{\lambda_{i,l}\}$, the functions $f^{(j)}$ converge locally uniformly to a continuous function of the form*

$$f = \sum_k \lambda_{i,k}\, \varphi_{i,k}.$$

One can show that $R(3) \geqslant .678$, $R(5) \geqslant 1.272$, $R(7) \geqslant 1.826$, $R(9) \geqslant 2.354$, and asymptotically $R(N) \approx .2075N$ [79]. More accurate and local results can be obtained using the *joint spectral radius* of the matrices t^0 and t^1, restricted to the subspace orthogonal to the constant eigenvector.

5.5.3 Weighted case

We consider the difference between the weighted and unweighted case. We outline the main ideas of the reasoning. For a detailed treatment, we refer to [172].

We first note that on a sufficiently fine scale we come arbitrarily close to the unweighted case, since

$$\lim_{j\to\infty} \frac{N^p_{j,k}}{n^p_{j,k}} \;=\; 1.$$

Consequently, the coefficients $h_{j,k,l}$ converge to the coefficients h_l from the unweighted case if j tends to infinity. Next, we look for a condition on the weight that will allow us to characterize how fast this convergence is. It turns out that with a very mild condition on the weight function, along the lines of a bounded oscillation condition, one can assure that

$$\|C_{j,k} \Leftrightarrow c_{j,k}\| \;=\; \mathcal{O}(2^{-j}) \quad \text{and} \quad \|T_j \Leftrightarrow t_j\| \;=\; \mathcal{O}(2^{-j}).$$

This answers the question of the invertibility of the matrices in the subdivision scheme. We know that the matrices $c_{j,k}$ in the unweighted case are invertible. Since the determinant depends continuously on the entries of the matrix, a level j_0 exists so that all matrices $C_{j,k}$ on levels $j \geqslant j_0$ are invertible. This tells us that we can only construct "half" a multiresolution analysis, as the matrices on the levels $j < j_0$ are not necessarily invertible. One can then show that a result along the lines of Theorem 5.1 holds for the weighted case, provided $i > j_0$. In order to get exactly the same Hölder regularity as in the unweighted case, we need more restrictive conditions on the weight, which ensure faster convergence to the weighted case.

5.6 Basis functions

In the previous sections we saw how the scaling functions are defined as the limit of a uniformly convergent series. In this section we study the properties of these scaling functions and confirm that they correspond with our expectations. We also show how to construct the wavelets.

Theorem 5.2 *The weighted scaling functions satisfy*

1. $\operatorname{supp} \varphi_{j,k} \;=\; 2^{-j}[\Leftrightarrow N + 1 + k, N + k]$,
2. $\displaystyle\int_{-\infty}^{+\infty} w(x)\, \varphi_{j,k}(x)\, dx \;=\; 1$,
3. $\langle\, \varphi_{j,k}, \widetilde{\varphi}_{j,k'} \,\rangle_w \;=\; \delta_{k-k'}$,
4. $\displaystyle\sum_k M^p_{j,k}\, \varphi_{j,k} \;=\; x^p \quad \textit{for} \quad 0 \leqslant p < N.$

Proof : Consider again the series $f^{(j)}$, starting from $\lambda_{i,l} \;=\; \delta_{k-l}$, that converges to $\varphi_{i,k}$. Most of the properties of $\varphi_{i,k}$ then follow from the properties of $f^{(j)}$ and the uniform convergence.

1. $\operatorname{supp} f^{(j)} = 2^{-i}[\Leftrightarrow s + k, s + 1 + k]$, where $s = (N \Leftrightarrow 1)(1 \Leftrightarrow 2^{i-j})$.
2. Because of (5.9), we have
$$\int_{-\infty}^{+\infty} w(x)\, f^{(j)}(x)\, dx \;=\; \sum_l \lambda_{j,l} \;=\; \sum_l \lambda_{i,l} \;=\; 1.$$
3. From (5.10), it follows that
$$\langle\, f^{(j)}, \widetilde{\varphi}_{j,k'} \,\rangle_w \;=\; \delta_{k-k'}.$$
4. We first take $p = 0$. Start a subdivision scheme from $\lambda_{i,k} = M^0_{i,k}$. Then $f^{(j)} = 1$ and thus $f = 1$. The case $p > 0$ is a little different. One way to prove it would be to use higher order splines to construct the $f^{(j)}$, so that $f^{(j)}(x) = x^p$. It can also be understood as follows. Start the subdivision from $\lambda_{i,k} = M^p_{i,k}$. We then see that
$$\lim_{j\to\infty} f^{(j)}(x) \;=\; x^p,$$
if x is not dyadic. The result for all x follows from the fact that the limit function is continuous.

□

The next problem is how to find the wavelets and dual wavelets. The answer is given in the following theorem.

Theorem 5.3 *Assume that the scaling functions and the dual scaling functions are given as above. Choose the wavelet and dual wavelet as*

$$\psi_{j,k} \;=\; \varphi_{j+1,2k} \Leftrightarrow \varphi_{j+1,2k+1},$$

and

$$\widetilde{\psi}_{j,k} \;=\; \sum_l g_{j,k,l}\, \widetilde{\varphi}_{j+1,2k+l} \qquad \textit{with} \qquad g_{j,k,l} \;=\; (\Leftrightarrow 1)^l\, h_{j,k+\lfloor l/2 \rfloor, 1-l}.$$

Then

1. $\langle\, \widetilde{\varphi}_{j,k}, \psi_{j,k'} \,\rangle_w = 0,$
2. $\langle\, \varphi_{j,k}, \widetilde{\psi}_{j,k'} \,\rangle_w = 0,$
3. $\langle\, \psi_{j,k}, \widetilde{\psi}_{j',k'} \,\rangle_w = \delta_{j-j'}\, \delta_{k-k'},$
4. $\displaystyle\int_{-\infty}^{+\infty} w(x)\, \psi_{j,k}(x)\, dx = 0,$
5. $\displaystyle\int_{-\infty}^{+\infty} w(x)\, x^p\, \widetilde{\psi}_{j,k}(x)\, dx = 0,$ *for* $0 \leqslant p < N,$

6. $\operatorname{supp}\psi_{j,k} \;=\; \operatorname{supp}\widetilde{\psi}_{j,k} \;=\; 2^{-j}[\Leftrightarrow D+k, D+1+k]$.

Proof : The biorthogonality follows from writing out the refinement equations and then using (5.7).

1. This case is the easiest.
$$\langle\, \widetilde{\varphi}_{j,k}, \psi_{j,k'} \,\rangle_w \;=\; \langle\, \widetilde{\varphi}_{j+1,2k} + \widetilde{\varphi}_{j+1,2k+1}, \varphi_{j+1,2k} \Leftrightarrow \widetilde{\varphi}_{j+1,2k+1} \,\rangle_w \;=\; 0.$$
2. From the refinement relations, it follows that
$$\langle\, \varphi_{j,k}, \widetilde{\psi}_{j,k-m} \,\rangle_w \;=\; \sum_l g_{j,k,l}\, h_{j,k-m,l+2m}.$$
Substitution yields
$$\sum_l (\Leftrightarrow 1)^l\, h_{j,k+\lfloor l/2\rfloor,1-l}\, h_{j,k-m,l+2m},$$
or, after splitting even and odd,
$$\sum_l \left(h_{j,k+l,1-2l}\, h_{j,k-m,2l+2m} \Leftrightarrow h_{j,k+l,-2l}\, h_{j,k-m,2l+2m+1}\right).$$
This is trivially zero in case $|m| > 2D$ because of the finiteness of the sequences. In case $m = 0$, all terms except the one with $l = 0$ cancel because of (5.7), while the term with $l = 0$ is equal to
$$h_{j,k,1}\, h_{j,k,0} \Leftrightarrow h_{j,k,0}\, h_{j,k,1} \;=\; 0.$$
In case $m \neq 0$, all terms cancel because of (5.7), except the ones with $l = 0$ or $l = \Leftrightarrow m$. These terms are equal to
$$h_{j,k,1}h_{j,k-m,2m} \Leftrightarrow h_{j,k,0}h_{j,k-m,2m+1} + h_{j,k-m,1+2m}h_{j,k-m,0} \Leftrightarrow h_{j,k-m,2m}h_{j,k-m,1},$$
which is equal to
$$h_{j,k-m,2m} \Leftrightarrow h_{j,k-m,2m}.$$
3. We first consider $j = j'$.
$$\langle\, \psi_{j,k}, \widetilde{\psi}_{j,k+l} \,\rangle_w \;=\; g_{j,k,2l} \Leftrightarrow g_{j,k,2l+1} \;=\; h_{j,k+l,1-2l} + h_{j,k+l,-2l} \;=\; \delta_l.$$
The case $j \neq j'$ now follows from combining this with (1) and (2).
4. Follows immediately from the definition.
5. Follows from the biorthogonality and the polynomial reproduction of the scaling functions.
6. Exercise.

□

The vanishing moment property implies that if a function f belongs to $\mathcal{C}^N$, then the weighted wavelet coefficients decay as
$$\langle\, f, \widetilde{\psi}_{j,k} \,\rangle_w \;=\; \mathcal{O}(2^{-j(N+1)}).$$

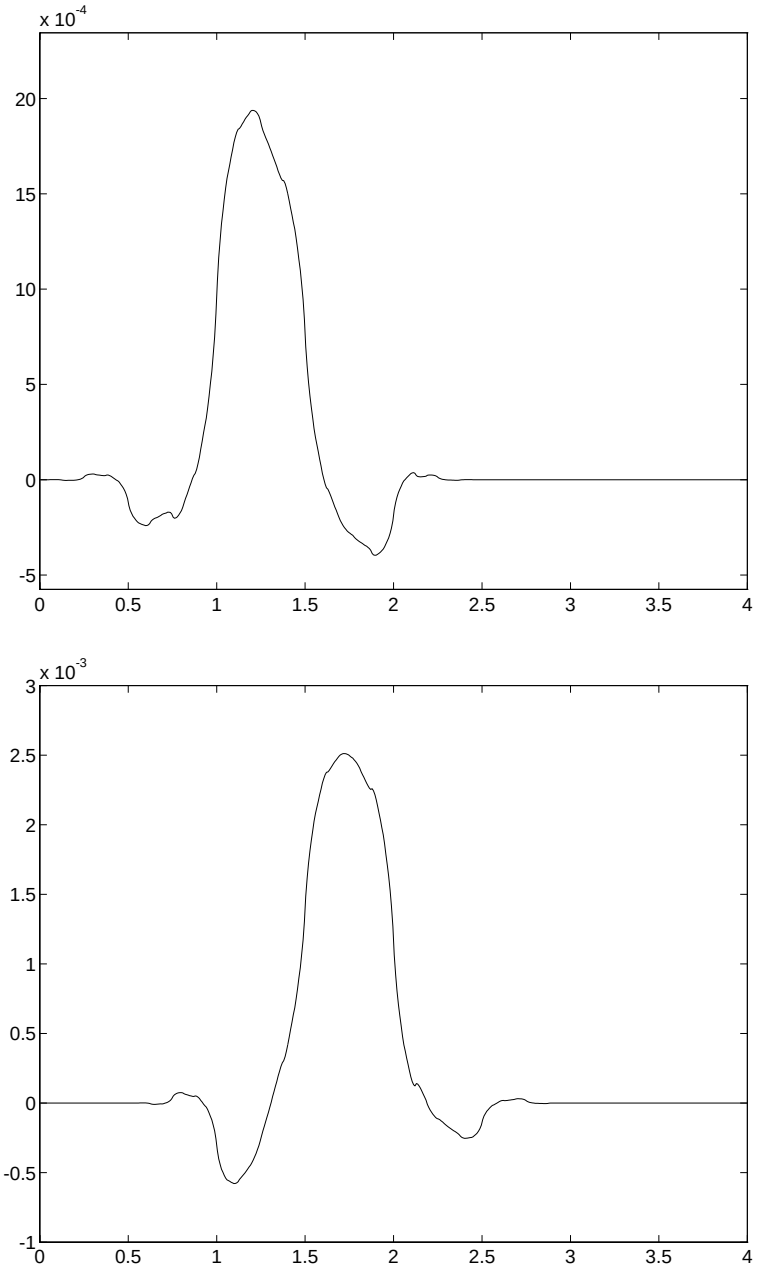

Figure 5.2: Weighted scaling functions $\varphi_{1,2}$ and $\varphi_{1,3}$.

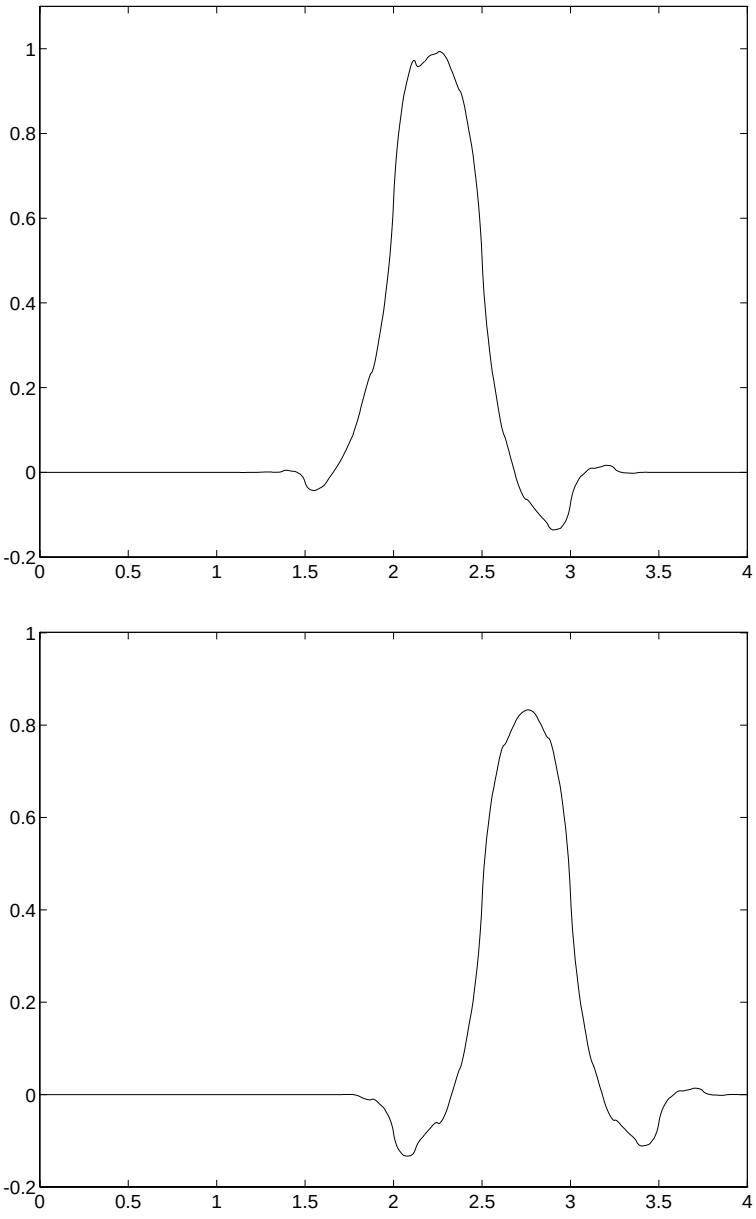

Figure 5.3: Weighted scaling functions $\varphi_{1,4}$ and $\varphi_{1,5}$.

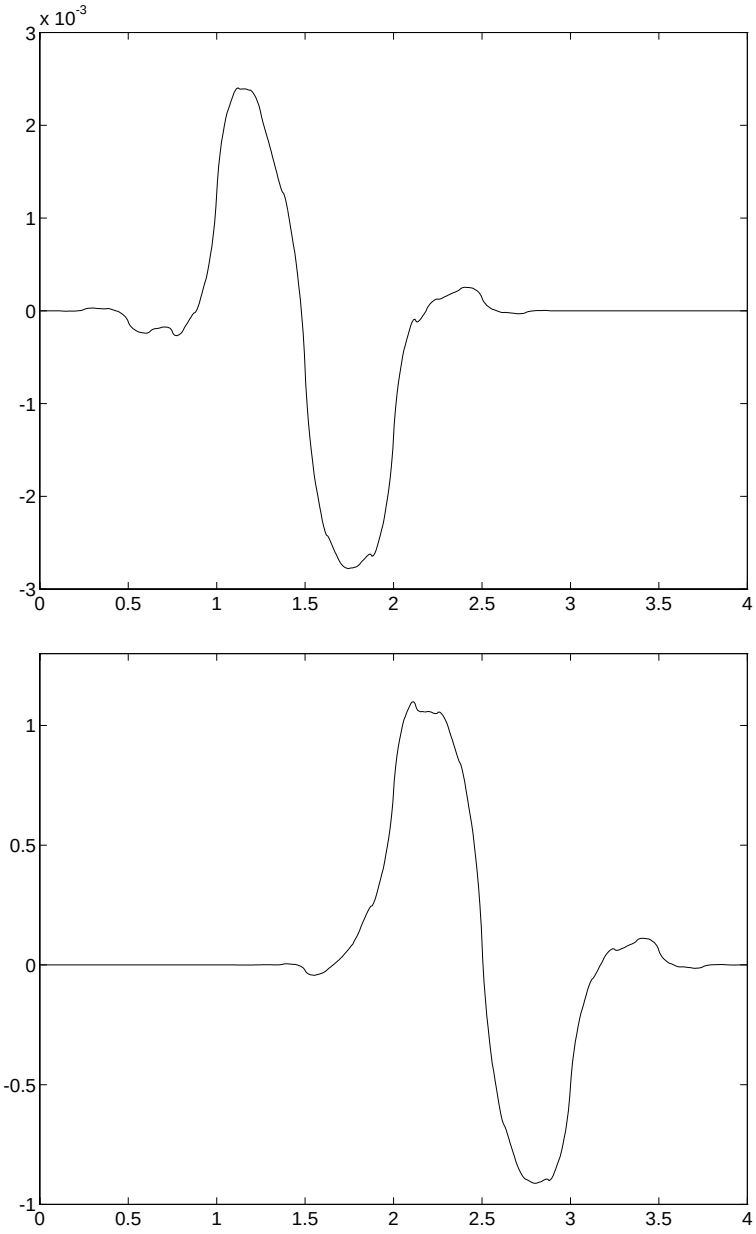

Figure 5.4: Weighted wavelets $\psi_{0,1}$ and $\psi_{0,2}$.

5.7 Example

We consider an example with the following weight function:

$$w(x) \;=\; x\,(\chi_{(-\infty,\alpha]} + 10^3\,\chi_{(\alpha,\infty)}) \quad \text{with} \quad \alpha \;=\; 5/\sqrt{8}.$$

This is a linear function with a jump of 3 magnitudes at $\alpha \approx 1.77$. The point α is chosen so that is doesn't coincide with a dyadic. Figures 5.2 and 5.3 show the weighted scaling functions $\varphi_{1,2}, \ldots, \varphi_{1,5}$. Figure 5.4 shows the weighted wavelets $\psi_{0,1}$ and $\psi_{0,1}$. We see that the wavelets and scaling functions are smooth in the neighborhood of α even though the weight has a large jump. Note that the basis functions on the left of α are much smaller, because of the normalization. All the matrices $C_{j,k}$ here are invertible.

5.8 Applications and future research

In the next chapter we will show how the weighted functions can be used for the solution of differential equations. Here we mention just one application. In Chapter 8 we will mention others.

Consider equally spaced points $\{x_k\}$, data $\{y_k\}$, and positive weights $\{w_k\}$. Try to find a function $y(x)$ that is smooth and minimizes

$$\sum_k w_k\,(y(x_k) \Leftrightarrow y_k)^2.$$

One can do this by building the weighted wavelets and using algorithms similar to the ones described in Section 2.12.1. The algorithm (including the calculation of the transform coefficients) is linear for any set of weights $\{w_k\}$. Also, the smoothness of $y(x)$ can easily be controlled by the decay of its weighted wavelet coefficients.

An initial direction for future research involves the study of the functional analytic properties of the weighted wavelets. It involves addressing the problem of when the weighted wavelets form a Riesz basis for L_2^w. One way would be to put extra conditions on the weight function, such as boundedness from above and below which implies that $\mathrm{L}_2^w = \mathrm{L}_2$. In that case, it is shown that the unbalanced Haar wavelets provide an unconditional basis [51]. This is highly non-trivial. An interesting problem is to find the mildest condition on the weight function for the weighted wavelets to form a basis. In the theory of weighted spaces one typically uses weight functions that belong to a Muckenhoupt class [163]. A natural problem that arises is how conditions on the weight function relate to these Muckenhoupt classes.

The construction of this chapter generates wavelets with only one vanishing moment and dual wavelets with several vanishing moments. One way to construct

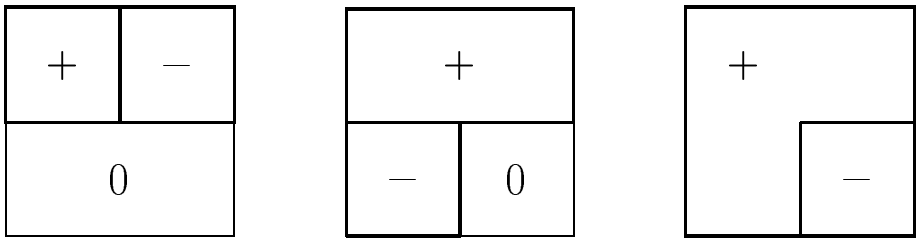

Figure 5.5: Mitrea wavelets.

wavelets with more vanishing moments is to use higher order B-splines as dual scaling functions.

A third direction involves the generalization to the multivariate case. Therefore, one first needs to design a multivariate average-interpolation scheme. Secondly, one has to construct compactly supported wavelets. This is highly non-trivial. In the case of the unbalanced Haar wavelet, it was solved only recently by Marius Mitrea [144]. His idea is simple, but extremely clever. We explain it in two dimensions. On each dyadic square, construct three unbalanced Haar wavelets, each of them constant and positive on one subset of the square, negative and constant on another, and zero elsewhere. These subsets are shown in Figure 5.5. If the values on these subsets are chosen so that the integral of the wavelets against the weight function vanishes, the Mitrea wavelets form an orthogonal set. The construction of smooth weighted wavelets from smooth weighted scaling functions could be inspired by this idea.

Finally, a drawback of the construction in this chapter is that it does not give a complete multiresolution analysis. Instead, one only finds the basis functions for levels bigger than j_0. A way to solve this is to give up on the dyadic structure of the dual scaling functions. Instead of letting them be indicator functions on dyadic intervals, one can use much more general partitionings. This implies relaxing condition (5.4). The extra flexibility can be used to ensure that the matrices of the subdivision scheme are invertible on every level.

Chapter 6

Wavelets for the Fast Solution of Ordinary Differential Equations

"γνωτι σεαυτον"
—Socrates, *(± 420 B.C.)*.

6.1 Introduction

As we saw in the previous chapters, wavelets have shown to be a powerful tool and a potential substitute for the Fourier transform in many problems. It is natural to use them for the solution of differential equations. In this chapter, we show how to use wavelets in the numerical solution of boundary value ordinary differential equations. Rather than using algebraic wavelets, we adapt the wavelets to the specific operator at hand. We want their construction to be easy to implement and computationally inexpensive in order to build a general solver. We show how to do this using the weighted wavelets of the previous chapter.

The chapter is organized as follows. In Section 6.2, we give a short outline of existing methods and their properties. We discuss the general idea of our construction in Section 6.3. We describe the construction and algorithm in detail for harmonic operators in Section 6.4, for the Helmholtz operator in Section 6.5, and for general variable coefficient operators in Section 6.6. Finally, we give some examples and discuss directions for future research.

6.2 Existing methods

Consider a linear ordinary differential equation of the form

$$\mathcal{L}\,u(x) \;=\; f(x) \quad \text{for} \quad x \in [0,1], \quad \text{where} \quad \mathcal{L} \;=\; \sum_{j=0}^{m} a_j(x)\,D^j,$$

with the boundary conditions

$$(\mathcal{B}_0\, u)\,(0) \;=\; g_0 \quad \text{and} \quad (\mathcal{B}_1\, u)\,(1) \;=\; g_1,$$

where

$$\mathcal{B}_i \;=\; \sum_{j=0}^{m} b_{i,j}\, D^j.$$

Presently, two major numerical solution techniques exist. First, if the coefficients $a_j(x)$ are independent of x, the Fourier transform is best suited for solving the equation. The underlying reason is that the complex exponentials are eigenfunctions of a constant coefficient operator. In the Fourier basis, the operator becomes diagonal and can thus trivially be inverted. The algorithm consists of calculating the Fourier transform of the right-hand side, dividing the coefficient of each basis function by the corresponding eigenvalue and taking the inverse Fourier transform. This can be implemented using the FFT with a complexity of $\mathcal{O}(M \log M)$, where M is the number of discretization points.

If the coefficients $a_j(x)$ are not constant, one typically uses finite element or finite difference methods. We focus here on the former and define the *operator inner product* associated with a self-adjoint operator $\mathcal{L}$ as

$$\langle\langle\, u, v \,\rangle\rangle \;=\; \langle\, \mathcal{L}\, u, v \,\rangle\,.$$

In a Galerkin method, one considers two spaces S and S^*, and looks for an approximate solution $\bar{u} \in S$ so that

$$\forall v \in S^* : \quad \langle\langle\, \bar{u}, v \,\rangle\rangle \;=\; \langle\, f, v \,\rangle\,.$$

If S and S^* are finite dimensional spaces with the same dimension, this leads to a linear system of equations. The matrix of this system is called the *stiffness matrix*. Its elements are the operator inner products of the basis functions of S and S^*.

Traditionally, one uses local finite elements which lead to a banded stiffness matrix. Since the matrix is sparse, the linear system is traditionally solved with an iterative method. Local finite elements, however, have the disadvantage that the stiffness matrix becomes ill-conditioned as the problem size grows. Typically, its condition number grows as a power of the number of elements. This slows down the convergence rate of the iterative algorithm dramatically.

This problem can be solved by using multiresolution techniques such as multigrid or hierarchical basis functions [100, 199]. Multiresolution finite element bases can provide preconditioners that result in a bounded condition number, see e.g. [29, 60, 145]. The convergence rate of the iterative solver is then independent of the problem size.

One possible way of using wavelets is to take (bi)orthogonal algebraic wavelets as basis functions in a Galerkin method. This was proposed by several researchers [10, 11, 92, 150]. It results in a linear system that is sparse because of the compact

support of the wavelets, and that, after preconditioning, has a condition number independent of problem size because of the multiresolution structure. However, in this setting the wavelets do not provide significantly better results than the other multiresolution techniques. In fact, one of their major properties, namely their (bi)orthogonality, is not fully exploited.

Three questions are addressed in this chapter:

1. How can one make use of the (bi)-orthogonality property of the wavelets?
2. Can wavelets diagonalize differential operators?
3. Can one construct fast algorithms?

6.3 General idea

We assume that $\mathcal{L}$ is self-adjoint and positive definite. Now write

$$\mathcal{L} \;=\; \mathcal{V}^* \mathcal{V},$$

where $\mathcal{V}^*$ is the adjoint of $\mathcal{V}$. We call $\mathcal{V}$ the *square root operator* of $\mathcal{L}$. Note that it is not unique.

Suppose that the functions $\{\Psi_{j,k}\}$ and $\{\Psi^*_{j,k}\}$, for an appropriate range of indices, are bases for S and S^* respectively. The entries of the stiffness matrix are then given by

$$\langle\langle\, \Psi_{j,k}, \Psi^*_{j',k'} \,\rangle\rangle \;=\; \langle\, \mathcal{L}\, \Psi_{j,k}, \Psi^*_{j',k'} \,\rangle \;=\; \langle\, \mathcal{V}\, \Psi_{j,k}, \mathcal{V}\, \Psi^*_{j',k'} \,\rangle\,.$$

The idea is to let

$$\Psi_{j,k} \;=\; \mathcal{V}^{-1}\, \psi_{j,k} \quad \text{and} \quad \Psi^*_{j,k} \;=\; \mathcal{V}^{-1}\, \widetilde{\psi}_{j,k}\,,$$

where $\psi_{j,k}$ and $\widetilde{\psi}_{j,k}$ are biorthogonal wavelets. Because of the biorthogonality, the stiffness matrix becomes a diagonal matrix and thus can trivially be inverted. This avoids the use of an iterative algorithm. We call the $\Psi_{j,k}$ and $\Psi^*_{j,k}$ functions the *operator wavelets* and the $\psi_{j,k}$ and $\widetilde{\psi}_{j,k}$ functions the *original wavelets*. The operator wavelets are biorthogonal with respect to the operator inner product, a property we refer to as *operator biorthogonality*. When using the operator wavelets as basis functions in a Galerkin method, the stiffness matrix becomes diagonal. Note that one can obtain diagonalization even though the operator wavelets are not eigenfunctions. This is not always a "true" diagonalization, as the operator wavelets and dual operator wavelets can differ. However, computationally the difference is irrelevant.

This idea has potential provided one can find a fast numerical algorithm to compute the operator wavelets. Therefore, the operator wavelets need to generate an *operator multiresolution analysis* with an associated fast wavelet transform.

This implies compactly supported operator wavelets and operator scaling functions $\Phi_{j,k}$. We will see that the latter cannot be constructed by simply applying $\mathcal{V}^{-1}$ to the original scaling functions.

The analysis is relatively straightforward for simple constant coefficient operators such as the Laplace and polyharmonic operator. The reason is that these operators preserve the algebraic structure of wavelets. The construction of the operator wavelets can thus rely on the Fourier transform. The situation becomes different for more general constant coefficient and for variable coefficient operators. We show how one then can use weighted wavelets.

The idea to adapt wavelets to a differential operator is also suggested elsewhere. In [57, 58], Stefan Dahlke and Ilona Weinreich construct wavelets that are operator *semi*orthogonal. As a result, one does not obtain a full diagonalization, but rather a decoupling of equations corresponding to different levels. In [128, 197], antiderivates of wavelets are used in a Galerkin method. This parallels our construction in the case of the Laplace or polyharmonic operator.

Our idea also is different from the technique Gregory Beylkin presents in [22]. He uses algebraic wavelets for the rapid calculation of the inverse of the matrix coming from a finite difference discretization. He also shows that the wavelets provide a diagonal preconditioner that yields uniformly bounded condition numbers.

6.4 Harmonic operators

The one-dimensional Laplace operator and a possible square root are

$$\mathcal{L} \;=\; \Leftrightarrow D^2 \quad \text{and} \quad \mathcal{V} \;=\; D.$$

The associated *operator inner product* is therefore the homogeneous Sobolev inner product,

$$\langle\langle\, u, v \,\rangle\rangle \;=\; \langle\, u', v' \,\rangle\,.$$

Since the action of $\mathcal{V}^{-1}$ is simply taking the antiderivative, the operator mother wavelets are given by

$$\Psi(x) \;=\; \int_{-\infty}^{x} \psi(t)\, dt, \qquad \text{and} \qquad \Psi^*(x) \;=\; \int_{-\infty}^{x} \widetilde{\psi}(t)\, dt.$$

Here ψ and $\widetilde{\psi}$ are compactly supported biorthogonal algebraic wavelets. The operator wavelets are also compactly supported because the integral of the original wavelets vanishes. Since translation and dilation is preserved, we define the operator wavelets as

$$\Psi_{j,k}(x) \;=\; \Psi(2^j x \Leftrightarrow k) \quad \text{and} \quad \Psi^*_{j,k}(x) \;=\; \Psi^*(2^j x \Leftrightarrow k).$$

It immediately follows that

$$\langle\langle\, \Psi^*_{j,k}, \Psi_{j',k'} \,\rangle\rangle \;=\; 2^j\, \delta_{j-j'}\, \delta_{k-k'}.$$

Consequently, the stiffness matrix is diagonal with powers of 2 on its diagonal.

We now define the spaces

$$\mathbf{W}_j \;=\; \text{span}\,\{\Psi_{j,k} \mid k \in \mathbf{Z}\}.$$

We want to find the associated multiresolution analysis. In other words, we need spaces $\mathbf{V}_j$ so that

$$\mathbf{V}_{j+1} \;=\; \mathbf{V}_j \oplus \mathbf{W}_j,$$

and operator scaling functions $\Phi_{j,k}$ so that

$$\mathbf{V}_j \;=\; \text{span}\,\{\Phi_{j,k} \mid k \in \mathbf{Z}\}.$$

These spaces are closed as they are finite dimensional.

The antiderivative of the original scaling function is not compactly supported and hence cannot be used as an operator scaling function. We instead construct the operator scaling function Φ by taking the convolution of the original scaling function with the indicator function on $[0,1]$,

$$\Phi \;=\; \varphi * \chi_{[0,1]},$$

and let

$$\Phi_{j,k}(x) \;=\; \Phi(2^j x \Leftrightarrow k).$$

Note that

$$\Phi'(x) \;=\; \varphi(x) \Leftrightarrow \varphi(x+1).$$

We next show that the $\mathbf{V}_j$ spaces are nested and that $\mathbf{W}_j$ complements $\mathbf{V}_j$ in $\mathbf{V}_{j+1}$.

In the Fourier domain we have

$$\widehat{\Phi}(\omega) \;=\; \frac{1 \Leftrightarrow e^{-i\omega}}{i\omega}\,\widehat{\varphi}(\omega),$$

and

$$\widehat{\Psi}(\omega) \;=\; \frac{1}{i\omega}\,\widehat{\psi}(\omega). \tag{6.1}$$

A simple calculation shows that the operator scaling function satisfies a refinement equation

$$\widehat{\Phi}(\omega) \;=\; \widehat{\Phi}(\omega/2)\,H(\omega/2) \quad \text{with} \quad H(\omega) \;=\; \frac{1+e^{-i\omega}}{2}\,h(\omega).$$

Consequently, the $\mathbf{V}_j$ spaces are nested. The space $\mathbf{W}_j$ is a subset of $\mathbf{V}_{j+1}$ if a trigonometric polynomial G exists so that

$$\widehat{\Psi}(\omega) \;=\; \widehat{\Phi}(\omega/2)\,G(\omega/2).$$

Substituting this in (6.1) yields,

$$G(\omega/2)\,\widehat{\Phi}(\omega/2) \;=\; \frac{1}{i\omega}\,g(\omega/2)\,\widehat{\varphi}(\omega/2).$$

It then follows that

$$\begin{aligned} G(\omega)\,\widehat{\varphi}(\omega)\frac{1 \Leftrightarrow e^{-i\omega}}{i\omega} &= \frac{1}{2i\omega}\,g(\omega)\,\widehat{\varphi}(\omega) \\ G(\omega)(1 \Leftrightarrow e^{-i\omega}) &= g(\omega)/2 \\ G(\omega) &= \frac{1/2}{1 \Leftrightarrow e^{-i\omega}}\,g(\omega). \end{aligned}$$

This function is a trigonometric polynomial, because g is a trigonometric polynomial with $g(0) = 0$.

The space $\mathbf{W}_j$ complements $\mathbf{V}_j$ in $\mathbf{V}_{j+1}$ if

$$\Delta(\omega) \;=\; \det \begin{bmatrix} H(\omega) & H(\omega+\pi) \\ G(\omega) & G(\omega+\pi) \end{bmatrix}$$

does not vanish, see Section 2.5. We readily see that

$$\Delta(\omega) \;=\; \delta(\omega)/4,$$

and $\delta(\omega)$ doesn't vanish since φ and ψ generate a multiresolution analysis. The construction of the dual functions Φ^* and Ψ^* from $\widetilde{\varphi}$ and $\widetilde{\psi}$ is completely similar. The coefficients of the trigonometric polynomials H, H^*, G and G^* now define the fast wavelet transform associated with the operator inner product.

Note that there is no reason why the operator scaling functions should be operator biorthogonal. One can even prove that this is impossible [110]. Note also that if true, this property would make the use of wavelets superfluous.

We have proven the following theorem, which we state informally:

Theorem 6.1 *Given an algebraic multiresolution analysis, one can construct a new multiresolution analysis adapted to the homogeneous Sobolev inner product by taking the antiderivate of the wavelets and convolving the scaling function with a box function. If the old basis and dual functions are compactly supported, so are the new ones.*

The idea to construct a new multiresolution analysis this way was also suggested independently elsewhere. Ingrid Daubechies mentioned it in [62] and David Donoho in [79]. Pierre-Gilles Lemarié-Rieusset used it in his "Formule de Commutation" [127], and it is also related to the difference subdivision schemes described in [32, pp. 150-151].

6.4.1 Algorithm

We describe the algorithm in the case of periodic boundary conditions. This implies that the basis functions on the interval $[0,1]$ are the periodization of the basis functions on the real line, see Section 2.9.

Let $S = \mathbf{V}_n$ (respectively $S^* = \mathbf{V}_n^*$) and consider a basis $\{\Phi_{n,k} \mid 0 \leqslant k < 2^n\}$ (respectively $\Phi^*_{n,k}$). Let $M = 2^n$. Define a vector $b \in \mathbf{C}^M$ as

$$b_k \;=\; \langle\, f, \Phi^*_{n,k} \,\rangle \quad \text{with} \quad 0 \leqslant k < M,$$

and a vector $x \in \mathbf{C}^M$ so that we can write $\bar{u} \in S$ as

$$\bar{u} \;=\; \sum_{k=0}^{M-1} x_k \, \Phi_{n,k}.$$

The Galerkin method with these bases then yields a linear system

$$A\,x \;=\; b \quad \text{with} \quad A_{k,l} \;=\; \langle\langle\, \Phi^*_{n,l}, \Phi_{n,k} \,\rangle\rangle\,.$$

As we mentioned earlier, the matrix A cannot be diagonal. Also, its condition number grows as $\mathcal{O}(M^2)$. Consider the decomposition

$$\mathbf{V}_n \;=\; \mathbf{V}_0 \oplus \mathbf{W}_0 \oplus \cdots \oplus \mathbf{W}_{n-1},$$

and the corresponding wavelet basis. The space $\mathbf{V}_0$ has dimension one and contains constant functions. We switch to a one index notation so that the sets

$$\{1, \Psi_{j,k} \mid 0 \leqslant j < n,\ 0 \leqslant k < 2^j\} \quad \text{and} \quad \{\Psi_k \mid 0 \leqslant k < 2^n\}$$

coincide and similarly for the dual functions. Define the vectors b' and x' of $\mathbf{C}^M$ so that

$$b'_k \;=\; \langle\, f, \Psi^*_k \,\rangle \quad \text{and} \quad \bar{u} \;=\; \sum_{k=0}^{M-1} x'_k \, \Psi_k.$$

We know that matrices T and T^* exists so that

$$b' \;=\; T^*\, b \quad \text{and} \quad x \;=\; T\, x'.$$

The action of the matrix T (respectively T^*) can be implemented using the fast wavelet transform decomposition with filters H and G (respectively H^* and G^*). The complexity of the matrix vector multiplication is $\mathcal{O}(M)$. In the wavelet basis the system becomes

$$A'\,x' \;=\; b' \quad \text{with} \quad A' \;=\; T^*\,A\,T,$$

where

$$A'_{k,l} \;=\; \langle\langle\, \Psi^*_{n,l}, \Psi_{n,k} \,\rangle\rangle\,.$$

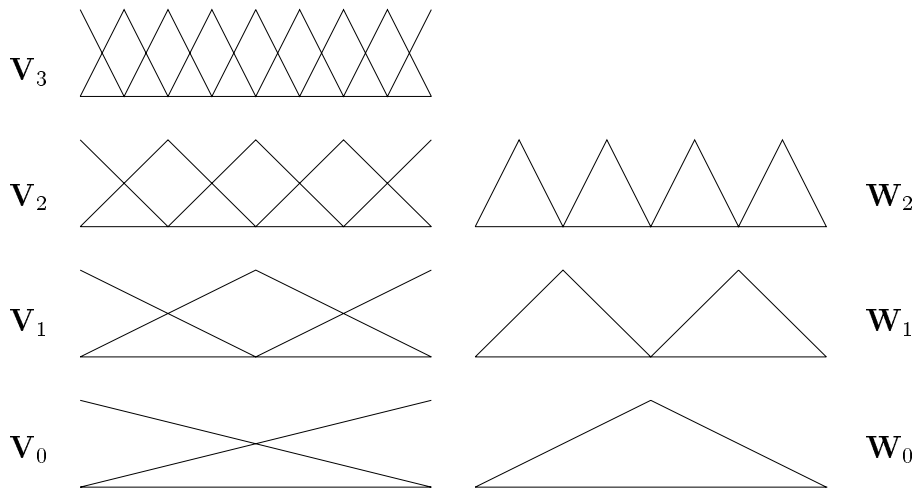

Figure 6.1: Basis for Dirichlet problem.

Since A' is diagonal, it can be trivially inverted. The coefficients of the solution in the scaling function basis are given by

$$x \;=\; T\,A'^{-1}\,T^{*}\,b.$$

The algorithm consists of calculating the wavelet decomposition of the right-hand side, dividing each coefficient by its corresponding diagonal element and reconstructing to find the solution. The complexity is $\mathcal{O}(M)$.

The constant basis function of $\mathbf{V}_0$ has a zero as corresponding diagonal element and its coefficient is thus undetermined. Indeed, the solution is only defined up to a constant. This does not lead to a division by zero as the integral of f has to vanish,

$$\int_0^1 f(x)\,dx \;=\; u'(1) \Leftrightarrow u'(0) \;=\; 0.$$

In the next section we will discuss how to deal with other boundary conditions.

6.4.2 Example

In this section we take a look at a simple example, namely the basis we get starting from the Haar wavelets. Remember that

$$\varphi(x) \;=\; \chi_{[0,1]}(x) \quad \text{and} \quad \psi(x) \;=\; \varphi(2x) \Leftrightarrow \varphi(2x \Leftrightarrow 1).$$

It immediately follows that both the operator wavelet and scaling functions are B-splines of order 2 (hat functions),

$$\Phi(x) \;=\; \Lambda(x) \quad \text{and} \quad \Psi(x) \;=\; \Lambda(2x).$$

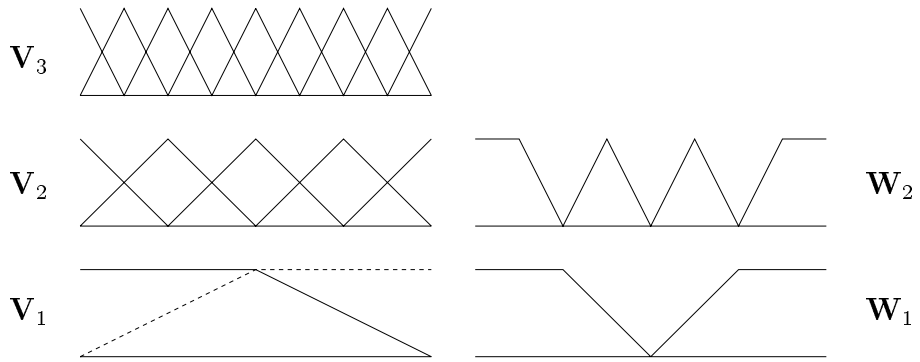

Figure 6.2: Basis for Neumann problem.

The original wavelets are orthogonal and as a consequence the basis functions and dual functions coincide.

The operator scaling functions satisfy the Strang-Fix condition with $N = 2$ and the convergence thus is of order h^2. One can prove that higher order wavelets with more vanishing moments (N) will, in general, not yield faster convergence because the solution u is not smooth enough. The underlying reason is that if $f \in \mathrm{L}_2([0,1])$, the solution u belongs to the Sobolev space $\mathrm{H}_{2,2}([0,1])$. One can get faster convergence only by imposing extra regularity conditions on the right-hand side. In a way this basis seems to be the most natural one to work with. Note that these piecewise linear basis functions are local solutions of the homogeneous equation. Hence the operator scaling functions and wavelets are $\mathcal{V}$-splines. This basis also coincides with Yserentant's hierarchical basis [199].

The idea to deal with boundary conditions is to let the operator wavelets satisfy the homogeneous boundary conditions and to let the component in the $\mathbf{V}_0$ space satisfy the imposed boundary conditions. Figure 6.1 shows the basis functions in the case of Dirichlet boundary conditions and $n = 3$. All operator wavelets vanish at the boundary. The coefficients of the two functions in the $\mathbf{V}_0$ space are determined by the boundary conditions. The fast wavelet transform differs from the periodic algorithm in the sense that different filter coefficients are used for the wavelets at the boundary. Note the "half hat" functions at the boundary.

The basis in case of the Neumann problem is shown in Figure 6.2. All operator wavelets have derivative zero at the boundary. The boundary conditions are handled by the two functions in the $\mathbf{V}_1$ space. Again the coefficient of the constant is undetermined. The integral of f is now equal to $u'(1) \Leftrightarrow u'(0)$.

6.4.3 The polyharmonic operator

The polyharmonic equation is defined as

$$\Leftrightarrow D^{2m} u \;=\; f,$$

and we take the square root operator to be

$$\mathcal{V} \;=\; D^m.$$

The operator scaling function Φ is now m times the convolution of the original scaling function φ with the box function, and the operator wavelet Ψ is m times the antiderivative of the original wavelet ψ. In order to get a compactly supported wavelet, the original wavelet needs to have at least m vanishing moments, a property that can be satisfied by all known wavelet families. The construction and algorithm are then completely similar to the case of the Laplace operator.

6.5 The Helmholtz operator

The one-dimensional Helmholtz operator is given by

$$\mathcal{L} \;=\; \Leftrightarrow D^2 + k^2$$

so that we can take

$$\mathcal{V} \;=\; D + k.$$

We assume that $k = 1$, which can always be obtained from a linear transformation. Observe that

$$\mathcal{V} \;=\; D + I \;=\; e^{-x}\, D\, e^{x}$$

and thus

$$\mathcal{V}^{-1} \;=\; e^{-x}\, D^{-1}\, e^{x}.$$

Applying $\mathcal{V}^{-1}$ to a wavelet does not necessarily yield a compactly supported function, since $e^x \psi_{j,k}$ does not have a vanishing integral. Therefore, we let $\Psi_{j,k} = \mathcal{V}^{-1} e^{-x} \psi_{j,k} = e^{-x} D^{-1} \psi_{j,k}$. If $\psi_{j,k}$ has a vanishing integral, then $\Psi_{j,k}$ is compactly supported.

In order to diagonalize the stiffness matrix, the original wavelets now need to be orthogonal with respect to a weighted inner product with weight function e^{-2x} because

$$\begin{aligned}
\langle\langle\, \Psi_{j,k}, \Psi^*_{j',k'} \,\rangle\rangle &= \langle\, \mathcal{V}\, \Psi_{j,k}, \mathcal{V}\, \Psi^*_{j',k'} \,\rangle \\
&= \langle\, e^{-x} \psi_{j,k}, e^{-x} \tilde{\psi}_{j',k'} \,\rangle \\
&= \int_{-\infty}^{+\infty} e^{-2x}\, \psi_{j,k}(x)\, \tilde{\psi}_{j',k'}(x)\, dx.
\end{aligned}$$

We see that the original wavelets need to be weighted wavelets. In this section we only use the unbalanced Haar wavelets as weighted wavelets. For a general treatment we refer to Section 6.6. The orthogonality of the unbalanced Haar wavelets on each level immediately follows from their disjoint support, since

$\text{supp}\,\psi_{j,k} = [2^{-j}k, 2^{-j}(k+1)]$. To get orthogonality between the different levels, $\mathbf{V}_j$ has to be orthogonal to $\mathbf{W}_{j'}$ for $j' \geqslant j$ or

$$\int_{-\infty}^{+\infty} e^{-2x}\,\varphi_{j,k}(x)\,\widetilde{\psi}_{j',k'}(x)\,dx \;=\; 0 \quad \text{for} \quad j' \geqslant j.$$

We let the scaling function coincide with e^{2x} on the support of the finer scale wavelets,

$$\varphi_{j,k} \;=\; e^{2x}\,\chi_{j,k},$$

where $\chi_{j,k}$ is the indicator function on the interval $[2^{-j}l, 2^{-j}(l+1)]$, normalized so that the integral of the scaling functions is a constant. We choose the wavelets as

$$\psi_{j,k} \;=\; \varphi_{j+1,2k} \Leftrightarrow \varphi_{j+1,2k+1},$$

so that they have a vanishing integral. The orthogonality between levels now follows from the fact that the scaling functions coincide with e^{2x} on the support of the finer scale wavelets, and from the vanishing integral of the wavelets

$$\begin{aligned}\int_{-\infty}^{+\infty} e^{-2x}\,\varphi_{j,k}(x)\,\widetilde{\psi}_{j',k'}(x)\,dx \;&=\; \int_{-\infty}^{+\infty} \chi_{j,k}(x)\,\widetilde{\psi}_{j',k'}(x)\,dx \\ &=\; C\int_{-\infty}^{+\infty} \widetilde{\psi}_{j',k'}(x)\,dx \;=\; 0.\end{aligned}$$

One can see that the operator wavelets are now piecewise hyperbolic functions (piecewise combinations of e^x and e^{-x}). The operator scaling functions are chosen as

$$\Phi_{j,k} \;=\; e^{-x}D^{-1}(\varphi_{j,k} \Leftrightarrow \varphi_{j,k+1}),$$

so that

$$\Psi_{j,k} \;=\; \Phi_{j+1,2k}.$$

With the right normalization, one gets

$$\Phi_{j,k}(x) \;=\; \begin{cases} \dfrac{\sinh(x \Leftrightarrow k2^{-j})}{\sinh(2^{-j})} & \text{for} \;\; x \in [k2^{-j}, (k+1)2^{-j}] \\ \dfrac{\sinh((k+2)2^{-j} \Leftrightarrow x)}{\sinh(2^{-j})} & \text{for} \;\; x \in [(k+1)2^{-j}, (k+2)2^{-j}] \\ 0 & \text{elsewhere.} \end{cases}$$

The operator scaling functions on one level are translates of each other but the ones on different levels are no longer dilates of each other. They are supported on exactly the same sets as the ones in Figure 6.1 and they roughly look similar. The operator scaling functions satisfy a refinement relation

$$\Phi_{j,k} \;=\; \sum_{l=0}^{2} H_{j,l}\,\Phi_{j+1,2k+l},$$

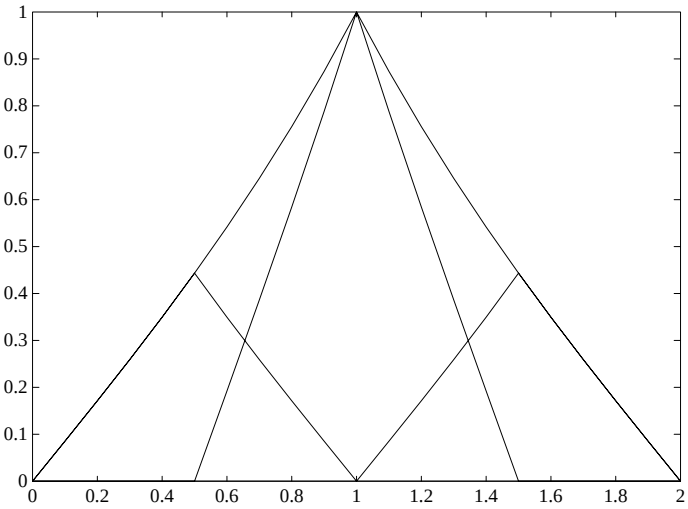

Figure 6.3: The refinement relation for the operator scaling functions.

with

$$H_{j,0} \;=\; H_{j,2} \;=\; \sinh(2^{-j-1})/\sinh(2^{-j}) \quad \text{and} \quad H_{j,1} \;=\; 1.$$

Figure 6.3 shows the refinement relation for the scaling functions. The three scaling functions on the finer scale are not the dilates of the one on the coarse scale, but they still add up to it.

The Helmholtz operator in the basis of operator wavelets again is diagonal and the algorithm is completely similar to the Laplace case. The only difference in implementation is that the filters in the fast wavelet transform now depend on the level.

Note that these functions again are $\mathcal{V}$-splines and, in a way, are the most natural to work with. Also note that

$$\lim_{j\to\infty} \Phi_{j,0}(2^{-j}x) \;=\; \Lambda(x).$$

Despite the fact that the Strang-Fix conditions are not satisfied, one can prove that the convergence is still of order h^2 [110].

We conclude that a wavelet transform can diagonalize constant coefficient operators similarly to the Fourier transform. The resulting algorithm is faster ($\mathcal{O}(M)$ instead of $\mathcal{O}(M \log M)$). This gain in speed is a consequence of the subsampling on the coarser levels in the wavelet transform (the ones that correspond to the low frequency components of the solution), which is not present in the Fourier transform.

6.6 Variable coefficients

Naturally, the next question is how to use wavelets for variable coefficient operators. Wavelets can diagonalize constant coefficient operators because of their locality in the frequency domain. We want to understand if we can exploit the localization in space to diagonalize variable coefficient operators. The answer is (perhaps quite surprisingly) yes, and this really justifies the use of wavelets for differential equations. No other technique (to our knowledge) has been able to accomplish this with a linear algorithm.

We take a closer look at the following operator

$$\mathcal{L} \;=\; \Leftrightarrow D\, p(x) D,$$

where p is sufficiently smooth and positive. The construction is based on the following observation:

$$\begin{aligned} \langle\langle\, f, g \,\rangle\rangle &= \langle \Leftrightarrow D\, p\, D\, f, g \,\rangle \\ &= \langle\, p\, D\, f, D\, g \,\rangle \\ &= \langle\, p\, D\, f, p\, D\, g \,\rangle_w\,, \end{aligned}$$

where the weight is taken to be

$$w \;=\; 1/p.$$

Now consider biorthogonal, compactly supported, weighted wavelets $\psi_{j,k}$ and $\widetilde{\psi}_{j,k}$. In Chapter 5 we showed how to construct such wavelets. Define the operator wavelets as

$$\Psi_{j,k} \;=\; D^{-1}\, \psi_{j,k}/p,$$

and similarly for the dual wavelets. The operator wavelets are compactly supported, since the weighted wavelets have at least one vanishing weighted moment (remember $w = 1/p$). It also follows that

$$\begin{aligned} \langle\langle\, \Psi_{j,k}, \Psi^*_{j',k'} \,\rangle\rangle &= \langle\, p\, D\, \Psi_{j,k}, p\, D\, \Psi^*_{j',k'} \,\rangle_w \\ &= \langle\, \psi_{j,k}, \widetilde{\psi}_{j',k'} \,\rangle_w \\ &= \delta_{j-j',k-k'}. \end{aligned}$$

Again we have diagonalization. The scaling functions now need to be chosen as

$$\Phi_{j,k} \;=\; D^{-1}\, (\varphi_{j,k} \Leftrightarrow \varphi_{j,k+1})/p.$$

This is a compactly supported function because the scaling functions have a constant weighted integral. The derivation of the operator fast wavelet transform now is straightforward.

In the case of the unbalanced Haar wavelets, the operator wavelets are piecewise functions where each piece looks like $AP + B$ with P the antiderivative of $1/p$. The operator wavelets again are $\mathcal{V}$-splines. Their support also coincides with the support of the functions of Figure 6.1, and they converge to hat functions as the level goes to infinity. The coefficients in the fast wavelet transform depend in a simple way on the Haar wavelet transform of $1/p$. The entries of the diagonal stiffness matrix can be calculated from the wavelet transform of $1/p$. The algorithm is completely similar to previous cases. Boundary conditions are as easy to handle as in the case of the Laplace operator. Note that the operator scaling functions do not satisfy the Strang-Fix conditions. It is possible to prove that the method has a convergence of order h^2. As mentioned earlier, higher convergence orders cannot be obtained in general.

6.7 Numerical example

We solve the equation

$$\Leftrightarrow D\, e^{x^2}\, D\, u(x) \;=\; e^{x^2} \left(\sin(x)(3x^2 \Leftrightarrow 2) + \cos(x)(2x \Leftrightarrow 2x^3)\right) / x^3,$$

with

$$u(0) = 1 \quad \text{and} \quad u(1) = \sin(1).$$

The exact solution is given by

$$u(x) \;=\; \sin(x)/x.$$

The L_∞ error of the numerically computed solution as a function of the number of levels (n) is shown in Table 6.1. Each time the number of levels is increased, the error is divided almost exactly by a factor of 4, which agrees with the $\mathcal{O}(h^2)$ convergence.

6.8 Conclusion and future research

In this chapter we showed how wavelets can be adapted to be useful in the solution of differential equations. Like the Fourier transform, wavelets can diagonalize constant coefficient operators. The resulting algorithm is slightly faster. The main result, however, is that even non-constant coefficient operators can be diagonalized with the right choice of basis. This evidently yields a much faster algorithm than the classic iterative methods.

An initial direction for future work is a careful analysis of when the operator wavelets provide unconditional bases for the Sobolev spaces. One can then use this to prove that $\bar{u}$ converges to the exact solution u as n goes to infinity.

The algorithm for the variable coefficient operators consists of first constructing the weighted wavelets using the subdivision scheme of the previous chapter and

Table 6.1: Error in function of level.

n	L_∞ error
1	1.22e-02
2	3.37e-03
3	8.66e-04
4	2.18e-04
5	5.45e-05
6	1.36e-05
7	3.41e-06
8	8.52e-07
9	2.13e-07

then taking the antiderivative in the proper way. It would be very useful to build a subdivision scheme that immediately yields the operator wavelets. This would be most helpful in cases where the operator cannot easily be split into explicit square root factors, e.g. when $\mathcal{L} = \Leftrightarrow D\, p\, D + q$, and in case $\mathcal{L}$ is not self-adjoint. Also, the subdivision scheme allows an easy incorporation of general boundary conditions.

Another idea is to use this technique for the solution of parabolic differential equations of the form

$$\partial u/\partial t = \mathcal{L}\, u + f,$$

using implicit time stepping discretizations. For example, a discretization in the time direction using the trapezoidal rule results in a Crank-Nicolson scheme. In every time step linear differential equations need to be solved. Using linearization, this can still be used when $\mathcal{L}$ is non-linear, e.g. Burgers' equation. One of the advantages is that it is very simple to work adaptively, i.e. only using particular wavelets on each level.

A third idea involves the study of the possible generalization of these ideas to partial differential equations. One of the problems here is how to split the operator. Already in the Laplace case it is not immediately clear what the right splitting is.

"Horace Walpole once said that the world is
comic to those who think
and tragic to those who feel."
—John Irving, *The World According to Garp (1976).*

Chapter 7

Data Compression with Smooth Local Trigonometric Functions

"To gild refined gold, to paint the lily,
To throw a perfume on the violet,
To smooth the ice, or add another hue
Unto the rainbow, or with taper light
To seek the beauteous eye of heaven to garnish,
Is wasteful and ridiculous excess."
—Shakespeare, *King John (1623).*

In this chapter we discuss bases formed by smooth local trigonometric functions. These can be seen as the Fourier transform of the wavelet or wavelet packets bases. Approximating of a function with smooth local trigonometric functions is thus equivalent with approximating its Fourier transform with wavelets. We are here mainly interested in applications to data compression. We present and compare two generalizations of the orthogonal basis of Coifman and Meyer: biorthogonal and equal parity folding bases. They have the advantage to allow an easy representation of constant and linear components. We show that they improve the error vs. compression ratio and reduce the blocking effect, a typical artifact in image compression.

7.1 Introduction

The general idea behind compression is to remove the redundancy in the data to find more compact representations. We consider lossy compression, i.e. we allow representations that, in some sense, are "close" to the original data to achieve higher compression ratios.

A popular method for lossy compression is so-called *transform coding*, i.e. represent the data in a different basis so that its coefficients show further decorrelation. The more decorrelated the coefficients, the more efficient the representation. An

approximation of the original data, which we refer to as the *compressed data*, can be obtained by retaining the coefficients with the highest information content and then performing the inverse transform. Karhunen and Loève showed that the optimal decorrelation is obtained by using the basis formed by the eigenvectors of the correlation matrix. However, the construction of this basis is computationally expensive. Moreover, in many situations the correlation matrix is not known.

The most commonly used transform is the Fourier transform or some variant of it. It has advantages such as analytic expressions for the basis functions, orthogonality and fast numerical algorithms. Its main disadvantage is that the basis functions (essentially sine and cosine functions) are non-local, while the correlation in the data often is local. In other words, the Fourier basis looks for correlation over the whole data set, which is usually small. For example, in an image, neighboring pixels usually are more correlated then ones that are far from each other.

We can solve this by splitting the data into parts and performing the Fourier transform on each part separately. Evidently, this leads to an orthogonal basis that is more local. We refer to this basis as the *local trigonometric basis*. It is the basic idea of JPEG, a standard still image compression algorithm [193]. The image here is divided into blocks of eight by eight pixels, after which the discrete cosine transform is used on each block.

The local trigonometric basis, however, has some disadvantages:

1. Fourier series are best suited for efficiently representing periodic data. Each part of the data is not necessarily periodic. This slows down the convergence and hinders high compression ratios.

2. Since each part is processed individually, the compressed data can reveal the splitting locations, cf. the blocking effect in JPEG.

3. Correlation among the parts is not exploited.

An improvement was proposed by Ronald Coifman and Yves Meyer in [49], and by Henrique Malvar in [138, 139]. The idea is to use smooth cutoff functions to split the data and to "fold" overlapping parts in a clever way so that the orthogonality is preserved. Moreover, the folded data are well-suited for representation by a trigonometric basis. We refer to such a basis as a *smooth local trigonometric basis*. This approach essentially addresses the first two disadvantages described above. An expository paper can be found in [17]. In [14], a connection between this basis and the Wilson basis of [68] was pointed out. Smooth local trigonometric bases were used successfully for image compression in [2, 3], where it is shown that blocking effect can be reduced significantly.

The third disadvantage can be resolved by using an adaptive algorithm where the splitting locations can depend on the data. An algorithm was presented by Ronald Coifman and Victor Wickerhauser in [54, 196]. An alternative was proposed in [84].

The basis of Coifman and Meyer has the disadvantage that the resolution of the constant is lost, i.e. on each interval the constant function is not a basis function. Consequently, representing smooth data becomes less efficient.

In this chapter we present two generalizations that preserve the resolution of the constant. The first one is based on a construction of so-called biorthogonal folding operators, while the second employs equal parity folding (EPF), an idea already suggested in [2]. We also show how to adapt the construction for bounded domains.

The chapter is organized as follows. In the first section we discuss trigonometric bases and their properties. Next (Section 7.3) we consider the basis of Coifman and Meyer. We present their construction from a different angle and with a different notation than in the original paper. This will facilitate the presentation of the biorthogonal construction in Section 7.5. Smooth local trigonometric functions are closely related to wavelets. We study this in detail in Section 7.4. In Section 7.6, we address the construction on an interval. Section 7.7 then contains a discussion of the EPF basis. Finally, we give some implementation issues and results.

7.2 Trigonometric bases

The classic trigonometric basis is the Fourier series basis. Consider the interval $I = [0, 1]$ for simplicity. The basis functions are given by

$$e_k(x) \;=\; \exp(i2\pi kx),$$

and we know that the set $\{e_k\}$ is an orthonormal basis for $\mathrm{L}_2([0, 1])$. The Fourier series of a function is given by

$$f \;=\; \sum_k c_k\, e_k, \qquad \text{with} \qquad c_k \;=\; \int_0^1 f(x)\, \overline{e_k(x)}\, dx.$$

The decay of the coefficients, and thus the convergence rate, depends on the smoothness of f when I is identified with the circle τ (i.e. the smoothness of the periodic extension of f). More precisely, if $f \in \mathrm{Lip}^\alpha(\tau)$, then [201]

$$c_k \;=\; \mathcal{O}(|k|^{-\alpha}).$$

However, the restriction of a smooth function defined on the real line to an interval is not necessarily a smooth function when extended periodically. This is due to the fact that the behavior of a function on the left end of the interval does not necessarily match the behavior on the right end. In that case the convergence is slow and the Gibbs phenomenon occurs. Basically this happens because the basis functions are all periodic, while the function we try to represent is not.

From the Fourier series, one can easily derive other orthonormal bases that only have sines or cosines as basis functions. One of them is the sine IV basis

where

$$s_k(x) \;=\; \sin\frac{2k+1}{2}\pi x,$$

and the set $\{\sqrt{2}\,s_k \mid k \in \mathbf{N}\}$ is an orthonormal basis for $\mathrm{L}_2([0,1])$. It is called the sine IV basis because it uses basis functions that have quarter wavelengths as compared to the Fourier series. The functions s_k all are odd and smooth around the left endpoint and even and smooth around the right endpoint. This means that the convergence is fast whenever the function, extended as an odd function around the left endpoint and as an even function around the right endpoint, is smooth. (By extending as an even (resp. odd) function we mean applying the operator $1+\mathcal{M}$ (resp. $1 \Leftrightarrow \mathcal{M}$), see below.) Other bases and their parity are: sine II (odd and odd),

$$S_k(x) \;=\; \sin k\pi x,$$

cosine II (even and even),

$$C_k(x) \;=\; \cos k\pi x,$$

and cosine IV (even and odd),

$$c_k(x) \;=\; \cos\frac{2k+1}{2}\pi x.$$

For each basis a discrete transform and a fast (linear) algorithm, inspired on the Fast Fourier Transform (FFT), exist, see [149, 192].

Whenever the data shows some special behavior such as periodicity or parity around an endpoint, it is important to pick the basis that reflects this property in order to obtain rapid decay of the coefficients. Note that this is the key to compression as it implies that only few coefficients are needed to represent the data within a certain accuracy.

7.3 Local trigonometric bases of Coifman and Meyer

7.3.1 The folding operator

The *mirror operator* $\mathcal{M}_\alpha$ around a point α is defined as

$$\mathcal{M}_\alpha\, f(x) \;=\; f(2\alpha \Leftrightarrow x).$$

It essentially flips the function around α. Note that it is self-adjoint and unitary. Consider an interval of length $2\epsilon_\alpha$ around α and a continuous cutoff function l so that

$$l_\alpha(x) \;=\; \begin{cases} 1 & \text{if} \quad x < \alpha \Leftrightarrow \epsilon_\alpha \\ 0 & \text{if} \quad x > \alpha + \epsilon_\alpha, \end{cases}$$

and let $r_\alpha = \mathcal{M}_\alpha\, l_\alpha$. This allows us to define the folding operator.

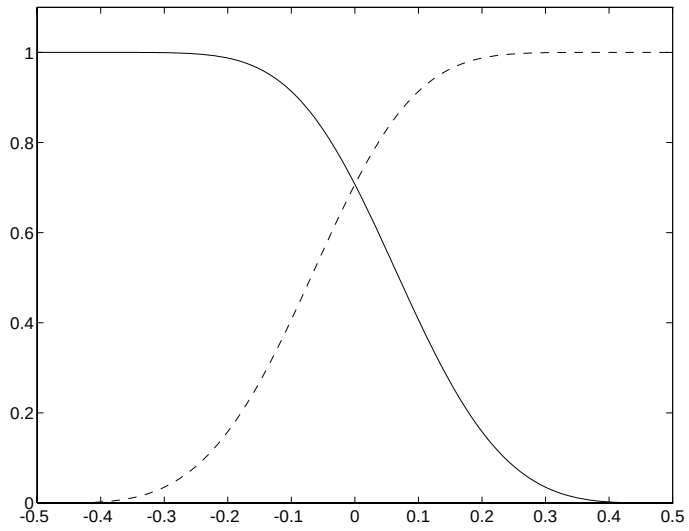

Figure 7.1: The cutoff functions l_0 and r_0 (dashed) with $\epsilon_0 = 1/2$.

Definition 7.1 *The* folding operator *around a point* α *is defined as*

$$\mathcal{F}_\alpha \;=\; \chi^l_\alpha\,(1+\mathcal{M}_\alpha)\,l_\alpha + \chi^r_\alpha\,(1 \Leftrightarrow \mathcal{M}_\alpha)\,r_\alpha,$$

where $\chi^l_\alpha = \chi_{(-\infty,\alpha]}$ *and* $\chi^r_\alpha = \mathcal{M}_\alpha\,\chi^l_\alpha$.

One can check that the adjoint of the folding operator is given by,

$$\begin{aligned}\mathcal{F}^*_\alpha &= l_\alpha\,(1+\mathcal{M}_\alpha)\,\chi^l_\alpha + r_\alpha\,(1 \Leftrightarrow \mathcal{M}_\alpha)\,\chi^r_\alpha \\ &= \chi^l_\alpha\,(1 \Leftrightarrow \mathcal{M}_\alpha)\,l_\alpha + \chi^r_\alpha\,(1+\mathcal{M}_\alpha)\,r_\alpha.\end{aligned}$$

Lemma 7.2 *The folding operator is unitary if and only if* $l^2_\alpha + r^2_\alpha \;=\; 1$.

Proof : Follows from the fact that

$$\begin{aligned}\mathcal{F}^*_\alpha\,\mathcal{F}_\alpha &= l_\alpha\,(1+\mathcal{M}_\alpha)\,\chi^l_\alpha\,(1+\mathcal{M}_\alpha)\,l_\alpha + r_\alpha\,(1 \Leftrightarrow \mathcal{M}_\alpha)\,\chi^r_\alpha\,(1 \Leftrightarrow \mathcal{M}_\alpha)\,r_\alpha \\ &= l^2_\alpha\,\chi^l_\alpha + l^2_\alpha\,\chi^r_\alpha + l_\alpha\,\chi^r_\alpha\,r_\alpha\mathcal{M}_\alpha + l_\alpha\,\chi^l_\alpha\,r_\alpha\mathcal{M}_\alpha \\ &\quad + r^2_\alpha\,\chi^r_\alpha + r^2_\alpha\,\chi^l_\alpha \Leftrightarrow l_\alpha\,\chi^r_\alpha\,r_\alpha\mathcal{M}_\alpha \Leftrightarrow l_\alpha\,\chi^l_\alpha\,r_\alpha\mathcal{M}_\alpha \\ &= l^2_\alpha + r^2_\alpha \;=\; \mathcal{F}_\alpha\,\mathcal{F}^*_\alpha.\end{aligned}$$

□

Figure 7.1 shows an example of cutoff functions that satisfy this condition. For the remainder of this section we assume it is satisfied.

Let us discuss what the action of the folding operator is. Multiplication with l_α lets the function die off smoothly to the left of $\alpha + \epsilon_\alpha$. The operator $1 + \mathcal{M}_\alpha$

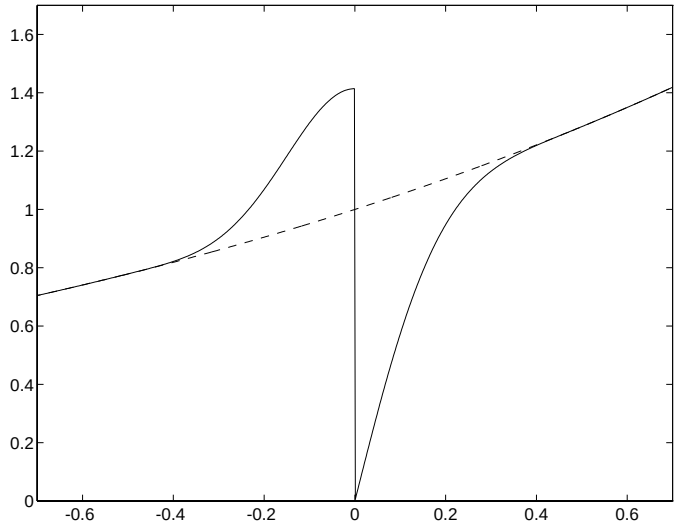

Figure 7.2: The folding of a function where $\alpha = 0$ and $\epsilon_\alpha = 0.5$.

then adds the function and its mirrored version. This results in a function even around α, which is then cut off by χ^l_α. The right part is similar and creates an odd function. Consequently, if f is smooth, then $\chi^l_\alpha \mathcal{F}_\alpha f$ is a function that is smooth when extended "even" to the right and $\chi^r_\alpha \mathcal{F}_\alpha f$ is a function that is smooth when extended "odd" to the left. Note that even when f is smooth, $\mathcal{F}_\alpha f$, in general, is discontinuous at α. The adjoint (or inverse) operator does exactly the same but switches even and odd. Figure 7.2 shows the folding of an exponential function around $\alpha = 0$.

7.3.2 The total folding operator

Consider a partition of the real line into a countable set of intervals $I = (\alpha, \beta]$, so that

$$\mathbf{R} \;=\; \bigcup_I \chi_I,$$

and

$$\beta \Leftrightarrow \alpha \;\geqslant\; \epsilon_\alpha + \epsilon_\beta.$$

We can use I, α and β as indices since they belong to countable sets. The operators $\mathcal{F}_\alpha$ and $\mathcal{F}_\beta$ commute because $\mathcal{F}_\alpha = 1$ on $\mathbf{R} \setminus (\alpha \Leftrightarrow \epsilon_\alpha, \alpha + \epsilon_\alpha)$. This allows the following definition.

Definition 7.3 *The total folding operator is defined as*

$$\mathcal{T} \;=\; \prod_\alpha \mathcal{F}_\alpha.$$

Being a product of unitary operators, it too is unitary.

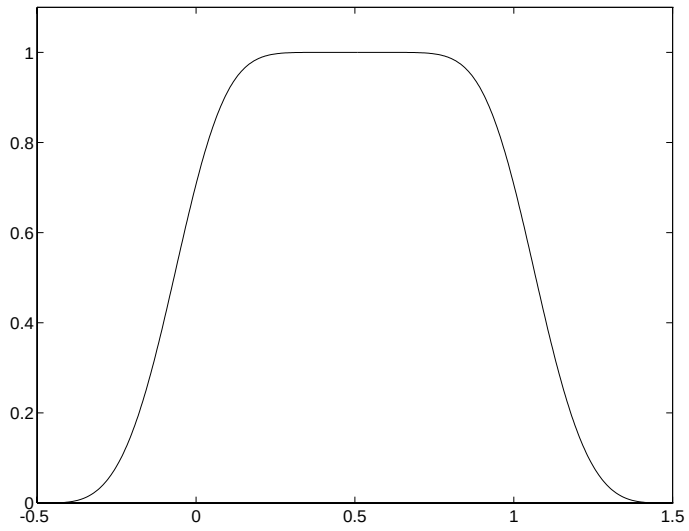

Figure 7.3: The bell function for $(1, 3]$.

We write the total folding operator as a sum of operators $\mathcal{G}_I$, each valid with one interval I, or

$$\mathcal{T} \;=\; \sum_I \chi_I \, \mathcal{G}_I .$$

One can understand that $\mathcal{G}_I$ is given by

$$\mathcal{G}_I \;=\; (1 \Leftrightarrow \mathcal{M}_\alpha + \mathcal{M}_\beta)\, b_I ,$$

where b_I is the *bell function* associated with the interval I,

$$b_I \;=\; r_\alpha \, l_\beta .$$

This follows from the fact that in both representations,

$$\mathcal{T} \;=\; \begin{cases} 1 & \text{on} \quad (\alpha + \epsilon_\alpha, \beta \Leftrightarrow \epsilon_\beta) \\ (1 \Leftrightarrow \mathcal{M}_\alpha)\, r_\alpha & \text{on} \quad (\alpha, \alpha + \epsilon_\alpha] \\ (1 + \mathcal{M}_\beta)\, l_\beta & \text{on} \quad [\beta \Leftrightarrow \epsilon_\beta, \beta]. \end{cases}$$

Note that

$$\mathcal{G}_I^* \;=\; b_I \, (1 \Leftrightarrow \mathcal{M}_\alpha + \mathcal{M}_\beta)$$

and

$$\sum_I b_I^2 \;=\; 1.$$

Figure 7.3 shows an example of the bell function for $I = (1, 3]$ and $\epsilon_0 = \epsilon_1 = 1$. We next study the properties of $\mathcal{G}_I$.

Definition 7.4 *A function f is* locally even (resp. odd) *around a point α if $f(x) = \mathcal{M}_\alpha f(x)$ (resp. $\Leftrightarrow \mathcal{M}_\alpha f(x)$) for $x \in [\alpha \Leftrightarrow \epsilon_\alpha, \alpha + \epsilon_\alpha]$.*

Lemma 7.5 *The function $\mathcal{G}_I\, f$ is locally odd around α and locally even around β.*

Proof : On the interval $[\alpha \Leftrightarrow \epsilon_\alpha, \alpha + \epsilon_\alpha]$, $\mathcal{G}_I\, f = (1 \Leftrightarrow \mathcal{M}_\alpha)\, r_\alpha\, f$, so that $\mathcal{M}_\alpha\, \mathcal{G}_I\, f = \Leftrightarrow \mathcal{G}_I\, f$. □

Lemma 7.6 *If a function s is locally odd around α and locally even around β, then*

$$\chi_I\, \mathcal{G}_I\, b_I\, s \;=\; \chi_I\, s \quad \textit{and} \quad \mathcal{G}_I^*\, \chi_I\, s \;=\; b_I\, s.$$

Proof : We prove the first equation. On the interval $(\alpha + \epsilon_\alpha, \beta \Leftrightarrow \epsilon_\beta)$, left and right-hand side are equal to s. On the interval $(\alpha, \alpha + \epsilon_\alpha]$, the left-hand side is equal to $(1 \Leftrightarrow \mathcal{M}_\alpha)\, r_\alpha^2\, s$, which is equal to s because s is locally odd and $r_\alpha^2 + l_\alpha^2 = 1$. The right side and the second equation are proven similarly. □

7.3.3 Splitting into subspaces

The idea is to split $\mathrm{L}_2(\mathbf{R})$ into subspaces so that each subspace contains functions localized around one of the intervals I. Moreover, we want a basis that is suited for representation of smooth functions on that interval. The easiest would be to let

$$\mathrm{L}_2(\mathbf{R}) \;=\; \bigoplus_I \mathrm{L}_2(I).$$

This obviously is an orthogonal decomposition and χ_I is the orthogonal projection associated with it. Unfortunately, the trigonometric basis on each interval is not suited for representations of smooth functions, cf. the discussion in Section 7.2. But, applying the total folding operator to a smooth function results in a function with specific parity properties at the endpoints of each interval and we then can use the right trigonometric basis. The orthogonality is preserved because the total folding operator is unitary. The orthogonal projection operator associated with an interval is given by

$$\mathcal{P}_I \;=\; \mathcal{T}^*\, \chi_I\, \mathcal{T}.$$

We decompose $\mathrm{L}_2(\mathbf{R})$ into orthogonal subspaces as

$$\mathrm{L}_2(\mathbf{R}) \;=\; \bigoplus_I V_I \quad \text{with} \quad V_I \;=\; \mathcal{P}_I\, \mathrm{L}_2(\mathbf{R}).$$

It is immediately clear that

$$\mathcal{T}\, V_I \;=\; \mathrm{L}_2(I) \quad \text{and} \quad \mathcal{T}^*\, \mathrm{L}_2(I) \;=\; V_I.$$

So the total folding operator maps one subspace splitting into the other. If we use folding operators associated with an interval, we see that the projection operator also can be written as

$$\mathcal{P}_I \;=\; \mathcal{G}_I^*\, \chi_I\, \mathcal{G}_I.$$

Lemma 7.7 *Each element of V_I is of the form $b_I\, s$, where s is locally odd around α and locally even around β, and every function of this form belongs to V_I.*

Proof : The first part follows from the fact that

$$\mathcal{P}_I \;=\; b_I\, \mathcal{G}_I,$$

which is a consequence of Lemmas 7.5 and 7.6.

The second part follows from the fact that

$$\mathcal{P}_I\, b_I\, s \;=\; \mathcal{G}_I^*\, \chi_I\, \mathcal{G}_I\, b_I\, s \;=\; b_I\, s,$$

which is a consequence of Lemma 7.6. □

The fact that the projection operators are orthogonal can also be understood as follows. Let I and J be two intervals. In case they are not neighbors, the supports of $\mathcal{P}_I f$ and $\mathcal{P}_J g$ do not intersect. In the case that I and J meet at a point α, $\mathcal{P}_I f\, \mathcal{P}_J g$ is only supported on $[\alpha \Leftrightarrow \epsilon_\alpha, \alpha + \epsilon_\alpha]$, where it is equal to $b_I\, s\, b_J\, t = l_\alpha\, r_\alpha\, s\, t$, where s is locally odd and t is locally even around α. As $l_\alpha\, r_\alpha$ is locally even, the integral vanishes.

The previous lemma tells us which is the right trigonometric basis to use. The orthonormal basis for $\mathrm{L}_2(I)$ that matches the parity is given by

$$\chi_I\, s_{I,k},$$

where

$$s_{I,k} \;=\; \sqrt{\frac{2}{|I|}}\, \sin \frac{2k+1}{2}\frac{\pi}{|I|}(x \Leftrightarrow \alpha).$$

This immediately corresponds to an orthogonal basis for V_I given by

$$\mathcal{T}^*\, \chi_I\, s_{I,k} \;=\; \mathcal{G}_I^*\, \chi_I\, s_{I,k} \;=\; b_I\, s_{I,k}, \qquad \text{with} \qquad k \in \mathbf{N}.$$

Consequently,

$$f \;=\; \sum_I \mathcal{P}_I f \;=\; \sum_{I,k} c_{I,k}\, b_I\, s_{I,k},$$

where the coefficients are given by

$$c_{I,k} \;=\; \langle\, f, b_I\, s_{I,k}\,\rangle \;=\; \langle\, \mathcal{T} f, \chi_I\, s_{I,k}\,\rangle\,. \tag{7.1}$$

If we look at the operators from a practical point of view, we see that $\mathcal{F}_\alpha$ (and thus $\mathcal{T}$), χ_I and their adjoints are easy to discretize and implement. This means we always use the second expression of (7.1) for the coefficients of a function. Also the machinery for local trigonometric bases on an interval then becomes readily available.

We can summarize the results from this section in the following theorem.

Theorem 7.8 *With the assumptions made in this section, the functions $\{b_I\, s_{I,k}\}$ form an orthogonal basis for L_2. Moreover, if f and the cutoff functions belong to Lip^α, the coefficients of f decay as*

$$c_{I,k} \;=\; \mathcal{O}(k^{-\alpha}).$$

7.4 Connection with wavelets

There is a close connection between local trigonometric functions and wavelets. To understand this, we take a look at the following example. Consider the multiresolution analysis formed by the Shannon wavelet. Since

$$\widehat{\psi}(\omega) \;=\; \chi_I(\omega) \quad \text{with} \quad I = [\Leftrightarrow 2\pi, \Leftrightarrow \pi] \cup [\pi, 2\pi],$$

we see that

$$W_j \;=\; \{f \in \mathrm{L}_2 \mid \operatorname{supp} \widehat{f} \subset 2^j\, I\}.$$

The splitting of L_2 into the wavelet spaces thus corresponds on to splitting the frequency axis into logarithmic intervals and letting

$$\mathrm{L}_2 \;=\; \bigcup_j \mathrm{L}_2(2^j\, I).$$

On each interval we use the Fourier series basis as,

$$\widehat{\psi}_{j,l} \;=\; \sqrt{2^{\Leftrightarrow j}} e^{\Leftrightarrow i\omega l 2^{\Leftrightarrow j}}\, \chi_{2^j\, I}.$$

These wavelets have slow decay. Remember that decay in the spatial domain corresponds to smoothness in the frequency domain. side. Indeed, $\widehat{\psi}_{j,l}$ is not continuous. We see that the $\widehat{\psi}_{j,l}$ form a local trigonometric basis.

It immediately follows that using the *smooth* local trigonometric basis on these intervals of the frequency domain leads to wavelets with rapid decay in the spatial domain. In case the cutoff functions belong to $\mathcal{C}^\infty$, the wavelets have faster than polynomial decay. The Meyer wavelet was constructed this way, and its generalization was the motivation for the work of Coifman and Meyer. These wavelets have an infinite number of vanishing moments since their Fourier transform vanishes in a neighborhood of the origin.

Note that a splitting into intervals of equal size corresponds to a certain set of wavelet packets. We can say that wavelet bases and smooth local trigonometric bases are two instances of the same idea.

7.5 Biorthogonal local trigonometric bases

An important property of a transform coding scheme is the so-called *resolution of the constant*, which concerns how constant functions are represented in the basis. We say that a basis has a resolution of the constant if, on a finite domain, the constant is represented with a finite number of basis functions. Since a smooth function locally resembles a constant, it is important to represent a constant on each interval with as few coefficients as possible (preferably only one). We adapt

the construction so that the constant is one of the basis functions. From the previous section we see that,

$$\chi_I \, \mathcal{G}_I \, 1 \;=\; \begin{cases} r_\alpha \Leftrightarrow l_\alpha & \text{on} \quad (\alpha, \alpha + \epsilon_\alpha] \\ 1 & \text{on} \quad (\alpha + \epsilon_\alpha, \beta \Leftrightarrow \epsilon_\beta) \\ r_\beta + l_\beta & \text{on} \quad [\beta \Leftrightarrow \epsilon_\beta, \beta]. \end{cases}$$

We want cutoff functions so that this function coincides with the first basis function $s_{I,0}$. This is evidently only possible if $\alpha + \epsilon_\alpha = \beta \Leftrightarrow \epsilon_\beta$. We therefore from now on only work with intervals of equal size and let $\epsilon = |I|/2$. In order to achieve a resolution of the constant and still have an orthogonal basis, the cutoff function needs to be chosen as $l_\alpha = l((x \Leftrightarrow \alpha)/\epsilon)$, where

$$l(x) \;=\; \begin{array}{ll} 1 & \text{for} \quad x < \Leftrightarrow 1 \\ \dfrac{\cos(\pi x/4) \Leftrightarrow \sin(\pi x/4)}{\sqrt{2}} & \text{for} \quad x \in [\Leftrightarrow 1, 1] \\ 0 & \text{for} \quad 1 < x. \end{array}$$

This cutoff function is continuous but it not differentiable. Consequently, the folding operator introduces discontinuities in derivatives that slow down convergence and thus hinder compression. What we need is a resolution of the constant with smoother cutoff functions.

In order to solve this problem, we add some flexibility to the construction, by abandoning the orthogonality requirement. In the remainder we let l_α and $r_\alpha = \mathcal{M}_\alpha l_\alpha$ be continuous cutoff functions that do not necessarily satisfy $l_\alpha^2 + r_\alpha^2 = 1$.

Lemma 7.9 *In case $l_\alpha^2 + r_\alpha^2$ is bounded from below and above, the folding operator is bounded and invertible. More precisely,*

$$A \, \|f\| \;\leqslant\; \|\mathcal{F}_\alpha f\| \;\leqslant\; B \, \|f\|,$$

where

$$A \;=\; \min_x \sqrt{l_\alpha^2 + r_\alpha^2} \quad \textit{and} \quad B \;=\; \max_x \sqrt{l_\alpha^2 + r_\alpha^2},$$

and these constants are sharp.

Proof : Follows from the fact that

$$\int_{-\infty}^{+\infty} (\mathcal{F}_\alpha f)^2 \, dx \;=\; \int_{-\infty}^{+\infty} (l_\alpha^2 + r_\alpha^2) \, f^2 \, dx,$$

which is the result from simple algebraic manipulations similar to the ones in the previous section. □

Lemma 7.10 *The inverse of an invertible folding operator $\mathcal{F}_\alpha$ again is a folding operator.*

Proof : The equation $g = \mathcal{F}_\alpha f$ can be written in matrix form as

$$\chi_\alpha^l \begin{bmatrix} g \\ \mathcal{M}_\alpha\, g \end{bmatrix} \;=\; \begin{bmatrix} l_\alpha & r_\alpha \\ \Leftrightarrow r_\alpha & l_\alpha \end{bmatrix} \chi_\alpha^l \begin{bmatrix} f \\ \mathcal{M}_\alpha\, f \end{bmatrix}.$$

From this we see immediately that the inverse of $\mathcal{F}_\alpha$ is given by $\tilde{\mathcal{F}}_\alpha^*$, where

$$\tilde{\mathcal{F}}_\alpha \;=\; \chi_\alpha^l\,(1+\mathcal{M}_\alpha)\,\tilde{l}_\alpha + \chi_\alpha^r\,(1\Leftrightarrow\mathcal{M}_\alpha)\,\tilde{r}_\alpha,$$

$$\tilde{l}_\alpha \;=\; \frac{l_\alpha}{l_\alpha^2 + r_\alpha^2}, \quad \text{and} \quad \tilde{r}_\alpha \;=\; \frac{r_\alpha}{l_\alpha^2 + r_\alpha^2}.$$

□

Note that

$$l_\alpha\,\tilde{l}_\alpha + r_\alpha\,\tilde{r}_\alpha \;=\; 1.$$

We call $\mathcal{F}_\alpha$ and $\widetilde{\mathcal{F}_\alpha}$ *biorthogonal folding operators* and more specifically refer to $\widetilde{\mathcal{F}_\alpha}$ as the *dual folding operator*.

This does not solve the problem completely. Indeed, if the cutoff functions belong to $\mathcal{C}^1$, the folded constant has derivative zero at $\alpha + \epsilon$ and never coincides with the first basis function $s_{I,0}$. We therefore generalize the construction further by allowing different parities. We want to have a folding operator that takes a smooth function into a function that is either odd at the left *and* right endpoint of an interval or even at both endpoints. One way to do so would be to use folding operators with the same parity left and right of the folding point. Unfortunately, these operators are not invertible.

Lemma 7.11 *A folding operator with the same parity left and right of the folding point is not invertible.*

Proof : Consider the twice even case and let

$$\mathcal{F}_\alpha \;=\; \chi_\alpha^l\,(1+\mathcal{M}_\alpha)\,l_\alpha + \chi_\alpha^r\,(1+\mathcal{M}_\alpha)\,r_\alpha.$$

The matrix representation leads to a matrix with determinant $l_\alpha^2 \Leftrightarrow r_\alpha^2$, which vanishes at the folding point. □

This basically tells us that the only way to get the same parity at both endpoints of an interval is to alternate the parity of folding operators. In other words, we need to change the parity from ... (odd even) (odd even) (odd even) ... to ... (even even) (odd odd) (even even) (odd odd) .... This implies defining the total folding operator by alternating $\mathcal{F}_\alpha$ and $\mathcal{F}_\alpha^*$. In the intervals with (even even) parity we use the cosine II basis and in the intervals with (odd odd) parity the sine II basis. To write this down in more detail we let

$$\alpha_l \;=\; l\,|I|,$$

and, in order to simplify notation, we replace every subscript α_l or I by the integer subscript l. Thus $\chi_l = \chi_{(l|I|,(l+1)|I|]}$. The total folding operator is given by

$$\mathcal{T} = \prod_l \mathcal{F}_{2l}\, \mathcal{F}^*_{2l+1},$$

where the factors commute. Again this operator is invertible, and since the individual folding operators do not interact spatially, it also satisfies

$$A\,\|f\| \leqslant \|\mathcal{T} f\| \leqslant B\,\|f\|,$$

with the same constants as above. The dual total folding operator is defined similarly (just add the tildes) and again $\mathcal{T}^{-1} = \widetilde{\mathcal{T}}^*$. The condition number of the total folding operator is B/A.

Following a reasoning similar to the orthogonal case, we understand that the total folding operator can also be written as a sum of folding operators associated with an interval,

$$\mathcal{T} = \sum_l \chi_l\, \mathcal{G}_l,$$

where

$$\mathcal{G}_{2l} = (1 \Leftrightarrow \mathcal{M}_\alpha \Leftrightarrow \mathcal{M}_\beta)\, b_{2l},$$

and

$$\mathcal{G}_{2l+1} = (1 + \mathcal{M}_\alpha + \mathcal{M}_\beta)\, b_{2l+1}.$$

We define the operators $\mathcal{P}_l$ and the subspaces V_l similar to the orthogonal case. The projection operator associated with an interval is given by

$$\mathcal{P}_l = \widetilde{\mathcal{T}}^*\, \chi_l\, \mathcal{T}.$$

We decompose $\mathrm{L}_2(\mathbf{R})$ into complementary subspaces as

$$\mathrm{L}_2(\mathbf{R}) = \bigoplus_l V_l \quad \text{with} \quad V_l = \mathcal{P}_l\, \mathrm{L}_2(\mathbf{R}),$$

where

$$\mathcal{T}\, V_l = \mathrm{L}_2(I) \quad \text{and} \quad \widetilde{\mathcal{T}}^*\, \mathrm{L}_2(I) = V_l.$$

Again the projection operator also can be written as

$$\mathcal{P}_l = \widetilde{\mathcal{G}}^*_l\, \chi_l\, \mathcal{G}_l = \tilde{b}_l\, \mathcal{G}_l.$$

If l is odd (resp. even), an element of V_l can be written as $\tilde{b}_l$ times a function that is locally even (resp. odd) around α and β.

We use the basis functions with the right parity on each interval:

$$t_{2l,k} = \sqrt{\frac{2}{|I|}} \sin(k+1) \frac{\pi}{|I|} (x \Leftrightarrow 2l) \quad \text{for} \quad k \geqslant 0,$$

$$t_{2l+1,k} = \sqrt{\frac{2}{|I|}} \cos k \frac{\pi}{|I|} (x \Leftrightarrow 2l \Leftrightarrow 1) \quad \text{for} \quad k \geqslant 1,$$

and

$$t_{2l+1,0} \;=\; \frac{1}{\sqrt{|I|}}.$$

Obviously, the $t_{l,k}$ with $l \in \mathbf{Z}$ and $k \in \mathbf{N}$ form an orthonormal basis for $\mathrm{L}_2(\mathbf{R})$. This implies that the basis formed by the $\widetilde{\mathcal{T}}^*\,\chi_l\,t_{l,k}$ is a Riesz basis for $\mathrm{L}_2(\mathbf{R})$. These functions are given by

$$\widetilde{\mathcal{T}}^*\,\chi_l\,t_{l,k} \;=\; \widetilde{\mathcal{G}}_l^*\,\chi_l\,t_{l,k} \;=\; \tilde{b}_l\,t_{l,k}.$$

Consequently,

$$f \;=\; \sum_l \mathcal{P}_l f \;=\; \sum_{l,k} c_{l,k}\,\tilde{b}_l\,t_{l,k},$$

where the coefficients are given by

$$c_{l,k} \;=\; \langle\, \mathcal{T} f, \chi_l\,t_{l,k}\,\rangle \;=\; \langle\, f, \mathcal{T}^*\,\chi_l\,t_{l,k}\,\rangle \;=\; \langle\, f, b_l\,t_{l,k}\,\rangle\,, \tag{7.2}$$

and

$$A\,\|f\| \;\leqslant\; \sqrt{\sum_{l,k} c_{l,k}^2} \;\leqslant\; B\,\|f\|,$$

with the same constants as above. We say that $\{b_l\,t_{l,k}\}$ is the dual basis corresponding to the basis $\{\tilde{b}_l\,t_{l,k}\}$. The first expression of (7.2) for the coefficients is the easiest to implement. We summarize the results in a theorem.

Theorem 7.12 *The sets of functions $\{b_l\,t_{l,k}\}$ and $\{\tilde{b}_l\,t_{l,k}\}$ both are Riesz bases of L_2. Moreover, they are biorthogonal in the sense that*

$$\langle\, b_l\,t_{l,k}, \tilde{b}_{l'}\,t_{l',k'}\,\rangle \;=\; \delta_{l-l'}\,\delta_{k-k'}.$$

The coefficients of a Lipschitz continuous function have the typical rapid rate of decay. More precisely, if f and the cutoff functions belong to Lip^α, then the coefficients of f decay as

$$c_{l,k} \;=\; \mathcal{O}(k^{-\alpha}).$$

The only thing left is to find cutoff functions so that $\chi_l\,\mathcal{T}\,1$ coincides (up to a constant factor) with $\chi_l\,t_{l,0}$. This can be done by letting $l_\alpha = l((x \Leftrightarrow \alpha)/\epsilon_\alpha)$ where

$$l(x) \;=\; \frac{1 \Leftrightarrow \sin(\pi x/2)}{2} \quad \text{for} \quad x \in [\Leftrightarrow 1, 1].$$

This cutoff function belongs to $\mathcal{C}^1$. It is easy to check that on I,

$$\mathcal{G}_{2l}\,1 \;=\; \sin\left(\frac{x \Leftrightarrow 2l}{|I|}\right) \quad \text{and} \quad \mathcal{G}_{2l+1}\,1 \;=\; 1.$$

In this case the constants A and B used in the comparison of norms are $1/\sqrt{2}$ and 1 respectively. The condition number of the folding operator is thus $\sqrt{2}$. We

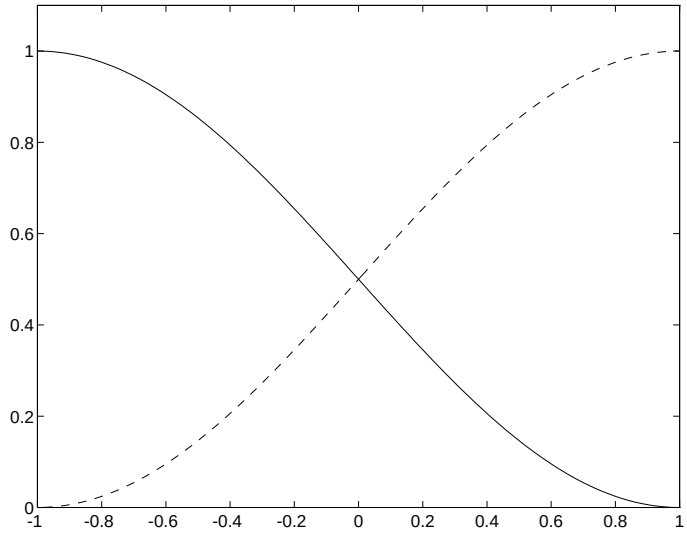

Figure 7.4: The biorthogonal cutoff functions.

have chosen this normalization because it is natural for the cutoff functions to have value $1/2$ at α. This means that folding a function does not make it grow on the even side of a folding point. The cutoff functions are shown in Figure 7.4 and the biorthogonal total folding of an exponential function is shown in Figure 7.5. The folded function on each interval closely resembles the first basis function. Figure 7.6 shows two basis functions, $\widetilde{b}_2\, t_{2,4}$ and $\widetilde{b}_5\, t_{5,9}$. The shape of the bell functions is drawn dotted. Note the parity of the basis functions at the endpoints.

7.6 Folding operators on an interval

So far the discussion has only concerned functions defined on the real line. In this section, we focus on folding operators on an interval. Since we can treat each boundary point independently we consider the case of the interval $I = [\alpha, \infty)$. We introduce an extension operator $\mathcal{X}$ that takes a function defined on the interval to a function defined on the whole real line, and a restriction operator $\mathcal{R}$ that does the opposite. We want them to satisfy $\mathcal{X} = \mathbf{1}$ on I, and $\mathcal{R}\,\mathcal{X} = \mathbf{1}$. Also, in case f is smooth, we want $\mathcal{X} f$ to have some smoothness too. For notational simplicity, we omit the subscript α and introduce a superscript b for cutoff functions and operators associated with the boundary of the interval. We define the folding operator at the boundary as

$$\mathcal{F}^b \;=\; \mathcal{R}\,\mathcal{F}\,\mathcal{X}.$$

Let $l^b = \mathcal{R}\, l$ (similarly for r), and let $\mathcal{M}^b$ be a mirror operator that maps a function defined on I to a function defined on $\mathbf{R} \setminus I$. One can understand that

$$\mathcal{F}^b f \;=\; r^b\, f \Leftrightarrow l^b\, \mathcal{R}\, \mathcal{M}\, \mathcal{X}\, f.$$

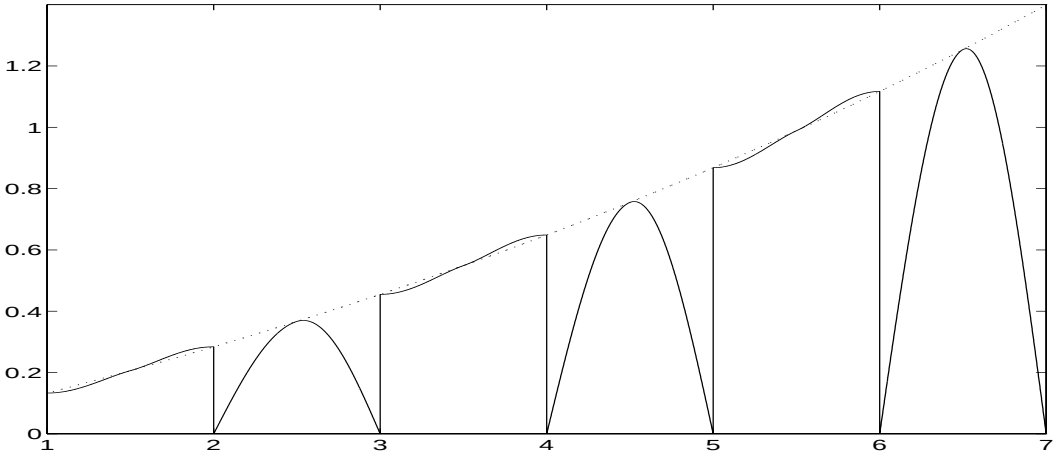

Figure 7.5: The biorthogonal folding.

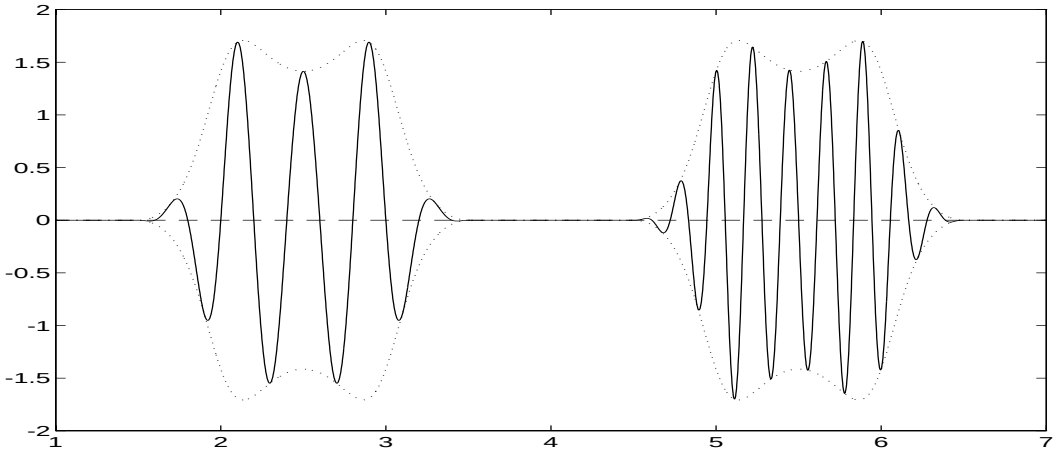

Figure 7.6: Biorthogonal basis functions.

The second term has a plus sign in the case of $\mathcal{F}^{*b} = \mathcal{R}\,\mathcal{F}^*\,\mathcal{X}$ (i.e. the even case).

Assuming that f is continuous, we propose to choose the extension operator in the odd case as

$$\mathcal{X}\,f \;=\; 2f(\alpha) \Leftrightarrow \mathcal{M}^b\,f \quad \text{on} \quad \mathbf{R} \setminus I.$$

This guarantees that if $f \in \mathcal{C}^1$, so is $\mathcal{X}\,f$. The folding operator is given by

$$\mathcal{F}^b\,f \;=\; (l^b + r^b)\,f \Leftrightarrow 2f(\alpha)\,l^b.$$

We can retrieve f from $\mathcal{F}^b\,f$ by

$$f \;=\; \frac{\mathcal{F}^b\,f + 2f(\alpha)\,l^b}{l^b + r^b}.$$

It also shows that we need to separately "store" the value $f(\alpha)$. This is no surprise as the folding operator can never "grasp" this information since $\mathcal{F}^b f(\alpha) = 0$ in case f is continuous. The reconstruction step is stable.

If we use the same extension operator in the even case, the reconstruction becomes unstable since it has $l^b \Leftrightarrow r^b$ in the denominator. We therefore introduce the following extension operator in the even case,

$$\mathcal{X}\,f \;=\; \mathcal{M}^b\,f + 2f'(\alpha)\,(x \Leftrightarrow \alpha) \quad \text{on} \quad \mathbf{R} \setminus I,$$

where we assume that $f \in \mathcal{C}^1$. Consequently, $\mathcal{X}f \in \mathcal{C}^1$. The folding operator is given by

$$\mathcal{F}^{*b} \;=\; (l^b + r^b)\,f + 2f'(\alpha)\,(x \Leftrightarrow \alpha)\,l^b,$$

from which we see that the inverse operator again is stable. Here one needs to separately "store" the value $f'(\alpha)$.

The construction on the right boundary of an interval is completely similar. Also, it is possible to construct extension operators that preserve more smoothness at a cost of having to store more information separately.

7.7 Equal parity folding

In this section, we take a closer look at the folding operator that takes a smooth function into a function with the same parity left and right of the folding point. We call such a folding operator an *equal parity folding* (EPF) operator. In Section 7.5, we proved that they are not invertible in $\mathrm{L}_2(\mathbf{R})$, see Lemma 7.11. Nevertheless they were used successfully for image compression, see [2]. In this section, we study their behavior more carefully and try to understand why they sometimes are useful.

We start by introducing two new operators as,

$$\mathcal{E} \;=\; 1 + \mathcal{M} \quad \text{and} \quad \mathcal{D} \;=\; 1 \Leftrightarrow \mathcal{M},$$

which map any function into an **e**ven (resp. o**d**d) function so that

$$\frac{\mathcal{E}+\mathcal{D}}{2} \;=\; 1.$$

Note that they are both self-adjoint and provide an orthogonal splitting of L_2 into $\mathcal{E}\,\mathrm{L}_2 \oplus \mathcal{D}\,\mathrm{L}_2$. We again take $\alpha = 0$ and omit the subscript α. We assume that the cutoff functions are continuous and that they satisfy

$$\lim_{x\to-\infty} l(x) \;=\; 1, \qquad\qquad \lim_{x\to+\infty} l(x) \;=\; 0,$$

and $r = \mathcal{M}\,l$. The EPF operator with (even – even) parity can be written as

$$\mathcal{F} \;=\; \chi^l\,\mathcal{E}\,l + \chi^r\,\mathcal{E}\,r. \tag{7.3}$$

We immediately see that the EPF operator is self-adjoint. Also, it commutes with $\mathcal{M}$ and maps even (resp. odd) functions into even (resp. odd) functions. We first study how the EPF operator behaves on these subspaces of even or odd functions.

Lemma 7.13 *On $\mathcal{E}\,\mathrm{L}_2$, $\mathcal{F}$ coincides with $l + r$. On $\mathcal{D}\,\mathrm{L}_2$, $\mathcal{F}$ coincides with $(\chi^l \Leftrightarrow \chi^r)\,(l \Leftrightarrow r)$.*

Proof : In the even case we have,

$$\mathcal{F}\,f \;=\; \chi^l\,\mathcal{E}\,l\,f + \chi^r\,\mathcal{E}\,r\,\mathcal{M}\,f \;=\; \chi^l\,\mathcal{E}\,l\,f + \chi^r\,\mathcal{E}\,\mathcal{M}\,l\,f \;=\; \mathcal{E}\,l\,f,$$

while in the odd case

$$\mathcal{F}\,f \;=\; \chi^l\,\mathcal{E}\,l\,f \Leftrightarrow \chi^r\,\mathcal{E}\,r\,\mathcal{M}\,f \;=\; \chi^l\,\mathcal{E}\,l\,f \Leftrightarrow \chi^r\,\mathcal{E}\,l\,f \;=\; (\chi^l \Leftrightarrow \chi^r)\,(l \Leftrightarrow r)\,f.$$

□

We introduce two new functions e and d as

$$e \;=\; \mathcal{E}\,l \quad \text{and} \quad d \;=\; \mathcal{D}\,l.$$

The lemma implies that we can write $\mathcal{F}$ as

$$\mathcal{F} \;=\; e\,\mathcal{E} + d\,(\chi^l \Leftrightarrow \chi^r)\,\mathcal{D}. \tag{7.4}$$

This helps us to formulate the following lemmas.

Lemma 7.14 *The EPF operator is bounded.*

Proof : Follows immediately from the representation (7.4), the fact that the cutoff functions are continuous and from their limit conditions. □

Lemma 7.15 $\ker \mathcal{F} \;=\; \{0\}$.

Proof : From (7.3) we see that $\mathcal{F} f = 0$ implies that both lf and rf are odd. Consequently ef and df are odd. The former implies that f is odd, the latter that f is even. Thus $f = 0$. □

We know that the kernel and the closure of the range of a self-adjoint operator form an orthogonal splitting of L_2. The former two lemmas thus imply that the range of $\mathcal{F}$ is dense in L_2. We show later that the range of $\mathcal{F}$ is actually smaller than L_2. From the representation (7.4) it also follows that the spectrum of $\mathcal{F}$ is equal to $[m, M]$ with

$$m = \min_x\{e, d\} < 0 \quad \text{and} \quad M = \max_x\{e, d\} > 0.$$

Lemma 7.16 *The EPF operator is not invertible on its range.*

Proof : Remember that an operator is invertible when its inverse exists and when the inverse is bounded. We prove that $\mathcal{F}$ is not invertible by constructing an odd function w with norm one so that $\mathcal{F} w$ can have arbitrarily small norm. Let

$$w = \frac{1}{2\sqrt{\delta}}\big(\chi_{[-\delta,0]} \Leftrightarrow \chi_{[0,\delta]}\big).$$

Then

$$\|\mathcal{F} w\| = \|d\, w\| \leqslant \max_{x\in[-\delta,\delta]} d.$$

Since the cutoff functions are continuous this can be made arbitrarily small. □

It is easy to see what the inverse operator, at least formally, looks like. It again is an EPF operator with the parity (even – even). We denote it with $\widetilde{\mathcal{F}}$ where

$$\widetilde{\mathcal{F}} = \tilde{e}\,\mathcal{E} + \tilde{d}\,(\chi^l \Leftrightarrow \chi^r)\,\mathcal{D},$$

with

$$\tilde{e} = 1/e \quad \text{and} \quad \tilde{d} = 1/d.$$

It immediately follows that formally $\widetilde{\mathcal{F}}\,\mathcal{F} = 1$. The inverse operator can also be written in a form similar to (7.3), where

$$\widetilde{\mathcal{F}} = \chi^l\,\mathcal{E}\,\tilde{l} + \chi^r\,\mathcal{E}\,\tilde{r},$$

with

$$\tilde{l} = \frac{l}{l^2 \Leftrightarrow r^2} \quad \text{and} \quad \tilde{r} = \frac{r}{r^2 \Leftrightarrow l^2}.$$

The fact that $\mathcal{F}$ is not invertible shows here in the fact that $\tilde{d}$ has a singularity at $x = 0$. This also tells us that $\widetilde{\mathcal{F}}$ can take a function out of L_2, since this singularity is not necessarily square integrable.

We understand that $\mathcal{F}$ is bounded and invertible on the subspace of even functions in case e is bounded away from 0 and ∞. For the odd functions we cannot make a similar statement as d always vanishes in the origin.

We can characterize the range of $\mathcal{F}$ as

$$\text{range}\,\mathcal{F} \;=\; \{f \in \mathrm{L}_2 \mid \mathcal{D}\,f/d \in \mathrm{L}_2\}.$$

If we also assume that the cutoff functions belong to $\mathcal{C}^1$, we see that $\tilde{d} = \mathcal{O}(1/x)$. This means that a function belongs to the range of $\mathcal{F}$ in case it behaves like $\mathcal{O}(x^{1/2-\delta})$ in a neighborhood of the origin. A typical function that does not belong to the range of $\mathcal{F}$ is a function with a discontinuity at the folding point. Similarly we can write the range of $\tilde{\mathcal{F}}$ as

$$\text{range}\,\tilde{\mathcal{F}} \;=\; \{f \in \mathrm{L}_2 \mid d\,\mathcal{D}\,f \in \mathrm{L}_2\},$$

and we note that $\mathrm{L}_2 \subset \text{range}\,\tilde{\mathcal{F}}$. Again, a function that $\tilde{\mathcal{F}}$ typically takes out of L_2 is a function with a discontinuity across the folding point.

Under the assumption that the cutoff functions belong to $\mathcal{C}^1$, a function of the range of $\tilde{\mathcal{F}}$ behaves like $\mathcal{O}(x^{-3/2-\delta})$ in a neighborhood of the origin.

We can also understand that $\mathcal{F}$ has some smoothing properties, i.e. it maps a function with a discontinuity at the origin into a continuous function as

$$\mathcal{F}\,\chi^l \;=\; l.$$

However, it doesn't smooth out discontinuities in the derivative as

$$\mathcal{F}\,x\,\chi^l \;=\; |x|\,l.$$

Let us discuss how these operators can be used in data compression. The idea again is to construct a total folding operator and then use a trigonometric basis with the right parity on each interval. In this case it is the cosine II basis. The compression is done in this basis, after which we use the inverse total folding operator to reconstruct the data. If we think of $\mathcal{F}$ as a kind of smoothing operator and of $\tilde{\mathcal{F}}$ as an operator that can blow up discontinuous functions, it makes sense to use $\tilde{\mathcal{F}}$ to construct the total folding operator and $\mathcal{F}$ for its inverse. This has the following advantages:

- The error introduced by the compression in the trigonometric basis cannot get blown up, as $\mathcal{F}$ is bounded.
- Discontinuities across the folding points get smoothed by $\mathcal{F}$. In other words, the blocking effect is reduced.

Note: the idea to switch the two operators around was also suggested in [2].

This approach works well as long as the function is smooth at the folding point. This can be understood as follows. If a function is smooth, we can write a local first order approximation as

$$f(x) \;\approx\; f(0) + f'(0)\,x + \mathcal{O}(x^2).$$

The first term is even and thus does not pose a problem as the folding operators are bounded and invertible on the space of even functions. The second term is odd, but (locally) belongs to the range of $\mathcal{F}$, and thus does not cause any trouble. Problems, however, occur when the function is discontinuous at a folding point. We illustrate this with an example in a later section.

The construction of the total folding operators is completely similar to the biorthogonal case and we adopt the same notation (i.e. the integer subscripts). Evidently, there is no need to alternate the parities. On each interval we use the cosine II basis or the functions $\chi_l\, C_{l,k}$ where

$$C_{l,k} \;=\; \sqrt{\frac{2}{|I|}}\cos k\frac{\pi}{|I|}(x \Leftrightarrow 2l \Leftrightarrow 1).$$

It should be clear from the discussion above that the $\mathcal{F}\,\chi_l\, C_{l,k}$ cannot generate a basis for L_2, but merely form a set whose linear span is dense. Out of curiosity we remark that still

$$\mathcal{F}\,\chi_l\, C_{k,l} \;=\; b_l\, C_{k,l},$$

where b_l is the usual bell function.

One gets a resolution of the unity in case $e = 1$. One degree of freedom is still left in the choice of d. We can use this to also obtain a *resolution of the linear*, i.e. a representation of each linear function by two basis functions on an interval after folding by letting

$$d \;=\; \frac{x}{1 \Leftrightarrow 2/\pi\cos(\pi x/2)} \qquad \text{for} \qquad x \in [\Leftrightarrow 1, 1],$$

and consequently

$$l \;=\; (1+d)/2 \qquad \text{and} \qquad r \;=\; (1 \Leftrightarrow d)/2.$$

Figure 7.7 shows the equal parity folding of the function x with folding points 0 and 1. We see that on the interval [0,1] it coincides with a function of the form $A + B\,\cos(\pi x)$ (dashed). The resolution of the constant implies that for each interval the constant and linear component of the function can be represented efficiently. If we eliminate these components from the function, what is left typically looks like a waveform, which in turn can be represented by the higher frequency basis functions.

7.8 Implementation and results

So far the discussion only handled functions of a continuous variable. In applications the construction needs to be discretized. A function f is then given as a sequence $\{f_n\}$ where the "samples" f_n can be seen as pointwise evaluations on a regular grid in case f is continuous, or as average values of f in a neighborhood of

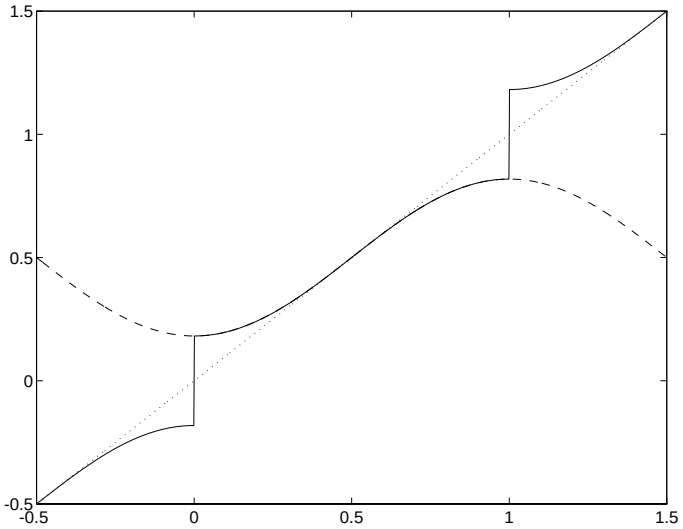

Figure 7.7: Equal parity folding of a linear function

the grid point if not. For each local trigonometric basis, a discrete implementation is available, which is based on the FFT. We know that the implementation of the FFT is most straightforward in case the number of samples is a power of two.

We first need to decide whether we want to use a staggered or non-staggered discretization. In a *non-staggered discretization*, the boundaries of the interval coincide with a grid point, while in a *staggered discretization* the boundaries of the interval fall between grid points.

In the orthogonal construction both discretizations are possible. The fact that a folded function is discontinuous at the folding point does not pose a problem in the non-staggered discretization. At the folding point we only need the value of the "even" part as we know that the "odd" part vanishes. In the biorthogonal case still both options can be used. In this case the non-staggered discretization has the disadvantage that the "even-even" intervals contain two more samples than the "odd-odd" intervals. This makes implementation harder.

In the EPF case one has to use the staggered implementation as some of the cutoff functions have a singularity at the folding point. This makes it possible to implement operators that in the L_2 sense are unbounded or not invertible. The fact that the range of $\mathcal{F}$ is dense in L_2 ensures that these discrete operators are invertible. However, this unavoidably results in ill-conditioned discrete operators. We always use the staggered discretization.

A transform coding scheme does not only involve a transform, but also a quantization and encoding step. However, we will not go into these issues. Instead, we include some results obtained by simply retaining the coefficients above a certain threshold and setting the others to zero ("clipping"). The compression ratio is then defined as the number of data samples divided by the number of transform coefficients that were retained.

The first example involves a function that has discontinuities at the folding points. We work on the interval $[0, 1]$ where we assume that the functions are extended periodically to the real line. Take the function

$$f \;=\; 2\chi_{[1/4,3/4)} \Leftrightarrow 1,$$

which generates a square wave. We take $1/4$ and $3/4$ as folding points, each with $\epsilon = 1/4$. We use the staggered discretization with grid size $h = 1/100$. After the appropriate local trigonometric transform on each interval, we retain the largest 15% of the coefficients and set the others to zero. We then perform the inverse trigonometric transform and unfolding. Figures 7.8 and 7.9 show the folding of f in the biorthogonal and EPF case and these functions after the clipping of the trigonometric coefficients (dashed). Figures 7.10 and 7.11 then show the unfolding of these functions versus the original function (dashed). As we predicted, the EPF performs badly in this case.

In a second example we consider a smooth function,

$$f(x) \;=\; e^{-(4x-2)^4},$$

and use the same folding points. Figure 7.12 gives the norm of the difference between the compressed and the original function as a function of the percentage of the coefficients that were retained. As we expected, the EPF behaves better here. If the percentage of coefficients kept is less than 15, its error is about 10 times smaller than in the biorthogonal case.

Next we consider an example involving image compression. As we mentioned, a standard here is JPEG, which uses a local cosine basis on 8×8 blocks. We compare it with the biorthogonal and EPF basis on the same size of blocks. Figure 7.13 and 7.14 give the mean square error (MSE) in function of the compression ratio, for two 512×512 gray value images (Lena and Peppers). We see that the two new folding techniques are better than JPEG. For small compression ratios biorthogonal is better, while for higher compression EPF is better. In Figures 7.15 and 7.16 (detail) we show the original and compressed (18:1) images for Lena. We see that the biorthogonal basis is the best at reducing the blocking effect.

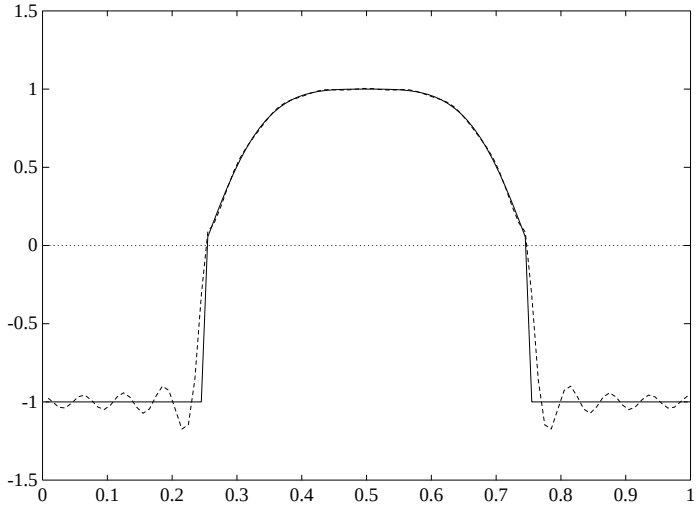

Figure 7.8: Biorthogonal folding of the block wave before and after clipping.

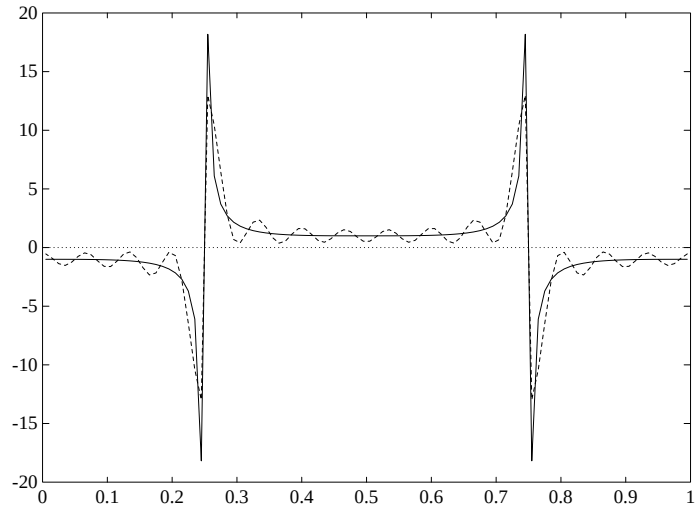

Figure 7.9: Equal parity folding of the block wave before and after clipping.

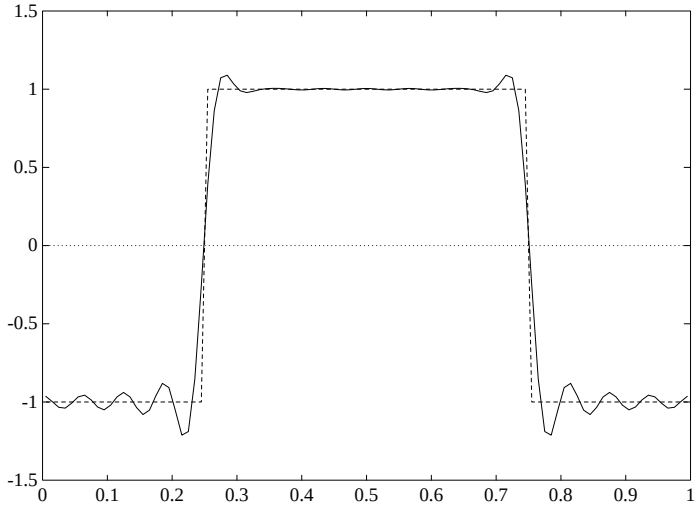

Figure 7.10: Reconstructed block wave (biorthogonal case).

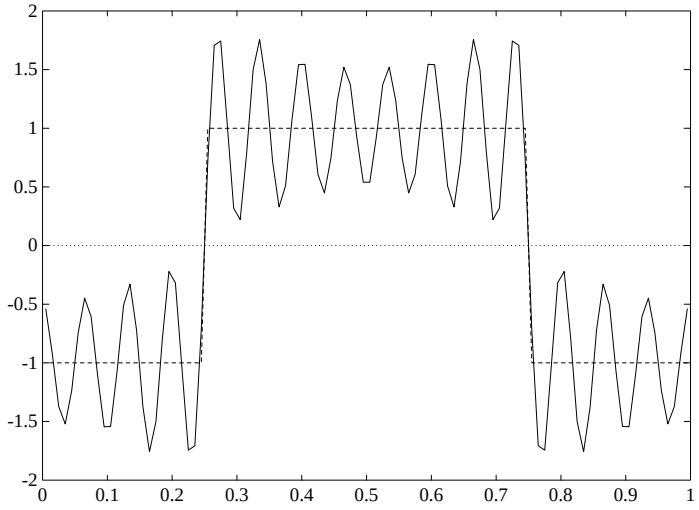

Figure 7.11: Reconstructed block wave (EPF case).

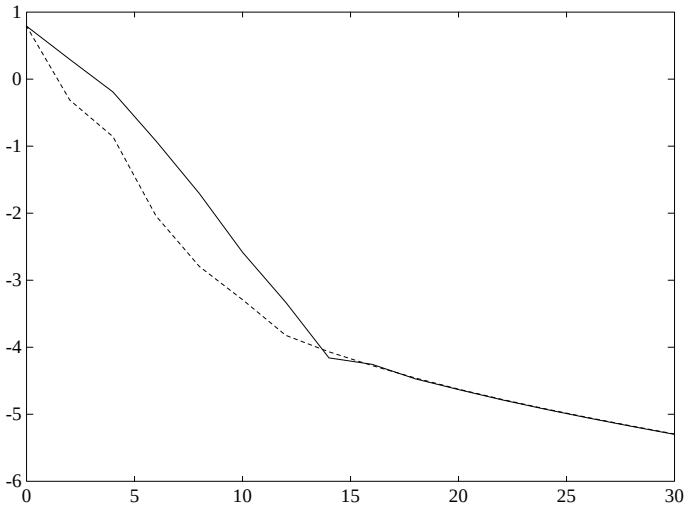

Figure 7.12: Error (logarithmic) as a function of percentage of coefficients.

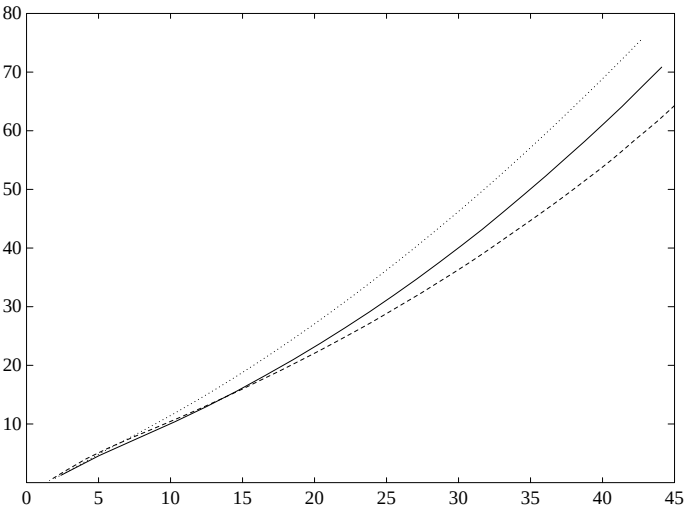

Figure 7.13: Mean square error for Lena image.
(JPEG = dotted, biorthogonal = full, EPF = dashed)

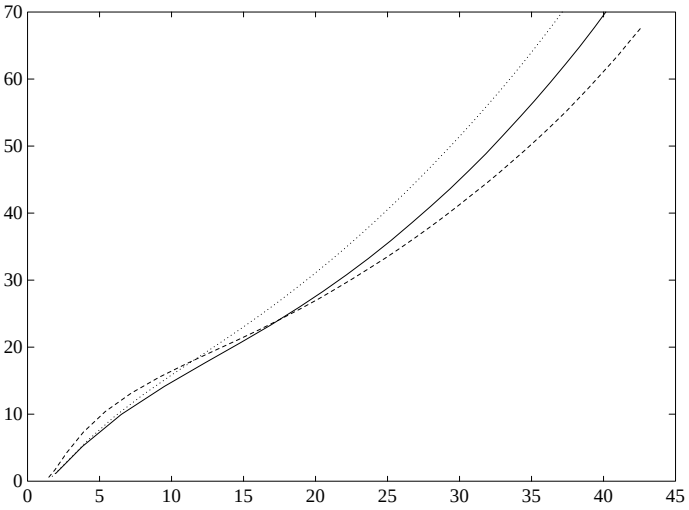

Figure 7.14: Mean square error for Peppers image.
(JPEG = dotted, biorthogonal = full, EPF = dashed)

Figure 7.15: Original and compressed images for Lena.

(top left: original, top right: JPEG,
bottom left: EPF, bottom right: biorthogonal)

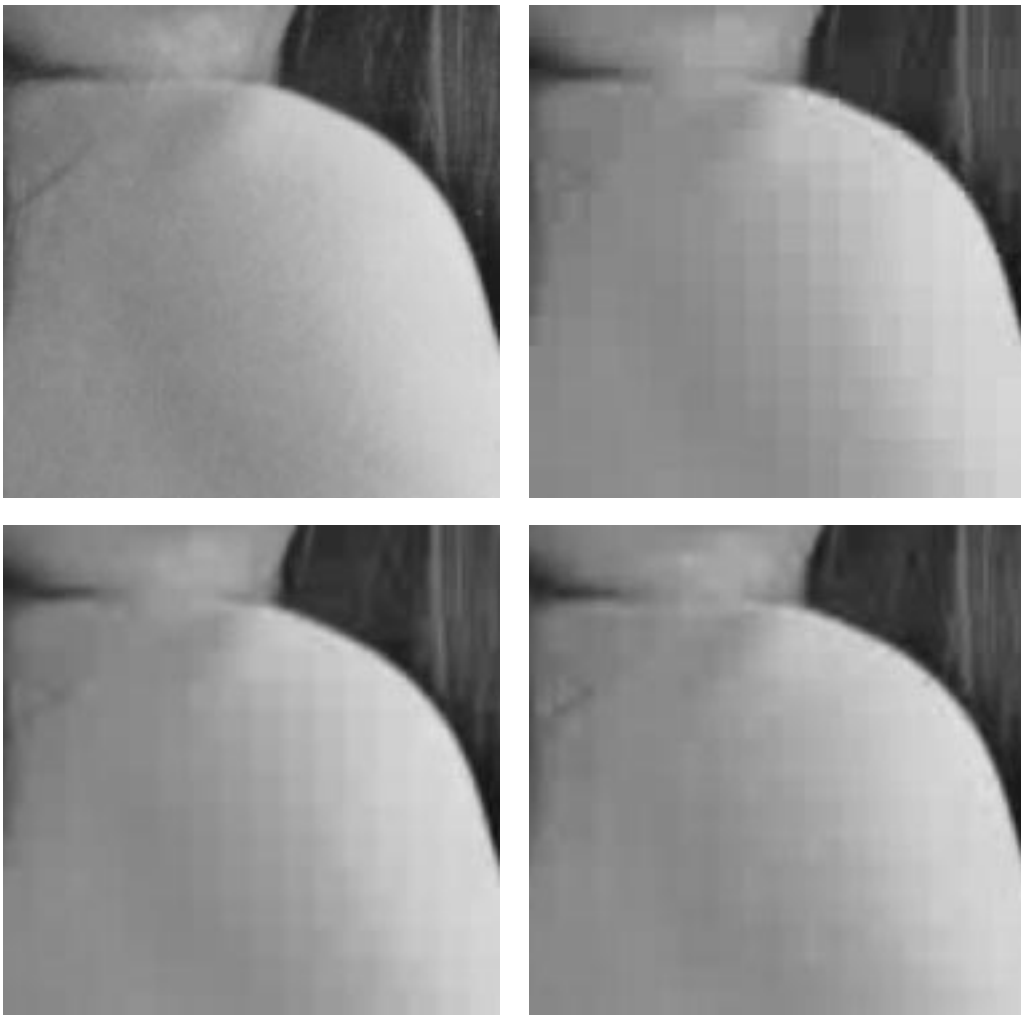

Figure 7.16: Original and compressed images for Lena (detail of the shoulder).

(top left: original, top right: JPEG,
bottom left: EPF, bottom right: biorthogonal)

7.9 Conclusion and future research

As we pointed out, the next stage in a transform coding scheme involves the *quantization* and *encoding* of the coefficients. The quantization step consists of only allowing a finite number of possible values for the coefficients. Every coefficient value that falls inside a certain interval is replaced by the same number according to a quantization table. Coefficients below a certain threshold are mapped to zero. The table needs to be designed in such a way that the coefficients can be represented with a minimal amount of bits without introducing too big an error in the compressed data. The encoding step consists of representing the values of the coefficients together with their position (essentially their index) by a bit stream.

One direction for future research is the design of a quantizer and encoder adapted to the smooth local trigonometric bases. This requires a careful study of the statistical behavior of the coefficients for a large set of possible data. Techniques that can be used here are vector quantization, run-length encoding, and arithmetic encoding.

Another direction is the application of smooth local trigonometric bases to the numerical solution of integral equations. We already pointed out in Chapter 2 how the discretization of a Calderón-Zygmund operator in the wavelet basis yielded a sparse matrix. However, operators arising in acoustic and electric scattering problems typically have oscillating kernels, and do not necessarily lead to sparse matrices. A simple example is an operator resulting from the Helmholtz equation, which looks like

$$T_k f(x) \;=\; \int_{-\infty}^{+\infty} \frac{e^{ik|z(x)-z(y)|}}{|z(x) \Leftrightarrow z(y)|} f(y)\, dy,$$

where $z(x) \in \mathbf{C}$ is the parametrization of a curve. We can discretize these operators in the smooth local trigonometric basis. First results [28] indicate that this yields a sparse matrix and thus allows fast computations.

A third direction involves the use of the folding operators for the design of other, not necessarily trigonometric based, smooth local bases. We know that the folding operator maps a smooth function into a function with specific parity properties. Any basis that matches these parities leads, after unfolding, to a smooth local basis. One idea is to use polynomials to generate the basis. This leads to a kind of spline basis. (It is not exactly a spline as the cutoff functions need not be polynomials). Building an orthogonal basis should be straightforward. This idea, combined with the construction of the Alpert wavelets (see Chapter 8) could result in a smooth wavelet basis based on polynomials.

Chapter 8

Conclusions and Future Research

"There is always one moment in childhood
when the door opens and lets the future in."
—Graham Greene, *The Power and the Glory (1940).*

In this chapter we draw some conclusions. We define a more general concept, *time-frequency basis*, and relate it to wavelets and smooth local trigonometric functions. We point out the shortcomings of the original constructions. We discuss how the work in this thesis not only solves some of these problems, but also raises directions for future research.

8.1 Time-frequency bases

8.1.1 General idea

Take a function f, defined over a time domain, and consider the following representations:

$$f(t) = \int_{-\infty}^{+\infty} f(s)\,\delta(t \Leftrightarrow s)\,ds,$$

and

$$f(t) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} \widehat{f}(\omega)\,e^{i\omega t}\,d\omega.$$

In both cases the function is represented as the integral of a building block (Dirac impulse or complex exponential) multiplied with a function. These representations can be seen as two extremes. In the first case the building blocks are extremely local in time, but loose all locality in frequency (since the Fourier transform of a Dirac impulse is a constant). In the second, the situation is precisely the opposite.

The whole idea of *time-frequency bases* is to find a compromise between these extremes. Let us be a little more precise. Consider a set of functions $\{\psi_\lambda\}$, where the general index λ belongs to a countable set. We look for a representation of f

as

$$f \;=\; \sum_{\lambda} c_\lambda \, \psi_\lambda. \tag{8.1}$$

We call a basis a time-frequency basis when it satisfies the following properties:

1. The basis functions form a "stable" basis for a general function space S. By "stable" we mean that for each $f \in S$, unique coefficients c_λ exist, so that the formal expression (8.1) converges in some sense. Typically, we consider convergence in the norm of S. Unconditional convergence, where the order of summation is irrelevant, is preferred. This is related to the fact that one can estimate the norm of f in S using the modulus of its coefficients $\{c_\lambda\}$, cf. the definition of a Riesz basis.

   We also want to have expressions for the coordinate functionals. These are continuous linear operators ψ_λ^* on S so that

   $$c_\lambda \;=\; \psi_\lambda^*(f).$$

2. The basis functions are local in time and frequency. This means that both ψ_λ and $\widehat{\psi}_\lambda$ are concentrated around their center, and die off rapidly away from it. The decay can be inverse polynomial or exponential, or the functions can even be compactly supported. Of course, this can only be achieved within the limitations of the Heisenberg uncertainty, cf. the discussion in Section 2.2. Usually, we also want to impose localization on the coordinate functionals.

   The localization implies that if f has some local behavior, either in time or in frequency, only a limited amount of coefficients will be large, while the others are small. Consequently, we can represent f within a certain accuracy using only a few basis functions. As we already saw on several occasions, this is the key to most applications.

   Note that localization of $\widehat{\psi}_\lambda$ corresponds to smoothness of ψ_λ and vice versa.

3. The basis is easy to implement on a computer. This means that we need explicit expressions for the coordinate functionals, so that given f, we can compute the c_λ. Secondly, the functions ψ_λ need to be easy to evaluate so that given the coefficients c_λ, we can reconstruct f.

   Moreover, we need a discrete equivalent of (8.1) and an associated fast transform algorithm.

8.1.2 Examples

In this thesis, we encountered two main examples of time-frequency bases.

Algebraic wavelets: In the case of algebraic wavelets, the basis functions are the dyadic translates and dilates of one function. We saw that they can form a Riesz basis for $L_2(\mathbf{R})$. The coordinate functionals are inner products with dual wavelets, which also are the translates and dilates of one function. We have localization in time domain through the compact support. Localization in frequency domain follows from the smoothness and the vanishing moments.

Evidently, when f is local in time, only few wavelet coefficients are large. When f is smooth, and thus localized in frequency, the wavelet coefficients decay rapidly.

The compact support of the basis and dual function, and the nestedness of the multiresolution spaces, immediately result in a fast wavelet transform.

Smooth local trigonometric functions: In the case of smooth local trigonometric functions, localization in time is obtained by splitting the real line into intervals with the use of smooth bell functions. Localization in frequency follows from the use of a trigonometric basis on each interval. All basis functions and dual functions are compactly supported. Again they give rise to Riesz bases. The transform algorithm builds upon the folding operators and the FFT.

As we already pointed out, smooth local trigonometric functions and wavelets are related through the Fourier transform. We call these functions *first generation wavelets*. Their merit is to provide a computational tool for ideas that had been used in abstract mathematics for a long time.

8.1.3 Shortcomings of first generation wavelets

Although they obviously are very useful, first generation wavelets have some shortcomings. In this section, we point out the main ones.

1. First generation wavelets form bases for functions defined on the real line. Using tensor products, a generalization to the whole Euclidean space $\mathbf{R}^n$ is possible. However, in many cases, we work with functions defined on fairly arbitrary, possibly non-smooth, subsets of $\mathbf{R}^n$. This is the case in applications such as data segmentation and the solution of partial differential equations.

   Also, we sometimes need time-frequency bases for functions that live on curves and surfaces. This is, for example, of use in the solution of differential equations on non-smooth domains. Using layer potential techniques, the differential equations are reformulated as integral equations on the boundary of the domain.

2. First generation wavelets provide a Riesz basis for L_2. This means that the basis and dual functions are biorthogonal with respect to the L_2 inner product.

However, often one needs basis adapted to different inner products, such as inner products that use a different measure, operator inner products, or inner products associated with a curve or surface.

3. A typical problem with the first generation wavelets is that they are not invariant under geometric operations ($\mathcal{G}$), such as translation, dilation and rotation. In other words, there is often no "simple" relationship between the coefficients of f and those of $\mathcal{G}f$. A "simple" relationship leads to an algorithm that is computationally far less expensive than the calculation of the coefficients.

   Such invariance is useful in the analysis and compression of digital video. One frame is often the image of the previous frame under such a geometric operation.

8.1.4 Solutions

We define *second generation wavelets* to be any functions forming a time-frequency basis that overcomes some of the above problems. Using this definition, already several instances of second generation wavelets exist, the most common example being wavelets on an interval.

The motivation for this thesis was the use of time-frequency bases in numerical analysis. It somehow reflects the transition between first and second generation wavelets. In the first part we studied how one can use first generation wavelets in numerical analysis. More specifically, we developed two basic tools, quadrature formula and error expansions. In the second part we showed how one can construct second generation wavelets for use in numerical analysis. More precisely, we constructed weighted wavelets and operator wavelets which lead to a fast algorithm for ordinary differential equations. We also generalized smooth local trigonometric bases to the biorthogonal case and showed how they can be used in data compression.

8.2 Future Research

The results of this thesis are only a small step in the whole evolution of wavelets and time-frequency bases. They provide answers to some of the open problems, and perhaps more important, open new directions and raise new questions. Already at the end of each chapter, we pointed out ideas for future research related to the material of the chapter. In this section, we discuss ideas for future research related to the thesis as a whole.

An initial direction involves the further exploration of the applicability of the average-interpolation scheme for the construction of second generation wavelets.

David Donoho already showed that it leads to an elegant construction of wavelets on an interval. This can easily be generalized to wavelets satisfying specific boundary conditions. First experiments show that average interpolation can also be used for the construction of wavelets on arbitrary domains and wavelets on curves and surfaces. The latter are closely related to weighted wavelets. This can be seen as follows. Consider a curve , in the complex plane $\mathcal{C}$. An inner product of two functions defined on this curve is given by

$$\langle f, g \rangle_\Gamma = \int_{z \in \Gamma} f(z)\, \overline{g(z)}\, dz.$$

Consider a parametrization of , , given by $z(t)$ for $t \in \mathbf{R}$, and let $f^*(t) = f(z(t))$. Then

$$\langle f, g \rangle_\Gamma = \int_{-\infty}^{+\infty} f^*(t)\, \overline{g^*(t)}\, z'(t)\, dt.$$

This is a weighted inner product with $w = z'$. Note that the weight is complex valued. With a simple condition of the curve (such as a "cord-arc" condition), it is possible to ensure that the the local moments $M_{j,l}^0$ vanish. Another common inner product on the curve uses $|dz|$ as differential, which leads to the weight $w = |z'|$. This weight is always positive.

A second direction involves the use of Alpert wavelets. In his thesis, Bradley Alpert came up with an alternative construction of orthogonal wavelets, which he called "multiwavelets" [7, 9]. His construction only relies on properties of polynomials and on Gram-Schmidt orthogonalization. Their only disadvantage is that they are non-smooth. Nevertheless they were used successfully in the solution of integral equations [8] and in data compression [184].

Since the construction does not rely on the Fourier transform, it can be used for second generation wavelets. First, Alpert wavelets can be generalized to the biorthogonal case. Secondly, since the support of the wavelets on one level does not intersect, one can build Alpert wavelets on very general geometries. For example, using triangulations, one can build wavelets adapted to life on domains or on a surface. For more details, we refer to [109, 114].

Several solutions dealing with the invariance problem already exist. One technique is to use overcomplete representations, or so-called *frames*. Beylkin presents an algorithm for the calculation of all circular shift of a vector in [23]. Other invariant bases were presented in [162]. Mallat uses a different technique. He uses the zero crossings or maxima of the wavelet bands to represent general functions [133, 137]. This, however, has the disadvantage that no explicit reconstruction algorithm exists, and that the representation is not unique. A new idea for finding shift invariant representation relies on a theorem of Bar-David. The theorem states that every bandlimited function can be uniquely represented by its crossing with a fixed cosine function. A simple reconstruction formula is available. For more details we refer to [179, 198].

Finally, once these second generation wavelets are available, we can return to the techniques of Chapters 3 and 4, and design quadrature formulae and error expansions for them.

"Who controls the past, ran the Party slogan, controls the future: who controls the present controls the past."
—George Orwell, *1984 (1949).*

Bibliography

[1] M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions.* Dover Publications, New York, 1965.

[2] G. Aharoni, A. Averbuch, R. Coifman, and M. Israeli. Local cosine transform — A method for the reduction of the blocking effect in JPEG. *J. Math. Imag. Vision*, 3:7–38, 1993.

[3] G. Aharoni, A. Averbuch, R. Coifman, and M. Israeli. Local cosine transform — A method for the reduction of the blocking effect in JPEG. In A. F. Laine, editor, *Mathematical Imaging: Wavelet Applications in Signal and Image Processing, San Diego, CA*, pages 205–217. SPIE Proceedings Series volume 2034, 1993.

[4] A. Aldroubi and M. Unser. Families of wavelet transforms in connection with Shannon's sampling theory and the Gabor transform. In [35], pages 509–528.

[5] A. Aldroubi and M. Unser. Sampling procedures in function spaces and asymptotic equivalence with Shannon's sampling theory. *Numer. Funct. Anal. Optim.*, to appear.

[6] A. Aldroubi and M. Unser. Families of multiresolution and wavelet spaces with optimal properties. *Numer. Funct. Anal. Optim.*, 14:417–446, 1993.

[7] B. Alpert. A class of bases in L_2 for the sparse representation of integral operators. *SIAM J. Math. Anal.*, 24(1):246–262, 1993.

[8] B. Alpert, G. Beylkin, R. Coifman, and V. Rokhlin. Wavelet-like bases for the fast solution of second-kind integral equations. *SIAM J. Sci. Comput.*, 14(1):159–184, 1993.

[9] B. K. Alpert. Wavelets and other bases for fast numerical linear algebra. In [35], pages 181–216.

[10] K. Amaratunga and J. R. Williams. Wavelet based Green's function approach to 2D PDEs. *Engrg. Comput.*, 10(4), 1993.

[11] K. Amaratunga, J. R. Williams, S. Qian, and J. Weiss. Wavelet Galerkin solutions for one dimensional partial differential equations. *Internat. J. Numer. Methods Engrg.*, to appear.

[12] L. Andersson, N. Hall, B. Jawerth, and G. Peters. Wavelets on closed subsets of the real line. In [160], pages 1–61.

[13] L. Andersson, B. Jawerth, and M. Mitrea. The Cauchy singular integral operator and Clifford wavelets. In [20], pages 525–546.

[14] P. Auscher. Remarks on the local Fourier bases. In [20], pages 203–218.

[15] P. Auscher and Ph. Tchamitchian. Bases d'ondelettes sur les courbes corde-arc, noyau de Cauchy et espaces de Hardy associés. *Rev. Mat. Iberoamericana*, 5:139–170, 1989.

[16] P. Auscher and Ph. Tchamitchian. Conjecture de Kato sur les ouverts de $\mathbf{R}$. *Rev. Mat. Iberoamericana*, 8(2):149–199, 1992.

[17] P. Auscher, G. Weiss, and V. Wickerhauser. Local sine and cosine bases of Coifman and Meyer and the construction of smooth wavelets. In [35], pages 237–256.

[18] E. Bacry, S. Mallat, and G. Papanicolaou. A wavelet based space-time adaptive numerical method for partial differential equations. *RAIRO Math. Modelling Numer. Anal.*, 26(7):793–834, 1992.

[19] G. Battle. A block spin construction of ondelettes. *Comm. Math. Phys.*, 110:601–615, 1987.

[20] J. Benedetto and M. Frazier, editors. *Wavelets: Mathematics and Applications*. CRC Press, Boca Raton, 1993.

[21] M. A. Berger. Random affine iterated function systems: Curve generation and wavelets. *SIAM Rev.*, 31(4):614–627, 1989.

[22] G. Beylkin. On wavelet-based algorithms for solving differential equations. In [20], pages 449–466.

[23] G. Beylkin. On the representation of operators in bases of compactly supported wavelets. *SIAM J. Numer. Anal.*, 29(6):1716–1740, 1992.

[24] G. Beylkin, R. Coifman, and V. Rokhlin. Wavelets in numerical analysis. In [156], pages 181–210.

[25] G. Beylkin, R. Coifman, and V. Rokhlin. Fast wavelet transforms and numerical algorithms i. *Comm. Pure Appl. Math.*, 44:141–183, 1991.

[26] M. E. Bock and G. Pliego. Estimating functions with wavelets part ii: Using a Daubechies wavelet in nonparametric regression. *Statistical Computing and Statistical Graphics Newsletter*, 3(2):27–34, November 1992.

[27] C. De Boor. On the cardinal spline interpolant to e^{iut}. *SIAM J. Matrix Anal. Appl.*, 7(6):930–941, 1976.

[28] B. Bradie, R. Coifman, and A. Grossmann. Fast numerical computations of oscillatory integrals related to acoustic scattering, i. *Appl. Comput. Harmon. Anal.*, 1(1):94–99, 1993.

[29] J. H. Bramble, J. E. Pasciak, and J. Xu. Parallel multilevel preconditioners. *Math. Comp.*, 55:1–22, 1990.

[30] L. Breiman. *Probability*. Classics in Applied Mathematics. SIAM, Philadelphia, 1992. Originally published by Addison-Wesley, 1968.

[31] R. L. Burden and J. D. Faires. *Numerical Analysis*. PWS-Kent Publishing Company, Boston, 4th edition, 1989.

[32] A. S. Cavaretta, W. Dahmen, and C. A. Micchelli. Stationary subdivision. *Memoirs Amer. Math. Soc.*, 93(453), 1991.

[33] C. Chui and E. Quak. Wavelets on a bounded interval. In D. Braess and L. L. Schumaker, editors, *Numerical Methods of Approximation Theory*, pages 1–24. Birkhäuser-Verlag, Basel, 1992.

[34] C. K. Chui. *An Introduction to Wavelets*. Academic Press, San Diego, CA, 1992.

[35] C. K. Chui, editor. *Wavelets: A Tutorial in Theory and Applications*. Academic Press, San Diego, CA, 1992.

[36] C. K. Chui and C. Li. Non-orthogonal wavelet packets. *SIAM J. Math. Anal.*, 24(3):712–738, 1993.

[37] C. K. Chui and J. Z. Wang. A cardinal spline approach to wavelets. *Proc. Amer. Math. Soc.*, 113:785–793, 1991.

[38] C. K. Chui and J. Z. Wang. A general framework of compactly supported splines and wavelets. *J. Approx. Theory*, 71(3):263–304, 1992.

[39] C. K. Chui and J. Z. Wang. On compactly supported spline wavelets and a duality principle. *Trans. Amer. Math. Soc.*, 330:903–915, 1992.

[40] Z. Ciesielski. Constructive function theory and spline systems. *Studia Math.*, 52:277–302, 1973.

[41] A. Cohen. Biorthogonal wavelets. In [35], pages 123–152.

[42] A. Cohen. Ondelettes, analyses multiresolutions et filtres miroirs en quadrature. *Ann. Inst. H. Poincaré, Anal. Non Linéaire*, 7(5):439–459, 1990.

[43] A. Cohen and I. Daubechies. Non-separable bidimensional wavelet bases. *Rev. Mat. Iberoamericana*, 9(1):51–137, 1993.

[44] A. Cohen and I. Daubechies. On the instability of arbitrary biorthogonal wavelet packets. *SIAM J. Math. Anal.*, 24(5):1340–1354, 1993.

[45] A. Cohen, I. Daubechies, and J. Feauveau. Bi-orthogonal bases of compactly supported wavelets. *Comm. Pure Appl. Math.*, 45:485–560, 1992.

[46] A. Cohen, I. Daubechies, B. Jawerth, and P. Vial. Multiresolution analysis, wavelets and fast algorithms on an interval. *C. R. Acad. Sci. Paris Sér. I Math.*, I(316):417–421, 1993.

[47] A. Cohen, I. Daubechies, and P. Vial. Multiresolution analysis, wavelets and fast algorithms on an interval. *Appl. Comput. Harmon. Anal.*, 1(1):54–81, 1993.

[48] A. Cohen and J.-M. Schlenker. Compactly supported bidimensional wavelet bases with hexagonal symmetry. *Constr. Approx.*, 9(2):209–236, 1993.

[49] R. Coifman and Y. Meyer. Remarques sur l'analyse de Fourier à fenêtre. *C. R. Acad. Sci. Paris Sér. I Math.*, I(312):259–261, 1991.

[50] R. R. Coifman. A real variable characterization of H^p. *Studia Math*, 51, 1974.

[51] R. R. Coifman, P. W. Jones, and S. Semmes. Two elementary proofs of the L_2 boundedness of Cauchy integrals on Lipschitz curves. *J. Amer. Math. Soc.*, 2(3):553–564, 1989.

[52] R. R. Coifman, Y. Meyer, S. Quake, and M. V. Wickerhauser. Signal processing and compression with wave packets. In Y. Meyer, editor, *Proceedings of the International Conference on Wavelets, Marseille, 1989*. Masson, Paris, 1992.

[53] R. R. Coifman, Y. Meyer, and V. Wickerhauser. Size properties of wavelet packets. In [156], pages 453–470.

[54] R. R. Coifman and M. L. Wickerhauser. Entropy based algorithms for best basis selection. *IEEE Trans. Inform. Theory*, 38(2):713–718, 1992.

[55] D. Colella and C. Heil. The characterization of continuous four-coefficient scaling functions and wavelets. *IEEE Trans. Inform. Theory*, 38(2):876–881, 1992.

[56] D. Colella and C. Heil. Characterizations of scaling functions: Continuous solutions. *SIAM J. Math. Anal.*, 15:496–518, 1994.

[57] S. Dahlke and I. Weinreich. Wavelet bases adapted to pseudo-differential operators. *Appl. Comput. Harmon. Anal.*, to appear.

[58] S. Dahlke and I. Weinreich. Wavelet-Galerkin-methods: An adapted biorthogonal wavelet basis. *Constr. Approx.*, 9(2):237–262, 1993.

[59] G. Dahlquist and A. Bjorck. *Numerical Methods*. Prentice-Hall, Englewood Cliffs, New Jersey, 1974.

[60] W. Dahmen and A. Kunoth. Multilevel preconditioning. *Numer. Math.*, 63(2):315–344, 1992.

[61] W. Dahmen and C. A. Micchelli. Using the refinement equation for evaluating integrals of wavelets. *SIAM J. Numer. Anal.*, 30(2):507–537, 1993.

[62] I. Daubechies. Two recent results on wavelets. In [160], pages 237–258.

[63] I. Daubechies. Orthonormal bases of compactly supported wavelets. *Comm. Pure Appl. Math.*, 41:909–996, 1988.

[64] I. Daubechies. The wavelet transform, time-frequency localization and signal analysis. *IEEE Trans. Inform. Theory*, 36(5):961–1005, 1990.

[65] I. Daubechies. *Ten Lectures on Wavelets*. CBMS-NSF Regional Conf. Series in Appl. Math., Vol. 61. Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.

[66] I. Daubechies. Orthonormal bases of compactly supported wavelets II: Variations on a theme. *SIAM J. Math. Anal.*, 24(2):499–519, 1993.

[67] I. Daubechies, A. Grossmann, and Y. Meyer. Painless nonorthogonal expansions. *J. Math. Phys.*, 27(5):1271–1283, 1986.

[68] I. Daubechies, S. Jaffard, and J.-L. Journé. A simple Wilson orthonormal basis with exponential decay. *SIAM J. Math. Anal.*, 22:554–572, 1991.

[69] I. Daubechies and J. C. Lagarias. Two-scale difference equations i. Existence and global regularity of solutions. *SIAM J. Math. Anal.*, 22(5):1388–1410, 1991.

[70] I. Daubechies and J. C. Lagarias. Two-scale difference equations ii. Local regularity, infinite products of matrices and fractals. *SIAM J. Math. Anal.*, 23(4):1031–1079, 1992.

[71] P.J. Davis and P. Rabinowitz. *Methods of Numerical Integration*. Academic Press, London, 1984.

[72] C. de Boor, R. A. DeVore, and A. Ron. On the construction of multivariate pre-wavelets. *Constr. Approx.*, 9(2):123–166, 1993.

[73] B. Deng and B. Jawerth. Biorthogonal wavelet packets on closed intervals. Preprint.

[74] B. Deng, B. Jawerth, G. Peters, and W. Sweldens. Wavelet probing for compression based segmentation. In A. F. Laine, editor, *Wavelet Applications in Signal and Image Processing*, pages 266–276. Proc. SPIE 2034, 1993.

[75] G. Deslauriers and S. Dubuc. Interpolation dyadique. In *Fractals, dimensions non entières et applications*, pages 44–55. Masson, Paris, 1987.

[76] G. Deslauriers and S. Dubuc. Symmetric iterative interpolation processes. *Constr. Approx.*, 5(1):49–68, 1989.

[77] R. A. DeVore, B. Jawerth, and B. J. Lucier. Image compression through wavelet transform coding. *IEEE Trans. Inform. Theory*, 38(2):719–746, 1992.

[78] R. A. DeVore, B. Jawerth, and B. J. Lucier. Surface compression. *Comput. Aided Geom. Des.*, 9(3):219–239, 1992.

[79] D. L. Donoho. Smooth wavelet decompositions with blocky coefficient kernels. In [160], pages 259–308.

[80] D. L. Donoho. Interpolating wavelet transforms. Preprint, Department of Statistics, Stanford University, 1992.

[81] D. L. Donoho. Unconditional bases are optimal bases for data compression and for statistical estimation. *Appl. Comput. Harmon. Anal.*, 1(1):100–115, 1993.

[82] T. Eirola. Sobolev characterization of solutions of dilation equations. *SIAM J. Math. Anal.*, 23(4):1015–1030, 1992.

[83] A. Erdélyri, W. Magnus, F. Oberhettinger, and F.G. Tricomi. *Higher Trancendental Functions*, volume I. McGraw-Hill Book Company, 1953.

[84] X. Fang and E. Séré. Adaptive multiple folding local trigonometric transforms and wavelet packets. *Appl. Comput. Harmon. Anal.*, 1(2), 1994.

[85] G. Fix and G. Strang. Fourier analysis of the finite element method in Ritz–Galerkin theory. *Stud. Appl. Math*, 48:265–273, 1969.

[86] P. Franklin. A set of continuous orthogonal functions. *Math. Ann*, 100:522–529, 1928.

[87] M. Frazier and B. Jawerth. Decomposition of Besov spaces. *Indiana Univ. Math. J.*, 34(4):777–799, 1985.

[88] M. Frazier and B. Jawerth. The φ-transform and applications to distribution spaces. In M. Cwikel et al., editor, *Function Spaces and Applications*, number 1302 in Lecture Notes in Math., pages 223–246, 1988.

[89] M. Frazier and B. Jawerth. A discrete transform and decompositions of distribution spaces. *J. Funct. Anal.*, 93:34–170, 1990.

[90] W. Gautschi. On the construction of Gaussian quadrature rules from modified moments. *Math. Comp.*, 24:245–260, 1970.

[91] W. Gautschi. Numerical condition related to polynomials. In C. de Boor and G. H. Golub, editors, *Recent Advances in Numerical Analysis*, pages 45–72. Mathematics Research Center, The University of Wisconsin, Academic Press, 1978.

[92] R. Glowinski, W. M. Lawton, M. Ravechol, and E. Tenenbaum. Wavelet solution of linear and nonlinear elliptic parabolic and hyperbolic problems in one space dimension. In *Proceedings of the 9th International Conference on Numerical Methods in Applied Sciences and Engineering*, Philadelphia, PA, 1990. SIAM.

[93] R. A. Gopinath and C. S. Burrus. On the moments of the scaling function. In *Proceedings of the ISCAS-92*. San Diego, CA, 1992.

[94] G. Gripenberg. Approximations of wavelet projections. Preprint, Department of Mathematics, University of Helsinki.

[95] A. Grossmann, R. Kronland-Martinet, and J. Morlet. Reading and understanding continuous wavelet transforms. pages 2–20.

[96] A. Grossmann and J. Morlet. Decompostion of Hardy functions into square integrable wavelets of constant shape. *SIAM J. Math. Anal.*, 15(4):723–736, 1984.

[97] A. Grossmann and J. Morlet. Decomposition of functions into wavelets of constant shape, and related transforms. In L. Streit, editor, *Mathematics and Physics, Lectures on Recent Results*. World Scientific Publishing, Singapore, 1985.

[98] A. Grossmann, J. Morlet, and T. Paul. Transforms associated to square integrable group representations i. General results. *J. Math. Phys.*, 26(10):2473–2479, 1985.

[99] A. Haar. Zur Theorie der orthogonalen Funktionen-Systeme. *Math. Ann.*, 69:331–371, 1910.

[100] W. Hackbusch. *Multi-Grid Methods and Applications.* Springer-Verlag, Berlin, 1985.

[101] C. E. Heil and D. F. Walnut. Continuous and discrete wavelet transforms. *SIAM Rev.*, 31(4):628–666, 1989.

[102] S. Jaffard. Wavelet methods for fast resolution of elliptic problems. *SIAM J. Numer. Anal.*, 29(4):965–986, 1992.

[103] A. J. E. M. Janssen. The Zak transform and sampling theorems for wavelet subspaces. Preprint, Philips Research Laboratories, Eindhoven, The Netherlands.

[104] A. J. E. M. Janssen. The Zak transform: a signal transform for sampled time-continuous signals. *Philips J. Res.*, 43:23–69, 1988.

[105] B. Jawerth. On Besov spaces. Technical Report 1, Lund, 1977.

[106] B. Jawerth, Y. Liu, and W. Sweldens. New folding operators for image compression. In H. H. Szu, editor, *Wavelet Applications*, Proc. SPIE 2224, pages 188–199, 1994.

[107] B. Jawerth, Y. Liu, and W. Sweldens. Signal compression with smooth local trigonometric bases. *Opt. Eng.*, 33(7):2125–2135, 1994.

[108] B. Jawerth and G. Peters. Wavelets on non smooth sets of $\mathbf{R}^n$. Preprint, 1993.

[109] B. Jawerth and W. Sweldens. Second generation wavelets. In preparation.

[110] B. Jawerth and W. Sweldens. Wavelet multiresolution analyses adapted for the fast solution of boundary value ordinary differential equations. In N. D. Melson, T. A. Manteuffel, and S. F. McCormick, editors, *Sixth Copper Mountain Conference on Multigrid Methods*, pages 259–273. NASA Conference Publication 3224, 1993.

[111] B. Jawerth and W. Sweldens. An overview of the theory and applications of wavelets. In Y.-L. O et al., editor, *Shape in Picture*, NATO ASI Series F, Vol. 126, pages 249–274, 1994.

[112] B. Jawerth and W. Sweldens. An overview of wavelet based multiresolution analyses. *SIAM Rev.*, 36(3):377–412, 1994.

[113] B. Jawerth and W. Sweldens. Biorthogonal smooth local trigonometric bases. *J. Fourier Anal. Appl.*, 2(1), 1995. (to appear), (`ftp://ftp.math.scarolina.edu/pub/imi_94/imi94_5.ps`).

[114] B. Jawerth and W. Sweldens. Weighted multiwavelets on general domains. In *Proceedings of the 14th Imacs World Congress*, To appear.

[115] R.-Q. Jia and C. A. Micchelli. Using the refinement equations for the construction of pre-wavelets ii: Powers of two. In P. J. Laurent, A. Le Méhauté, and L. L. Schumaker, editors, *Curves and Surfaces*, New York, 1991. Academic Press.

[116] F. Keinert. Biorthogonal wavelets for fast matrix computations. *Appl. Comput. Harmon. Anal.*, 1(2), 1993.

[117] S. Kelly, M. Kon, and L. Raphael. Pointwise convergence of wavelet expansions. *Bull. Amer. Math. Soc. (N.S.)*, 30(1):87–94, 1994.

[118] T. H. Koornwinder. Een golflet. In *Is er nog nieuws?* Boek aangeboden aan Prof. Dr. Th. J. Dekker ter gelegenheid van zijn emeritaat aan de Universiteit van Amsterdam op 27 november 1992.

[119] T. H. Koornwinder, editor. *Wavelets: an elementary treatment of theory and applications*. Number 1 in Series in Approximations and Decompositions. World Scientific, Singapore, 1993.

[120] N. Korneichuk. *Exact Constants in Approximation Theory*. Number 38 in Encyclopedia of mathematics and its applications. Cambridge University Press, 1991.

[121] A. Latto and E. Tenenbaum. Les ondelettes à support compact et la solution numérique de l'équation de burgers. *C. R. Acad. Sci. Paris Sér. I Math.*, 311:903–909, 1990.

[122] W. M. Lawton. Necessary and sufficient conditions for constructing orthonormal wavelets bases. *J. Math. Phys.*, 32(1):57–61, 1991.

[123] M. Ledoux and M. Talagrand. *Probability in Banach spaces*. Springer Verlag, Berlin, 1991.

[124] P. G. Lemarié. Some remarks on wavelet theory and interpolation. Preprint, Mathématiques, Paris XI, Orsay, 1991.

[125] P.-G. Lemarié. Ondelettes a localisation exponentielle. *J. Math. Pures Appl.*, 67(3):227–236, 1988.

[126] P.-G. Lemarié and Y. Meyer. Ondelettes et bases hilbertiennes. *Rev. Mat. Iberoamericana*, 2:1–18, 1986.

[127] P. G. Lemarié-Rieusset. Analyses multi-résolutions non orthogonales, commutations entre projecteurs et derivation et ondelettes vecteurs à divergence nulle. *Rev. Mat. Iberoamericana*, 8:221–238, 1992.

[128] R. A. Lorentz and W. R. Madych. Spline wavelets for ordinary differential equations. Preprint, Geselschaft für Mathematik und Datenverarbeitung, St. Augustin, Germany, 1990.

[129] Y. Maday, V. Perrier, and J.-C. Ravel. Adaptivité dynamique sur bases d'ondelettes pour l'approximation d'équations aux dérivées partielles. *C. R. Acad. Sci. Paris Sér. I Math.*, I(312):405–410, 1991.

[130] W. Madych. Some elementary properties of multiresolution analysis of $L_2(\mathbf{R})$. In [35], pages 259–294.

[131] S. Mallat and W. L. Hwang. Singularity detection and processing with wavelets. *IEEE Trans. Inform. Theory*, 38(2):617–643, 1992.

[132] S. Mallat and S. Zhong. Wavelet transform maxima and multiscale edges. In [156], pages 67–104.

[133] S. Mallat and S. Zhong. Characterization of signals from multiscale edges. *IEEE Trans. Patt. Anal. Mach. Intell.*, 14:710–732, 1992.

[134] S. G. Mallat. Multifrequency channel decompositions of images and wavelet models. *IEEE Trans. Acoust. Speech Signal Process.*, 37(12):2091–2110, 1989.

[135] S. G. Mallat. Multiresolution approximations and wavelet orthonormal bases of $L^2(\mathbf{R})$. *Trans. Amer. Math. Soc.*, 315(1):69–87, 1989.

[136] S. G. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Trans. Patt. Anal. Mach. Intell.*, 11(7):674–693, 1989.

[137] S. G. Mallat. Zero crossings of a wavelet transform. *IEEE Trans. Inform. Theory*, 37(4):1019–1033, 1991.

[138] H. S. Malvar. Lapped transforms for efficient transform/subband coding. *IEEE Trans. Acoust. Speech Signal Process.*, 38:969–978, 1990.

[139] H. S. Malvar and D. H. Staelin. The LOT: Transform coding without blocking effects. *IEEE Trans. Acoust. Speech Signal Process.*, 37:553–559, 1989.

[140] Y. Meyer. *Ondelettes et Opérateurs,* I: *Ondelettes,* II: *Opérateurs de Calderón-Zygmund,* III: (with R. Coifman), *Opérateurs multilinéaires*. Hermann, Paris, 1990. English translation of first volume, *Wavelets and Operators*, is published by Cambridge University Press, 1993.

[141] Y. Meyer. Ondelettes sur l'intervalle. *Rev. Mat. Iberoamericana*, 7:115–133, 1992.

[142] C. A. Micchelli. Using the refinement equations for the construction of pre-wavelets. *Numerical Algorithms*, 1(1):75–116, 1991.

[143] C. A. Micchelli, C. Rabut, and F. I. Utretas. Using the refinement equations for the construction of pre-wavelets iii: Elliptic splines. *Numerical Algorithms*, 1(1):331–352, 1991.

[144] M. Mitrea. *Singular integrals, Hardy spaces and Clifford wavelets*. Number 1575 in Lecture Notes in Math. 1994.

[145] P. Oswald. On discrete norm estimates related to multilevel preconditioners in the finite element method. In *Constructive Theory of Functions, Proc. Int. Conf. Varna 1991*, pages 203–214. Bulg. Acad. Sci., Sofia, 1992.

[146] J. Peetre. *New Thoughts on Besov Spaces*. Duke Univ. Math. Series, Durham, NC, 1976.

[147] R. Piessens and M. Branders. The evaluation and application of some modified moments. *BIT*, 13:443–450, 1973.

[148] R. Piessens, E. de Doncker, C. W. Uberhuber, and D. K. Kahanar. *QUADPACK: A Subroutine Package for Automatic Integration*. Springer Verlag, New York, 1983.

[149] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes*. Cambridge University Press, 2nd edition, 1993.

[150] Z. Qian and J. Weiss. Wavelets and the numerical solution of partial differential equations. 106:155–175, 1993.

[151] S. D. Riemenschneider and Z. Shen. Wavelets and pre-wavelets in low dimensions. *J. Approx. Theory*, 71(1):18–38, 1992.

[152] O. Rioul. Simple regularity criteria for subdivision schemes. *SIAM J. Math. Anal.*, 23(6):1544–1576, 1992.

[153] O. Rioul and M. Vetterli. Wavelets and signal processing. *IEEE Signal Proc. Mag.*, pages 14–38, October 1991.

[154] W. Royden. *Real Analysis*. Macmillan Publishing Company, New York, 3rd edition, 1988.

[155] W. Rudin. *Functional Analysis*. McGraw-Hill, Boston, 11th edition, 1987.

[156] M. B. Ruskai, G. Beylkin, R. Coifman, I. Daubechies, S. Mallat, Y. Meyer, and L. Raphael, editors. *Wavelets and their Applications*. Jones and Bartlett, Boston, 1992.

[157] N. Saito and G. Beylkin. Multiresolution representations using the auto-correlation functions of compactly supported wavelets. *IEEE Trans. Signal Process.*, 41(12):3584–3590, 1993.

[158] I. J. Schoenberg. Cardinal interpolation and spline functions. *J. Approx. Theory*, 2:167–627, 1969.

[159] I. J. Schoenberg. *Cardinal Spline Interpolation*. Number 12 in CBMS–NSF Series in Applied Mathematics. SIAM Publications, Philadelphia, 1973.

[160] L. L. Schumaker and G. Webb, editors. *Recent Advances in Wavelet Analysis*. Academic Press, New York, 1993.

[161] M. J. Shensa. Wedding the à trous and Mallat algorithms. *IEEE Trans. Signal Process.*, 40(10):2464–2482, 1992.

[162] E. P. Simoncelli, W. T. Freeman, E. H. Adelson, and D. J. Heeger. Shiftable multiscale transforms. *IEEE Trans. Inform. Theory*, 38(2):587–607, 1992.

[163] E. M. Stein. *Harmonic Analysis: Real-variable methods, orthogonality, and oscillatory integrals*. Princeton University Press, 1993.

[164] J. Stöckler. Multivariate wavelets. In [35], pages 325–356.

[165] G. Strang. Wavelets and dilation equations: A brief introduction. *SIAM Rev.*, 31(4):614–627, 1989.

[166] G. Strang. Wavelet transforms versus Fourier transforms. *Bull. Amer. Math. Soc. (N.S.)*, 28(2):288–305, 1993.

[167] G. Strang and G. Fix. *An analysis of the finite element method*. Prentice Hall, 1973.

[168] G. Strang and G. Fix. A Fourier analysis of the finite element variational method. In *Constructive aspects of Functional Analysis*. Edizione Cremonese, Rome, 1973.

[169] J. 0. Strömberg. A modified Franklin system and higher order spline systems on $\mathbf{R}^n$ as unconditional bases for Hardy spaces. In Beckner et al., editor, *Conference on Harmonic Analysis in Honor of Antoni Zygmund*, volume II, pages 475–494, Chicago, 1981. Univ. of Chicago Press.

[170] A. H. Strout and D. Secrest. *Gaussian Quadrature Formulas*. Prentice-Hall, Englewood Cliffs, New Jersey, 1966.

[171] W. Sweldens. Asymptotic error expansions for spline wavelets and connections with Euler and Bernoulli splines. Manuscript.

[172] W. Sweldens. Weighted wavelets. In preparation.

[173] W. Sweldens. The construction of compactly supported wavelets, biorthogonal with respect to a weighted inner product. In *Proceedings of the 14th Imacs World Congress*, To appear.

[174] W. Sweldens and R. Piessens. Calculation of the wavelet decomposition using quadrature formulae. In [119], pages 139–160.

[175] W. Sweldens and R. Piessens. Calculation of the wavelet decomposition using quadrature formulae. *CWI Quarterly*, 5(1):33–52, 1992.

[176] W. Sweldens and R. Piessens. Asymptotic error expansions of wavelet approximations of smooth functions ii. *Numer. Math.*, 3(68):377–401, 1994.

[177] W. Sweldens and R. Piessens. Quadrature formulae and asymptotic error expansions for wavelet approximations of smooth functions. *SIAM J. Numer. Anal.*, 31(4):1240–1264, 1994.

[178] W. Sweldens and R. Piessens. Wavelet sampling techniques. To appear. Proceedings of the Joint Statistical Meetings, San Francisco, August 1993.

[179] W. Sweldens and K. Yarnall. Implicit sampling representations of functions. In preparation.

[180] Ph. Tchamitchian. Ondelettes et intégrale de Chauchy sur les courbes lipschitziennes. *Ann. Math.*, 129:641–649, 1989.

[181] M. Unser and A. Aldroubi. Polynomial splines and wavelets — a signal processing perspective. In [35], pages 543–601.

[182] M. Unser, A. Aldroubi, and M. Eden. On the asymptotic convergence of B-spline wavelets to Gabor functions. *IEEE Trans. Inform. Theory*, 38:864–872, 1992.

[183] M. Unser, A. Aldroubi, and M. Eden. A family of polynomial spline wavelet transforms. *Signal Process.*, 30:141–162, 1993.

[184] G. Uytterhoeven. Ontwerp en implementatie van compressiealgoritmen met multi-wavelets. Master's thesis, Departement Computerwetenschappen, K.U.Leuven, Belgium, 1994. (in Dutch).

[185] P. P. Vaidyanathan. Theory and design of M-channel maximally decimated quadrature mirror filters with arbitrary M, having perfect reconstruction property. *IEEE Trans. Acoust. Speech Signal Process.*, 35(2):476–492, 1987.

[186] P. Verlinden. *Cubature Formulas and Asymptotic Expansions.* PhD thesis, K.U.Leuven, Belgium, 1993.

[187] P. Verlinden and A. Haegemans. An asymptotic expansion in wavelet analysis and its application to accurate numerical wavelet decomposition. *Numer. Algor.*, 2:287–298, 1992.

[188] M. Vetterli. Filter banks allowing perfect reconstruction. *Signal Process.*, 10:219–244, 1986.

[189] M. Vetterli and C. Herley. Wavelets and filter banks: theory and design. *IEEE Trans. Acoust. Speech Signal Process.*, 40(9):2207–2232, 1992.

[190] L. F. Villemoes. Energy moments in time and frequency for two-scale difference equation solutions and wavelets. *SIAM J. Math. Anal.*, 23(6):1119–1543, 1992.

[191] L. F. Villemoes. Wavelet analysis of refinement equations. *SIAM J. Math. Anal.*, 25(5):1433–1460, 1994.

[192] J. S. Walker. *Fast Fourier transforms.* Studies in Advanced Mathematics. CRC Press, Boca Raton, 1991.

[193] G. K. Wallace. The JPEG still picture compression standard. *Comm. ACM*, 34(4):30–44, 1991.

[194] G. G. Walter. A sampling theorem for wavelet subspaces. *IEEE Trans. Inform. Theory*, 38:881–884, 1992.

[195] M. V. Wickerhauser. Acoustic signal compression with wavelet packets. In [35], pages 679–700.

[196] M. V. Wickerhauser. Lectures on wavelet packets algorithms. 1991.

[197] J.-C. Xu and W.-C. Shann. Galerkin-wavelet methods for two-point boundary value problems. *Numer. Math.*, 63(1):123–142, 1992.

[198] K. Yarnall. PhD thesis, University of South Carolina, 1992.

[199] H. Yserentant. On the multi-level splitting of finite element spaces. *Numer. Math.*, 49:379–412, 1986.

[200] J. Zak. Finite translations in solid state physics. *Phys. Rev. Lett.*, 19:1385–1397, 1967.

[201] A. Zygmund. *Trigonometric Series*. Cambridge University Press, London, 2nd edition, 1959.

"Double, double toil and trouble;
Fire burn and cauldron bubble
(Witches)"
—Shakespeare, *Macbeth (1623).*

List of Figures

List of Tables

List of Algorithms

Index

"The Art of Taking Time to Live"

To get the most out of life,
we must take time to live
as well as to make a living.
We must practice the art of filling our moments
with enriching experiences that will
give new meaning and depth to our lives.

We should take time for good books;
time to absorb the thoughts of poets and
philosophers, seers and prophets.

Time for friendships; time for talks by
the fire and walks beneath the stars.

Time for travel; time for pilgrimage
and festival, for shrine and exhibit,
for rockbound cost and desert,
mountain and plain.

Time for nature; time for flower gardens,
trees, birds and sunsets.

Time to love and be loved, for love is the
greatest thing in the world.

Time for people; time for the interplay
of personalities and the interchange of
ideas.

Time for solitude; time to be quiet and
alone and to look within.

Time to give of ourselves, our talents,
abilities, devotions, convictions, that we
may contribute to the onward march
of man.

from "The Art of the Great Life" by Wilferd A. Peterson

The research leading to this thesis could never have been done without the financial support of the following institutions:

Barco

Centrum voor Wiskunde en Informatica

Dutch Mathematical Conference 1993

Efficient Program for Stimulation of Competitive Research
(NSF Grant EHR 9108772)

Howard University

IBM

Illinois Institute of Technology

Katholieke Universiteit Leuven

Nationaal Fond voor Wetenschappelijk Onderzoek België

Nato workshop "Shape in Picture" 1992

Office of Naval Research (Grant N00014-90-J-1343)

Purdue University

Seventh Texas International Symposium on Approximation Theory 1992

Texas A&M University

The Mathworks

University of South Carolina

Vlaamse Leergangen

"This is the end ... my only friend, the end."
—Jim Morrison, *(1969).*